

Modeling of Image Variability for Recognition

Von der Fakultät für
Mathematik, Informatik und Naturwissenschaften der
Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von
Diplom-Informatiker
Daniel Martin Keyzers
aus Düsseldorf

Berichter: Universitätsprofessor Dr.-Ing. Hermann Ney
Universitätsprofessor Dr.-Ing. Hans Burkhardt

Tag der mündlichen Prüfung: 14. März 2006

Diese Dissertation ist auf den Internetseiten der Hochschulbibliothek online verfügbar.

Abstract

This thesis presents the application of different models of image variability to visual recognition problems using the paradigm of appearance-based recognition. We first discuss linear models of variability and relate them to the use of Gaussian distributions. This allows us to use well-understood estimation methods to determine the vectors representing the variability. We also relate the discriminative maximum entropy approach to the Gaussian case and use the relationship to derive the novel maximum entropy linear discriminant analysis. Secondly, we investigate discrete deformation models — that map pixels onto pixels — of order zero, one, and two, where the order is determined by the constraints imposed on the two-dimensional image distortion. We prove for the first time that the determination of the best match for the second order model belongs to the class of NP-hard problems. We show that it is important to include a suitable context for each pixel to achieve low error rates, which is then possible using the less complex models of lower order. We furthermore discuss the use of local patches for visual object categorization as a model allowing high image variability and show how the use of discriminative training leads to very competitive results. Finally, we describe a model for holistic scene analysis that allows us to determine a visual representation of objects present in a set of images.

The methods are primarily applied to the tasks of handwritten character recognition and medical image categorization, yielding excellent results in both cases. In particular, we achieve an error rate of 0.52% on the well-known MNIST benchmark and 12.6% on the IRMA-10,000 database, the lowest within the 2005 ImageCLEF evaluation. We show that the models of image variability also improve the recognition performance of appearance-based sign language and gesture recognition systems. This emphasizes the models' broad applicability.

Zusammenfassung

In dieser Arbeit werden verschiedene Modelle zur Beschreibung von Variabilität in Bildern für die erscheinungsbasierte Klassifikation von Objekten untersucht. Zunächst werden lineare Variabilitätsmodelle behandelt und ihre Beziehung zu Gaußverteilungen dargestellt. Diese Beziehung erlaubt es, bekannte Schätzverfahren zur Bestimmung der Vektoren, die die Variabilität repräsentieren, einzusetzen. Darüber hinaus wird der Zusammenhang zwischen dem diskriminativen ‚Maximum Entropy‘-Ansatz und Gaußverteilungen behandelt, der dann verwendet wird um die neuartige ‚Maximum Entropy Linear Discriminant Analysis‘ herzuleiten. Weiterhin werden diskrete Verformungsmodelle der Ordnung null, eins und zwei untersucht, die Bildpixel auf Bildpixel abbilden. Dabei wird die Ordnung der Modelle durch die Art der angesetzten zweidimensionalen Verformungseinschränkungen bestimmt. Es wird erstmals gezeigt, dass die Bestimmung der besten Abbildung zwischen zwei Bildern für das Modell der Ordnung zwei zur Klasse der NP-harten Probleme gehört. Für die diskreten Modelle ist es entscheidend, dass ein geeigneter Kontext der Pixel verwendet wird, um niedrige Fehlerraten zu erreichen. Durch die Hinzunahme von Kontext werden auch mit den weniger komplexen Modellen geringerer Ordnung sehr gute Fehlerraten erreicht. Des Weiteren werden Modelle zur Behandlung von starker Variabilität in der Objekterkennung erörtert, die auf der Verwendung von lokalen Bildteilen basieren, und die in Kombination mit diskriminativen Trainingsverfahren sehr konkurrenzfähige Ergebnisse erlauben. Schließlich wird ein Modell für die holistische Bildanalyse vorgestellt, das es erlaubt, eine visuelle Beschreibung von Objekten zu bestimmen, die in einer gegebenen Menge von Bildern vorhanden sind.

Die Anwendung der vorgestellten Methoden wird vor allem für die Klassifikation handgeschriebener Zeichen und die Kategorisierung von medizinischen Bildern untersucht, wobei in beiden Fällen hervorragende Ergebnisse erzielt werden. Insbesondere wird eine Fehlerrate von 0,52% auf der bekannten Datenbank MNIST erzielt und eine von 12,6% auf der Datenbank IRMA-10.000, welche die niedrigste in der 2005 durchgeführten ImageCLEF-Evaluation ist. Es wird außerdem gezeigt, dass die Variabilitätsmodelle auch die Resultate für die erscheinungsbasierte Erkennung von Gesten und Gebärdensprache verbessert, was die breite Anwendbarkeit der beschriebenen Modelle unterstreicht.

Acknowledgments

At this point, I would like to express my gratitude to the people who supported and accompanied me during the preparation of this work. Assuming that this section is the part of the thesis that will be read most often (and for many people it will be the only part they read), I tried to put a special emphasis on it (as you can probably infer from its length).

My foremost thanks go to Prof. Dr.-Ing. Hermann Ney, head of the Lehrstuhl für Informatik VI of RWTH Aachen University, for supervising this thesis. He gave me the opportunity to pursue my ideas and to cooperate with many interesting and helpful colleagues and students at his institute. He supported me with numerous ideas and discussions, made it possible for me to attend a variety of conferences and meetings, and gave me the possibility to head the image processing and recognition group at the Lehrstuhl für Informatik VI. His expertise in all fields of pattern recognition was always amazing and helpful. (“Hasn’t Smith written something about that? Have a look at that book, around page 142.”) From him I also learned most of what I know about how to write scientific articles, proposals, or reports (although I am not sure if I succeeded in meeting all of his criteria with this document).

I am very grateful to Prof. Dr.-Ing. Hans Burkhardt for agreeing to take the time to evaluate this thesis as a co-referee, for his interest in my work, and for interesting discussions on various aspects of pattern recognition.

Probably the two people at the Lehrstuhl most important for this work were Jörg Dahmen, head of the image group until 2001 and supervisor of my diploma thesis, and Thomas Deselaers, head of the group from 2005 on and former diploma thesis student under my supervision. I am grateful to Jörg for first introducing me to the topics I worked on, for asking me to join the research group as a student researcher, for years of fruitful cooperation, for many enjoyable conference journeys and of course for teaching me everything I know about the English language. I equally want to thank Thomas for many things, for the loads of fun while working in an effective team, for teaching me why bash is better in ‘comparishon’ to tcsh, and for the many interesting and entertaining trips we had. I am also thankful that Thomas kindly refrained from throwing wooden shoes and instead helped me in various ways while I was writing this document. Both Jörg and Thomas made it possible in their own ways to arrive at many interesting results in our team. This is of course also true for the other members of the image group during the time I spent at the Lehrstuhl, especially the other diploma thesis students I had the chance to supervise: Philippe Dreuw, Christian Gollan, Tobias Kölsch, and Michael Motter. This thesis would have much less content without their interesting contributions.

Thanks also go to the student researchers of the image group, to Ilja Bezrukov and David Rybach for holistic results, to Andre Hegerath for entropic results, to Stephan Mayer and Sami Celik for practical results. I would also like to thank Morteza Zahedi for his contributions and Roberto Paredes for many interesting discussions during his stay in Aachen, not only about ‘Rasterelektronenmikroskopie’ and ‘the power ranger’.

I thank all the members of our data mining team, especially Thomas, Arne, Ilja, and Andre for their enthusiasm, the very presentable results, and the interesting journeys originating from these.

Many thanks go to my colleagues, who not only gave helpful comments and hints, but who made the lunch hours and the coffee breaks in Café Bender interesting and fun and created such a friendly atmosphere: the old school, Achim ‘nee, lass mal’, ‘Skater’ Florian, Frank, `typedef Franz F` [11am, working early?], ‘Erdalkali’ Jörg, ‘Joghurt’ Kalle

& Wolle -ansi -pedantic -macherey, std::Max, Michael ‘I’ll sue them’, Ralf ‘Weichmacher’, Sirko ‘Kepler’, Sonja ‘Wir gehen jetzt’, and ostream_iterator<Stephan>; the new school, András ‘no more beer?’, Maja, Morteza ‘pssst’, Nicola ‘Mate de Coca? Schon!’, Oliver ‘Der Kaffee ist fertig!’, Richard, and Shahram; and the newbies, Björn, Christian ‘training on the database’, ‘metapost’ David, Evgeny, Georg, Jan ‘in Ulm ist immer Nebel’, Jia ‘hi’, Jonas, NicolB ‘Hi Thomas, uh, Daniel’, Philippe ‘arschitecture’, Saša ‘keine Zeit’, Thomas, and Yuqi. I also want to thank my office-mates Jan and Morteza, Shahram, Sonja, Stephan, and Tibor for the conversations and for putting up with me, and also Gisela ‘ich dachte ja nur’, Mirko ‘saugel’, and Stefan ‘the builder’ for keeping everything running. It was a great pleasure to be a member of such an excellent team.

I am very thankful for the friendly atmosphere and the support I received at the Instituto Tecnológico de Informática in Valencia during my stays there in 2002. Most of all I want to thank Enrique, who was always very encouraging, and Alfons, Rafa, Javi, and Alejandro. A very special thanks however goes to Ismael, who helped me in so many ways and organized everything from a place to stay to trips across Spain, and to Roberto, who contributed a lot to making the stays both stimulating and fun.

I would like to thank all researchers that made data, results, or software that I used available. Without their contributions, this thesis would not exist in its present form.

I also want to take this opportunity to thank several people for various reasons: Thomas Lehmann and the whole IRMA-team for constructive discussions, data, seminars, and barbecues; Volker Steinbiss for many helpful tips; Bernt Schiele for his interest, for taking the time to discuss things with me, and for many useful recommendations; Alex, Jörg, Marc, Martin, Michael, Rachid, and the other soccer players for some physical exercise; Thomas Breuel for introducing me to new topics and for providing the possibility to continue research in related areas; Christoph Lampert for more soccer and for insisting that I would not be a *real* post-doc until I finished this thesis; Rainer Lindwurm for the fruitful cooperation; Walter Unger, Bernard Haasdonk, Alfons Juan, and Alejandro Toselli for the possibility to work on some very interesting papers with them; and Thomas Deselaers, Jörg Herbers, Christoph Lampert, and Daniel Wright for proof-reading parts of this document.

I am very grateful for all the time spent with Katrin & Jörg, Melanie & Klaus, Sarah & Christian, and Dimitra & Franz. They made Aachen a fun and exciting place to work. I am also very thankful for the support I received from my family, which since 2002 ‘officially’ includes many more people, who have contributed their shares to the life outside University.

Last but definitively not least I have to thank Christina for her love, encouragement, and patience (when I said I would hand in this thesis in December, in February, in April, in June, ...) and for getting up at night and letting me sleep when Philipp needed something. I also want to thank Philipp for just being there and reminding me that there are things more important than this work.

This dissertation was written during my time as a researcher with the Lehrstuhl für Informatik VI of RWTH Aachen University in Aachen, Germany. This work was partially funded by the DFG (Deutsche Forschungsgemeinschaft) under contract NE-572/6.

Contents

1	Introduction	1
2	Scientific goals	5
3	Data sets and state of the art	7
3.1	Handwritten character recognition	7
3.2	Medical images	20
3.3	General images and complex scenes	28
4	Pattern recognition for image classification	35
4.1	Basic structure of a classifier	36
4.2	Bayes' decision rule	38
4.3	Other classification approaches	39
4.4	Parameter estimation	41
4.5	Dimensionality reduction	43
4.6	Distance-based classification	48
4.7	Invariance in classification	50
5	Gaussian and related models	59
5.1	Gaussian models	59
5.2	Tangent vectors in Gaussian models	61
5.3	Maximum entropy models	76
5.4	Maximum entropy linear discriminant analysis	90
5.5	Conclusion	96
6	Nonlinear deformation models	97
6.1	Related work on image matching	98
6.2	Framework for recognition using nonlinear matching	101
6.3	Second-order: two-dimensional model	112
6.4	Two-dimensional matching is NP-complete	116
6.5	First-order: one- and pseudo two-dimensional models	125
6.6	Zero-order: the image distortion model	130
6.7	Matching using the Hungarian algorithm	133
6.8	Gesture and sign language recognition using models of variability	139
6.9	Conclusion	142
7	Recognition based on local patches	145
7.1	Introduction and related work	145
7.2	Feature extraction	149
7.3	Different models for patch-based classification	154

7.4	Experiments and results using histograms	163
7.5	Combination of patch-based classification and tangent distance	171
7.6	Conclusion	173
8	Holistic scene analysis	175
8.1	Introduction and related work	175
8.2	Statistical model and training	179
8.3	Experiments and results	182
8.4	Conclusion	188
9	Conclusion	191

1 Introduction

This theory goes as follows and begins now. All brontosauruses are thin at one end, much much thicker in the middle, and then thin again at the far end. That is my theory, it is mine, and belongs to me and I own it, and what it is, too.

– A. Elk (*Monty Python*)

The goal of this thesis is to show that appearance-based two-dimensional models of image variability lead to state-of-the-art recognition results for diverse image analysis applications.

Specifically, we will demonstrate how to apply linear and nonlinear models of variability to recognition tasks ranging from the categorization of medical images to the analysis of image sequences containing human gesture. However, the special focus of this thesis is the application of handwritten character recognition, for which we show that simple models of variability combined with a suitable representation of the image pixel context lead to very competitive results across five different tasks, as measured by the error rate.

It is well-known that the robustness with respect to transformations of the input is an important subject in pattern recognition in general, and for the recognition of objects in images in particular. For many tasks the recognition accuracy can be significantly improved by explicitly modeling the variability of the data. This is especially effective in cases where the training set is small, thus introducing a-priori knowledge into the classification algorithm that cannot be extracted from the training data.

Often, the method upon which is focused to achieve robustness is the extraction of invariant features from the input. In many image recognition tasks this involves the segmentation of the input, which can lead to errors if the segmentation is difficult, as, for example, in the presence of clutter or an inhomogeneous background. We investigate the application of different models of image variability to visual recognition problems using the paradigm of appearance-based recognition, i.e. we operate directly on the image as given by its pixel intensities.

Our approach of using models of image variability to yield invariant distance functions between images can be regarded as the implicit construction of a much enlarged training data set. However, the best fitting prototypes from the large number of possible transformed training samples are not explicitly constructed, but instead determined using the matching algorithms.

We discuss linear deformation models and discrete deformation models of order zero, one, and two. For the discrete models, which map pixels to pixels, we observe that it is important to include a suitable representation of the image context for each pixel. The models are applied to the tasks of handwritten character recognition and the categorization of medical images, yielding excellent results in both cases. In particular, we achieve an error rate of 0.52% on the popular MNIST benchmark and the best result, a 12.6% error rate, within the 2005 ImageCLEF/IRMA evaluation. Finally, we show that the models of image variability

also improve the recognition performance of appearance-based sign language and gesture recognition systems, which underlines the broad applicability of the models of variability. We discuss the use of local patches for visual object categorization and show how the use of discriminative training leads to very competitive results. Classification using local patches can be seen as the employment of highly deformable models.

Throughout this work, to allow for a meaningful comparison of different recognition approaches, the proposed classifiers are applied to different standard corpora, for which other research groups have produced many results. We also present results on some data sets for which this statement is not true, to compare various aspects of the recognition systems and underline their broad applicability.

The main contributions of this work are summarized on page 6.

Application areas for image object recognition

Automatic image analysis and object recognition in images is an important task in many real-world applications, for example

- the recognition of handwritten characters and digits, enabling for example the automatic reading of bank checks and postal envelopes;
- medical applications, such as the automated evaluation of medical image data; typical tasks are for instance the counting of cells in a medical probe, the diagnosis of skin lesions as malignant or benign, or the retrieval of a similar image from a database;
- image and video indexing in large databases; interpreting an image as to be composed of multiple objects, an image index can be automatically obtained by detecting and classifying the objects present in a given scene or by defining suitable global image similarity measures;
- robot vision, including for example the use in autonomous vehicles or driver assistance systems that warn the driver of a car about pedestrians that are about to cross the path of the vehicle;
- automatic license plate reading of vehicles entering and leaving a car park or passing toll roads;
- industrial applications such as quality control, for example the matching of solar cell wafers before and after processing to allow for an evaluation of the production process with respect to wafer quality;
- biometric applications, such as fingerprint or face recognition; these applications are widely assumed to be crucial for the successful implementation of modern security systems.

Although the above list is far from complete, it shows that object recognition is an important tool that can be used in many practical applications. As the considerations in the following chapters show, state-of-the-art results in image object recognition can be obtained by using the described models of variability.



Figure 1.1: Examples of image variability: handwritten digits ‘2’ from the USPS data set.

Within the domain of image analysis, the recognition of handwritten characters is one of the oldest tasks that has been investigated by researchers and engineers. The success of automatic analysis methods is underlined by the fact that today millions of postal envelopes are automatically sorted using image analysis techniques. A reduction of the error rate has an immediate effect for such systems. Although this task has been investigated for a long time, it still remains prototypical for robust object recognition, because at the same time

- different writing styles and pens lead to strongly varying appearances, and
- the reference model for each class is very well-defined.

Figure 1.1 shows nine different images of handwritten digits ‘2’ that were taken from zip codes written on postal envelopes. Although we can recognize each image as representing a ‘2’, the variability is very large. We therefore use the task of handwritten character as a test-bed for the performance of the investigated models of variability throughout this work.

Structure of this document

The work is organized as follows: The following chapter briefly summarizes the goals of this work. Chapter 3 introduces the data sets used in the experiments, summarizes the methods used in current research, and discusses the error rates presented by other researchers. Chapter 4 introduces the basic concepts of pattern recognition in the context of image analysis, which serve as the basis for the material in the following chapters. Chapter 5 discusses Gaussian and related models including the linear model of variability, the tangent distance. Due to the relation to Gaussian models, this chapter also includes the discussions of the maximum entropy framework for classification and the derivation of the maximum entropy linear discriminant analysis. Chapter 6 then discusses the nonlinear deformation models that use pixel-to-pixel mappings of the images, while Chapter 7 presents some results in the context of classification using local image patches. Chapter 8 discusses a holistic model that explains a complex scene by implicitly assigning the parts of the image to the fore- and background, respectively. Experimental results for the different techniques are presented along with their descriptions in each chapter. Finally, Chapter 9 concludes the text with an overview of the main results.

2 Scientific goals

His principal research interests include scale space theory, folklore theorems, the back-transmutation of gold into lead, linear and nonlinear dimensionality reduction methods, and pattern analysis techniques for supervised image processing.

– M. Loog, 2004

The goals set out at the starting point of the work for this thesis (and supplemented at different points in time along the work) were to

- improve upon the state-of-the-art results for various image classification tasks (measured by the error rate) using statistical and appearance-based models;
- investigate the applicability of different models of image variability in this context, especially by using image context at the pixel level for nonlinear models;
- investigate the applicability of the Hungarian algorithm for image matching;
- investigate the use of the maximum entropy framework and discriminative training for classification and feature extraction in the context of image classification;
- investigate the applicability of the appearance-based approach for video analysis, i.e. for gesture and sign language recognition;
- investigate the use of local patches for image object recognition, which leads to good results for tasks such as face recognition.
- investigate the applicability of the holistic approach for image analysis;

These goals were planned to be pursued in the context of the basic concepts followed at the Lehrstuhl für Informatik VI:

- analyses of classification concepts start with Bayes' decision rule;
- use the appearance-based approach, i.e. avoid segmentation of the image object whenever possible and use features directly derived from the pixel intensities of the images for classification;
- try to learn from the experience gained in speech recognition and machine translation, e.g. with respect to discriminative training;
- regard the performance for handwritten character recognition and the classification of medical images, which are the two standard tasks used in the past.

The main contributions of this work are:

- to show that two conceptually simple models of image variability (the tangent distance as a linear model and the image distortion model using pixel contexts as a nonlinear model) lead to consistent improvements in a variety of real-world image recognition tasks;
- to show that the straightforward paradigm of appearance-based image classification with appropriate models of variability leads to very competitive results in the domain of handwritten character recognition, for which a large number of results from other researchers are available;
- to prove the competitiveness of the appearance-based approach in a number of different settings; for example, it is novel that an appearance-based approach can lead to state-of-the-art results for the analysis of gestures in video;
- to experimentally confirm that the use of fewer two-dimensional constraints in image matching can be compensated by using the local image context at the pixel level;
- to prove that the computation of the optimal matching of two images under two-dimensional constraints is a computationally hard problem; although it has been well established that no polynomial time algorithms are known for this problem, we provide a proof that the problem is NP-hard;
- to investigate the applicability of the Hungarian algorithm for pixel-to-pixel matchings of images for recognition;
- to show the applicability of discriminative training in the form of the maximum entropy framework in the context of image analysis, e.g. for the direct estimation of the posterior probability and for the use in patch-based classification;
- to discuss the relationship between maximum entropy (or log-linear) models and Gaussian models and to show that the maximum entropy framework can be effectively used to discriminatively train the equivalent of full covariance matrices for Gaussian models;
- the derivation of a novel discriminative linear dimensionality reduction technique based on the maximum entropy framework, i.e. the maximum entropy linear discriminant analysis.

3 Data sets and state of the art

But they don't know we know they know we know!
– Phoebe (*Friends*)



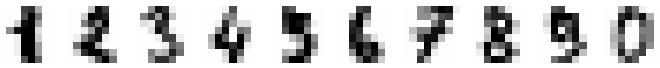

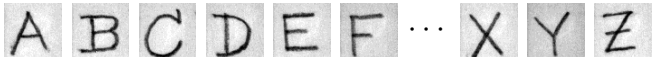
This chapter contains an overview of the recognition problems considered within this work. We also provide brief overviews of the methods that other authors use to obtain the results presented where appropriate. We also anticipate results based on the methods discussed in later chapters and put them into their frame of reference here. Related work on classification and other techniques is discussed in Chapter 4. Work related to the approaches that are discussed in the main part is described in each of the individual chapters to serve as a context for the discussions.

In the comparisons of the results we concentrate on the error rate as a measure of performance, although in practical applications also other aspects have to be considered, as e.g. the algorithmic resources used in training and classification. These are disregarded in the basic comparisons here. The following sections are grouped according to the type of the addressed problem. Each section introduces datasets that can be used to evaluate different approaches for such tasks along with known results for these datasets.

3.1 Handwritten character recognition

Handwritten character recognition continues to be a very relevant task within the pattern recognition domain. Two important applications are the automatic reading of addresses on postal envelopes and of filled-in forms like questionnaires or money orders. Digit recognition serves as an evaluation task because the problem is well defined and common databases are in widespread use. “A digitized handwritten numeral can be represented as a binary or gray

Table 3.1: Corpus and image sizes and example images.

name	example images	size	#train	#test
USPS		16×16	7 291	2 007
MNIST		28×28	60 000	10 000
UCI		8×8	3 823	1 797
MCEDAR		8×8	11 000	2 711
ETL6A		64×63	15 600	13 000

scale image. An important pattern recognition task that has received much attention lately is to automatically determine the digit, given the image.” [Hastie & Simard 98]

We assume here, that the individual elements are already segmented and thus given as separate images. In many applications it will be more appropriate to classify a whole sequence of characters or even a complete text or address image, because this allows us to take into account the interdependencies between the individual decisions. Furthermore, if a sequence of characters is to be classified, even if segmentation is not a problem, we might be interested in not only classifying each character correctly, but to also estimate reliable measures of confidence in these decisions. Nevertheless, when dealing with the tasks presented here, we only measure the overall classification accuracy of a single-image-based classification process. We do so because for the datasets described here generally no other information is available and it is common practice to publish the error rate as a summary of the performance of an approach.

Segmented character recognition data sets are among the most widely used reference data sets in image recognition. Another reason to use handwritten digit data is that the images are in various ways prototypical for a recognition problem with high demands on robustness, as already briefly discussed in the introduction (cp. Figure 1.1). Table 3.1 shows an overview and example images for the five data sets of handwritten characters that are used in this work and described in more detail in the following. Along the way we will discuss various methods that are used for the classification of these images and cite results that are presented in other publications. Naturally, we cannot discuss all possible classification strategies that are in use, but we limit the discussion to those that provide reference results on standard, publicly available data sets. For example there are state-of-the-art systems for the recognition of handwritten characters that are not referenced in the following sections and that are in use in the postal industry for many years now. Such systems use for example hierarchical classifiers with network structures based on polynomial decision functions [Lindwurm & Breuer⁺ 96].

3.1.1 The US Postal Service task

The well-known United States Postal Service Handwritten Digit Database (USPS) consists of isolated and normalized images of handwritten digits taken from US mail envelopes. The references to these data go back to [Wang & Srihari 88, LeCun & Boser⁺ 89, LeCun & Boser⁺ 90]. The images were originally binary and of sizes between 40 and 60 pixels. The images were then down-scaled (keeping the aspect ratio) and linearly transformed resulting in multiple gray levels. (The data we use is quantized to 1,000 gray levels.) The scaled images are of the size 16×16 pixels. The database contains a separate training and test set, where the training set includes 7,291 images and the test set consists of 2,007 samples. We present this data set first because most of the experiments for this work regarding handwritten character recognition were performed using these data.

Different versions of this database are available. The experiments performed here are based on the data as available via FTP from the Max Planck Institute Tübingen¹. The USPS data are also used as an example data set in [Hastie & Tibshirani⁺ 01] and thus available on the corresponding web site². The data is given in the range [-1:1] at these sites and was

¹<ftp://ftp.kyb.tuebingen.mpg.de/pub/bs/data>

²<http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/data.html>

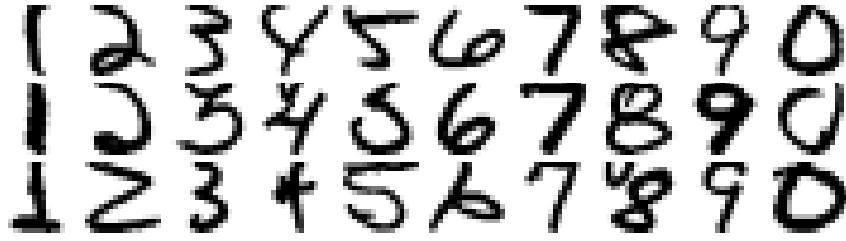


Figure 3.1: Some examples images from the USPS test set.



Figure 3.2: Some ‘difficult’ examples from the USPS test set along with their target labels.

These samples illustrate the existence of some labeling errors (at least without availability of the context) and make the estimated human error rate of 1.5%–2.5% seem reasonable.

transformed to the range $[0:2]$ for the experiments. The exact data used in the experiments here are also available³.

Figure 3.1 shows some example images for each of the ten classes taken from the USPS corpus. Despite the preceding normalization step there is still a large amount of variability in the data that the classifier needs to take into account. Furthermore one can see artifacts due to the segmentation from an area containing more writing, for example in the image of the digit ‘8’ in the last row of Figure 3.1.

The USPS test set is known as a hard recognition task which can be concluded from the human error rate on the data of 2.5% as given in [Simard & Le Cun⁺ 93]. A slightly lower human error rate of 1.5% was estimated in [Dong & Krzyzak⁺ 02b]. Figure 3.2 shows some ‘difficult’ test samples along with the target class label. The notion of difficulty for the choice of the samples in the figure is based on the ability of an automatic classifier to correctly classify the data here [Keysers & Dahmen⁺ 00b].

One disadvantage of the USPS corpus is that there exists no development test set, which leads to effects known as ‘training on the testing data’ for each of the research groups performing experiments. The tendency exists to evaluate one method with different parameters or different methods several times on the same data until the best performance seems to have been reached. This procedure leads to an overly optimistic estimation of the error rate of the classifier and the number of tuned parameters should be considered when judging such error rates. Ideally, a development test set would be used to determine the best parameters for the classifiers and the results would be obtained from one run on the test set

³<http://www-i6.informatik.rwth-aachen.de/~keysers/usps.html>

Table 3.2: Error rates for the USPS task [%]. Error rates using the USPS+ training set are indicated by *. (Abbreviations: see page 193f.)

reference / group	method	ER
[Simard & Le Cun ⁺ 93] / AT&T	human performance	2.5
[Dong & Krzyzak ⁺ 01] / CENPARMI	human performance	1.5
– / –	Euclidean nearest neighbor	5.6
[Vapnik 95] / AT&T	decision tree	16.2
– / –	logistic model tree (WEKA toolkit)	8.8
[Dahmen & Keysers ⁺ 01b] / i6	Gaussian mixture densities (baseline)	7.2
[Keysers & Och ⁺ 02a] / i6	maximum entropy second order feat.	5.6
this work / i6	Gabor features, maximum entropy	5.4
[Tipping 00] / Microsoft	relevance vector machine	5.1
[LeCun & Bottou ⁺ 98] / AT&T	neural net LeNet1	5.0
[DeCoste & Schölkopf 02] / AT&T	optimal margin classifier	4.6
[LeCun & Boser ⁺ 90] / AT& T	neural net	*4.6
[Zhang & Huang ⁺ 05] / NU Singapore	kernel auto-associator	4.4
[Bottou & Cortes ⁺ 94] / AT&T	neural net	4.2
[Keysers & Dahmen ⁺ 00b] / i6	kernel densities, virtual data	4.2
[Schölkopf 97] / TU Berlin	support vector machine	4.0
[Perrey 00] / i6	invariant features	4.0
[Haasdonk & Keysers 02] / LMB+i6	support vector m. + tangent distance	3.6
[Haasdonk & Halawani ⁺ 04] / LMB	support vector m. + invariant features	3.5
[Dahmen & Keysers ⁺ 01b] / i6	LDA, virt. data, Gauss. mix. dens.	3.4
[DeCoste & Schölkopf 02] / AT&T	local learning	*3.3
[Keysers & Dahmen ⁺ 00b] / i6	one-sided tangent distance	3.3
[Haasdonk & Vossen ⁺ 05] / LMB	invariant support vector m.	3.2
[Schölkopf 97] / TU Berlin	invariant support vector machine	3.0
[Keysers & Dahmen ⁺ 00b] / i6	two-sided tangent distance	3.0
[Keysers & Gollan ⁺ 04b] / i6	1-NN, 2-D deformation model	2.7
[Drucker & Schapire ⁺ 93] / AT&T	neural net + boosting	*2.6
[Kölsch 03] / i6+ITI	tangent distance for local patches	2.6
[Simard & Le Cun ⁺ 93] / AT&T	tangent distance	*2.5
[Dong & Krzyzak ⁺ 02b] / CENPARMI	preprocessing, support vector machine	2.5
[Keysers & Dahmen ⁺ 00b] / i6	two-sided tangent distance, virtual data	2.4
[Keysers & Gollan ⁺ 04b] / i6	1-NN, zero-order deformation model	2.4
[Dong & Krzyzak ⁺ 02b] / CENPARMI	preprocessing, virtual support vector m.	2.2
[Keysers & Deselaers ⁺ 04] / i6	deformation, Hungarian matching	2.2
[Keysers & Paredes ⁺ 02] / i6+ITI	tangent distance + local patches	2.0
[Keysers & Gollan ⁺ 04b] / i6	3-NN, pseudo 2-D deformation model	1.9

itself. Nevertheless a comparison of ‘best performing’ algorithms may lead to valid conclusions, especially if these perform well on several different tasks. This is summarized by the following quotation referring to the USPS data: “Although there is an official test set of data to be used to evaluate different methods, it can be overused. For example, a group may attempt tens or hundreds of different configurations, but only report the results of the best. These caveats hold for any technique with tunable parameters, but are especially pertinent for neural networks which have many.” [Hastie & Simard 98] Note that this disadvantage

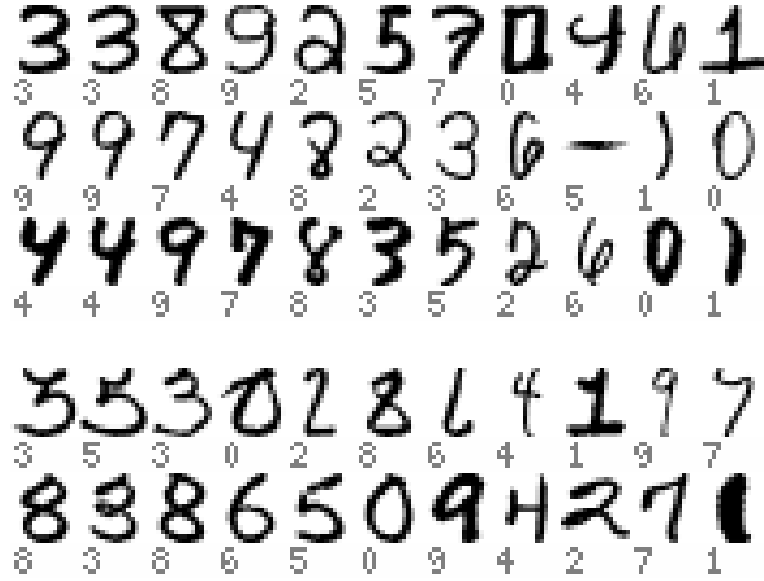


Figure 3.3: Examples of nearest neighbor recognition on USPS (with class labels), first image: test pattern, following images: best references from each class in order of increasing distance to the test pattern. Top rows: correct classification. Bottom rows: incorrect classification.

holds for almost all data sets available for image object recognition.

Another disadvantage of the USPS data is that the test set only contains 2,007 samples. This means that differences in error rate alone are unlikely to be statistically significant. If the numbers of the misclassified samples are known for two classifiers, a bootstrap analysis can determine if an improvement is statistically significant even if the error rate alone does not allow this. On the other hand the small size of the USPS data set can also be an advantage, because new algorithms can easily be tested on (and tuned to) the data and then applied to other data sets as e.g. the MNIST task.

One strong advantage of the USPS task is the availability of many recognition results reported by international research groups, allowing a meaningful comparison of results. Results for different algorithms are listed in Table 3.2. We can observe that the USPS data set continues to be used in a number of publications even in recent years.

A slightly different setup of the USPS data exists, sometimes called USPS+ [DeCoste & Schölkopf 02]. Here, 2,549 additional images of machine printed digits were added to the training set [LeCun & Boser⁺ 90]. This makes a comparison of the results difficult, because in almost all pattern recognition tasks the performance (in terms of error) can be improved by adding more training data (cp. Section 4.7.4).

Figure 3.3 shows some examples of a basic 1-NN classifier using the Euclidean distance on the USPS database. The 1-NN was used as a baseline result for most experiments here and achieves an error rate of 5.6% on the data. The figure demonstrates the determination of (dis-)similarity of the Euclidean distance for the appearance-based approach taken. For instance in the examples of correct classification it can be noticed that similar line thickness seems a very strong factor for overall similarity. This is a result confirmed by the investigations on tangent distance, since best improvements could be obtained using the tangent

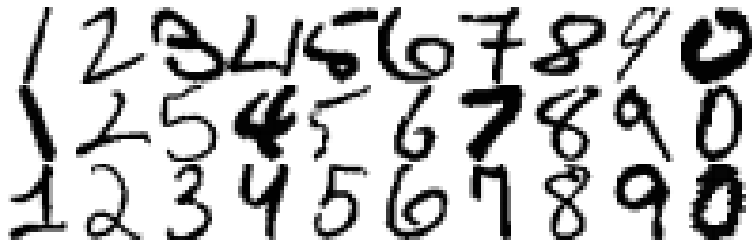


Figure 3.4: Some example images from the MNIST data set.

for line thickness. Also, in the correct classifications the best matching references are very similar to the observation images, which underlines the necessity of large training data sets to be able to recognize varying input patterns.

3.1.2 The MNIST task

The modified NIST (National Institute of Standards and Technology) handwritten digit database (MNIST, [LeCun & Bottou⁺ 98]) is very similar to the USPS database in its structure. The main differences are that the images are not normalized and that the corpus is much larger. It contains 60,000 images in the training set and 10,000 patterns in the test set of size 28×28 pixels with 256 graylevels. The data set is available online⁴. Some examples from the MNIST corpus are shown in Figure 3.4, which illustrate the effects of (missing) normalization when compared to Figure 3.1.

The preprocessing of the images is described as follows in [LeCun & Bottou⁺ 98]: “The original black and white (bilevel) images were size normalized to fit in a 20×20 pixel box while preserving their aspect ratio. The resulting images contain gray levels as result of the antialiasing (image interpolation) technique used by the normalization algorithm. [...] the images were centered in a 28×28 image by computing the center of mass of the pixels and translating the image so as to position this point at the center of the 28×28 field.”

The task is generally considered to be an easier recognition task than the USPS task for two reasons. First, the human error rate is estimated to be only 0.2%, although it has not been determined for the whole test set [Simard & Le Cun⁺ 93]. Second, the (almost ten times) larger training set allows machine learning algorithms to generalize better. With respect to the connection between training set size and classification performance for OCR tasks it is argued in [Smith & Bourgoïn⁺ 94] that increasing the training set size by a factor of ten cuts the error rate approximately to half the original figure.

The arguments given for the USPS data concerning the absence of a development test set and the availability of research results from other groups are equally true for the MNIST database.

Table 3.3 gives an overview of the error rates reported in other publications for the MNIST data. You can observe that the error rates are smaller by a factor of two to five than the corresponding error rates for the USPS task. The best systems will be described in more detail in the following Section 3.1.3. By best systems we refer to the results marked with a * in Table 3.3, although another very good result is reported in [Dong & Krzyzak⁺ 05],

⁴<http://www.research.att.com/~yann/ocr/mnist/>

Table 3.3: Error rates for the MNIST task in %. The systems marked with * are those we refer to as the ‘four best systems’ because they represent a variety of established approaches even though there are other methods in the same range of error rates (see text).

reference / group	method	ER
[Simard & Le Cun ⁺ 93] / AT&T	human performance	0.2
	Euclidean nearest neighbor	3.5
[Marée & Geurts ⁺ 04] / U Liège	decision trees + sub-windows	2.63
this work / i6	Gabor features + maximum entropy	2.5
[LeCun & Bottou ⁺ 98] / AT&T	deslant, Euclidean 3-NN	2.4
[Matsumoto & Uchida ⁺ 04] / Kyushu U	elastic matching	2.10
[Keysers & Dahmen ⁺ 00b] / i6	one-sided tangent distance	1.9
[Bottou & Cortes ⁺ 94] / AT&T	neural net LeNet1	1.7
[Mayraz & Hinton 02] / U Coll. London	products of experts	1.7
[Dahmen 01] / i6	Gaussian mixtures	1.7
[Milgram & Sabourin ⁺ 05] / U Québec	hyper-planes + support vector m.	1.5
[Schölkopf 97] / TU Berlin	support vector machine	1.4
[Bottou & Cortes ⁺ 94] / AT&T	neural net LeNet4	1.1
[Simard & Le Cun ⁺ 93] / AT&T	tangent distance	1.1
[Keysers & Dahmen ⁺ 00b] / i6	two-sided tangent d., virt. data	1.0
[Dong & Krzyzak ⁺ 02a] / CENPARMI	local learning	0.99
[Schölkopf & Simard ⁺ 98] / MPI & AT&T	virtual SVM	0.8
[LeCun & Bottou ⁺ 98] / AT&T	distortions, neural net LeNet5	0.82
[LeCun & Bottou ⁺ 98] / AT&T	distortions, boosted LeNet4	0.7
[Teow & Loe 00] / U Singapore	bio-inspired features + SVM	0.72
[DeCoste & Schölkopf 02] / Caltech & MPI	virtual SVM (box jitter)	0.68
[Belongie & Malik ⁺ 02]* / Berkeley	shape context matching	0.63
[Dong & Krzyzak ⁺ 05] / CENPARMI	support vector machine	0.60
[Teow & Loe 02] / U Singapore	deslant, biology-inspired features	0.59
[Athistos & Alon ⁺ 05] / Boston U	cascaded shape context	0.58
[DeCoste & Schölkopf 02]* / Caltech & MPI	virtual SVM (box jitter + shift)	0.56
[Athistos & Alon ⁺ 05] / Boston U	shape context matching	0.54
[Keysers & Gollan ⁺ 04b]* / i6	deformation model (IDM)	0.54
this work / i6	deformation model (P2DHMDM)	0.52
[Liu & Nakashima ⁺ 03] / Hitachi	preprocessing, support vector m.	0.42
[Simard & Steinkraus ⁺ 03]* / Microsoft	neural net + virtual data	0.42
this work / –	hyp. combination of four systems (*)	0.35

because these four systems best represent a range of different approaches. The methods based on deformable models will be described in Chapter 6.

Note that Dong gives lower error rates than in [Dong & Krzyzak⁺ 05] of 0.38 to 0.44 percent on his web page (accessed April 2005), but it remains unclear how these error rates were obtained and if possibly these low error rates are due to the effect of ‘training on the testing data’. Also, [Teow & Loe 02] try a variety of SVMs and networks which yield error rates ranging from 0.59 percent to 0.81 percent. To illustrate another possibility to observe the effect of ‘training on the testing data’ consider for example to use a support vector machine for each pair of classes based on the IDM distances as used in [Keysers & Gollan⁺ 04b].



Figure 3.5: Some difficult examples from the MNIST test set along with their target labels. At least one of the four best systems (cp. Table 3.3) misclassifies these images. The framed examples are misclassified by all four systems.

Then, the best parameters are chosen for each pair of classes. Using voting to arrive at the final decision leads to an error rate of 0.42% on the MNIST data [Bernard Haasdonk, personal communication, 2004]. We feel that this procedure involves looking at the test data too many times before computing the final result and therefore it is a very optimistic estimate. In contrast, the IDM as used in [Keysers & Gollan⁺ 04b] was not optimized for the MNIST task. The same parameters that proved to work well on the USPS data were chosen and only the 3-NN was preferred over the 1-NN by looking at the test results.

Figure 3.5 shows some difficult examples from the MNIST test set. At least one of the four best systems misclassifies each sample. Those samples that are misclassified by

all four systems are marked by a surrounding frame. This presentation is possible because both in [DeCoste & Schölkopf 02] and in [Belongie & Malik⁺ 02] the authors present the set of samples misclassified by their systems. Furthermore, Patrice Simard kindly provided the classification results of his system as described in [Simard & Steinkraus⁺ 03] for all test data. The availability of these results also makes it possible to determine the error rate of a hypothetical system that combines these four best systems as described in the following Section 3.1.3.

Some of the images in Figure 3.5 are a good illustration of the inherent class overlap that exists for this problem: some instances of e.g. ‘3’ vs. ‘5’, ‘4’ vs. ‘9’, and ‘8’ vs. ‘9’ are not distinguishable by taking into account the observed image only. This suggests that we are dealing with a problem with non-zero Bayes error rate. Further improvements in the error rate on this data set might therefore be problematic. For example, consider a classifier that classifies the second framed image as a ‘9’: despite the fact that this classifier would not make an error with this decision according to the class labels, we might prefer a classifier that classifies the image as a ‘4’.

3.1.3 MNIST: state-of-the-art in handwritten digit recognition

In this section we briefly describe the four best systems for handwritten digit recognition as measured by their MNIST data performance. We also discuss the statistical significance of their results and present a simple classifier combination of these four methods that achieves a (hypothetical) error rate of 0.35%.

Shape context matching [Belongie & Malik⁺ 02] present the shape context matching approach. The method proceeds by first extracting contour points of the images. In the case of handwritten character images the resulting contour points trace both sides of the pen strokes the character is composed of. Then, at each contour point a local descriptor of the shape as represented by the contour points is extracted. This local descriptor is called a shape context and is a histogram that counts how many contour points are present in the surrounding of the central point. This histogram has a finer resolution at points close to the central point and a coarser for regions farther away, which is achieved using a log-polar representation.

The classification is then done by using a nearest neighbor classifier (although the authors chose to use only one third of the training data for the MNIST task). The distance within the classifier is determined using an iterative matching based on the shape context descriptors and two-dimensional deformation. The shape context of training and test image are assigned to each other by using the Hungarian algorithm (as also in this work; cp. Section 6.7) on a bipartite graph representation with edge weights according to the similarity of the shape context descriptors. This assignment is then used to estimate a two-dimensional spline transformation best matching the two images. The images are transformed accordingly and the whole process (including extraction of shape contexts) is iterated until a stopping criterion is reached. The resulting distance is used in the classifier.

Very recently, [Athistos & Alon⁺ 05] discuss a cascading technique to speed up the slow nearest neighbor matching by “two to three orders of magnitude”. While the result that this discussion is based on only used the first 20,000 training samples for reasons of efficiency and resulted in an error rate of 0.63% [Belongie & Malik⁺ 01],

[Athistos & Alon⁺ 05] report an error rate of 0.54% for the full training set and 0.58% for the cascaded classifier that uses only about 300 distance calculations per test.

Invariant support vector machine [DeCoste & Schölkopf 02] present a support vector machine (SVM) that is especially suited for handwritten digit recognition by incorporating prior knowledge about the task. This is achieved by using virtual data or a special kernel function within the SVM. The special kernel function applies several transformations to the compared images that leave the class identity unchanged and return the kernel function of the appropriate pair of transformed images. This method is referred to as kernel jittering. The second uses so-called virtual support vectors. This approach consists of first training a support vector machine. Now, experimental results suggest that the set of support vectors contains sufficient information about the recognition problem and can therefore be considered a condensed representation of the training data for discrimination purposes. The method proceeds to create transformed versions of the support vectors, which are the virtual support vectors. In the experiments leading to the error rate of 0.56% the transformations used were image shifts within the eight-neighborhood plus horizontal and vertical shifts of two pixels, thus resulting in $9 + 4 = 13$ virtual support vectors for each original support vector. (This experiment also used the deslanted version of the MNIST data [LeCun & Bottou⁺ 98].) On this new set of virtual support vectors, another support vector machine was trained and evaluated on the test set.

Zero order discrete image matching with local contexts [Keysers & Gollan⁺ 04b] present deformable models for handwritten character recognition as they are discussed in more detail in Chapter 6. It is shown that a simple zero-order matching approach called image distortion model (IDM) can lead to very competitive results if the local context of each pixel is considered in the distortion. The IDM allows to choose for each pixel of the test image the best fitting counterpart of the reference image within a suitable corresponding range. The context that each pixel is represented by is chosen to be a 3×3 local window of the image gradient. The distance as determined by the best match between two images is then used within a three-nearest-neighbor classifier. For more details the reader is referred to Chapter 6.

Convolutional neural net and distorted virtual data [Simard & Steinkraus⁺ 03] presents a large convolutional neural network of about 3,000 nodes in five layers that is especially designed for handwritten character classification. The new concept in the approach is to present a new set of virtual training images to the learning algorithm of the neural net in each iteration of the training. The virtual training set is constructed from the given training data by applying a separate two-dimensional random displacement field that is smoothed with a Gaussian filter to each of the images. This makes it possible to generate a very large amount of virtual data in the order of 1,000 virtual samples for each original element of the training data set. The data is generated on the fly in each training iteration and therefore does not have to be saved, which avoids the problems with data handling. Apart from the generation of virtual examples there is another point where prior knowledge about the task comes into play, namely the use of a *convolutional* neural net. This architecture, which is described in greater detail in [LeCun & Bottou⁺ 98], contains prior knowledge in that it uses tying of weights within the neural net to extract low-level features from the input that are invariant

Table 3.4: Probabilities of improvement for all pairs of the four best classifiers according to a bootstrap analysis. Probabilities in boldface show **significant** improvements with respect to the 5% level. This table is read to be as follows: the classifier given for each row improves over the classifiers given in the columns with the stated probability (e.g. the probability of improvement for SVM over SC is 0.60). The second table shows the difference in error rates for comparison. SC: shape context matching; SVM: invariant support vector machine; IDM: image distortion model; CNN: convolutional neural net with distortions;

probability of improvement					difference in error rate				
	SC	SVM	IDM	CNN		SC	SVM	IDM	CNN
SC	—				SC	—			
SVM	0.60	—			SVM	0.07	—		
IDM	0.85	0.58	—		IDM	0.09	0.02	—	
CNN	0.99	0.96	0.92	—	CNN	0.21	0.14	0.12	—

with respect to the position within the image, and only in later layers of the neural net the position information is used.

We can observe that all four methods take special measures to deal with the image variability present in the images, using virtual data and image matching methods. At the same time the concrete classification algorithm seems to play a somewhat smaller role in the performance as nearest neighbor classifiers, support vector machines, and neural networks all perform very well. Only a slight advantage of the neural net can be seen in the possibility to use very large amounts of virtual data in training because the training proceeds in several iterations, which need not use the same data but can use distorted samples of the images instead.

As mentioned in the previous section, we can perform a more detailed analysis of the results of these four methods because both in [DeCoste & Schölkopf 02] and in [Belongie & Malik⁺ 02] the authors present the set of samples misclassified by their systems. Furthermore, Patrice Simard kindly provided the classification results of his system as described in [Simard & Steinkraus⁺ 03] for the test data.

The more detailed analysis performed here is an estimation of the probability that a classifier performs generally better than a second classifier (probability of improvement) by using the decisions of the two classifiers on the same test samples. We estimate this probability by drawing a large number of bootstrap samples from the test data set and observing the relative performance of the two classifiers on these resampled test sets [Bisani & Ney 04]. This estimation tells us more than just using a comparison based on the individual error rates alone. For example, we will intuitively be more inclined to believe that the first classifier is better if it leads to better classifications on 2% of the test data and to the same results on the remaining 98% than if the first classifier performs better on 50% of the test data but worse on 48% of the data. (For an interesting discussion of significance in the context of comparisons of machine learning algorithms, see [Salzberg 97].) Table 3.4 shows the probabilities of improvement based on this technique for the four methods described above along with the differences in error rate. [LeCun & Bottou⁺ 98] states that improvements of more than 0.1% in the error rate may be considered significant. The analysis performed here allows a more detailed assessment of the significance of improvements.

Table 3.5: Results for the UCI task, error rates [%].

reference / group	method	ER
	Euclidean 1-NN	2.0
[Kim & Kim ⁺ 02] / Pohang U, S. Korea	independent comp. analysis	5.8
[Alpaydi & Kaynak 98] / Bogazici U, Istanbul	cascading classifier	4.7
[Alpaydi & Kaynak 98] / Bogazici U, Istanbul	Euclidean 3-NN	2.2
[Kim & Kim ⁺ 02] / Pohang U, S. Korea	PCA 12 comp.	2.2
[Kim & Kim ⁺ 02] / Pohang U, S. Korea	PCA mixture, 5 mix., 13 comp.	2.1
[Kim & Kim ⁺ 02] / Pohang U, S. Korea	PCA mixture, 2 mix., 12 comp.	1.5
this work / i6	3-NN, P2DHMM, local context	1.1
[Keysers & Gollan ⁺ 04b] / i6	3-NN, P2DHMDM, local context	0.8
[Keysers & Gollan ⁺ 04b] / i6	3-NN, IDM, local context	0.8

Figure 3.5 shows all the errors made by one of the four classifiers. Only eight samples are classified incorrectly by all four systems, which suggests the use of classifier combination to further reduce the error rate. The availability of the results of the other classifiers makes it possible to determine this error rate of a simple hypothetical combined system. We are somewhat restricted for the choice of combination scheme, though, because for two of classifiers we only know if the result was correct or not. If we use a simple majority vote combination based on the four classifiers, where the neural net classifier is used for tie-breaking (because it has the best single error rate), the resulting error rate of the combination is 0.35%.

3.1.4 The UCI task

The data of the UCI task was obtained from the the University of California, Irvine, (UCI) Repository of Machine Learning Databases⁵ [Merz & Murphy⁺ 97]. The data set contains handwritten digits of size 8×8 pixels with 17 gray levels. It is separated into a training set of 3,823 images and a test set comprising 1,797 images. Its construction is described in more detail in [Alpaydi & Kaynak 98]. The images were binary images of size 32×32, where each 4×4 block was then converted to an integer in the range of 0 to 16, resulting in the final images of size 8×8. Table 3.1 shows some example images from the UCI task and Table 3.5 shows a summary of the available results.

3.1.5 The MCEDAR task

The modified CEDAR (MCEDAR) data set is based on the data published by the Center of Excellence for Document Analysis and Recognition at the State University of New York at Buffalo (CEDAR) in form of a CD-ROM with the title “CEDAR CDROM Image Database 1: USPS Office of Advanced Technology Database of Handwritten Cities, States, ZIP Codes, Digits, and Alphabetic Characters”. The data set contains images of handwritten digits with a resolution of 8×8 pixels. There are 11,000 training images and 2,711 images in the test set.

We call the data modified CEDAR data, because the data is based on the subsets chosen in [Hinton & Dayan⁺ 97] and also the preprocessing performed by the authors: “The binary

⁵<ftp://ftp.ics.uci.edu/pub/machine-learning-databases/optdigits>

Table 3.6: Error rates [%] for the MCEDAR data set.

reference / group	method	ER
	Euclidean 1-NN	5.7
[Hinton & Dayan ⁺ 97] / Toronto	PCA	4.9
[Bishop & Winn 00] / Microsoft	Bayesian PCA	4.8
[Hinton & Dayan ⁺ 97] / Toronto	factor analysis	4.7
[Tipping & Bishop 99] / Microsoft	probabilistic PCA	4.6
[Deselaers (personal comm.) 02] / i6	local patches	4.3
[Keysers & Gollan ⁺ 04b] / i6	3-NN, IDM, local context	3.5
[Keysers & Gollan ⁺ 04b] / i6	3-NN, P2DHMDM, local context	3.3

Table 3.7: Error rates [%] for the CEDAR ‘goodbs’ data set.

reference / group	method	ER
[Revow & Williams ⁺ 96] / U Toronto	Euclidean 2-NN, 16×16	4.7
[Cai & Liu 99] / U Melbourne	HMM on contours	4.5
[Cai & Liu 99] / U Melbourne	HMM on contours, structure matching	3.8
[Revow & Williams ⁺ 96] / U Toronto	generative spline models	3.4
[Breuel 93] / IDIAP	bounded error matching	3.3
[Revow & Williams ⁺ 96] / U Toronto	generative spline mixture models	3.1

images in the data set are of varying sizes, so we first scaled them to lie on an 8×8 pixel grid and then smoothed with a Gaussian filter with a standard deviation of half a pixel.” [Hinton & Dayan⁺ 97] We used exactly the same data as Hinton, Dayan and Revow, which was also used by Michael Tipping (whom we would like to thank for providing the modified data) in [Tipping & Bishop 99]. Table 3.6 gives an overview of the error rates obtained on these data using various methods. Table 3.1 shows some example images.

Note that [Cai & Liu 99] uses a different training data set with 18,468 images and the unprocessed original images (bindigis/bs dataset of the CEDAR CD-ROM, which is otherwise the same test data as in the MCEDAR case). The preprocessing consists of region connection, slant correction, and outer contour extraction, here. [Breuel 93] used a total of 35,500 training images, including some NIST data. Also, [Revow & Williams⁺ 96] uses the unprocessed original data, but the same training set of 11,000 images, also unprocessed. Table 3.7 gives an overview of the error rates obtained on these data using various methods. We can observe that the error rates are much lower on the original data which was not used in this work, most clearly documented by the nearest-neighbor error rate, which is one percent absolute smaller for the unprocessed data. This difference is most probably due to the small image size of 8×8 pixels used in the MCEDAR data set.

3.1.6 The ETL6A task

The ETL6A task is named after the Electrotechnical Laboratory (ETL), Japan, which is part of the National Institute of Advanced Industrial Science and Technology (AIST). The complete ETL database contains about 1.2 million images of handwritten and machine printed characters, which are Japanese, Chinese, and Latin characters, and Arabic numbers. The database is available for research purposes⁶. From the web page (accessed March 2005)

⁶<http://www.is.aist.go.jp/etlcdb>

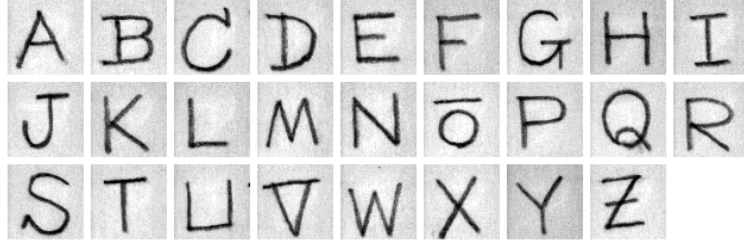


Figure 3.6: One example of each class of the ETL6A data.

Table 3.8: Results for ETL6A, error rates [%].

reference / group	method	error rate
	Euclidean, 1-NN	4.5
[Uchida & Sakoe 03b] / Kyushu U	preprocessing, Euclidean, 1-NN	1.9
[Uchida & Ronee ⁺ 02] / Kyushu U	eigen-deformations	1.1
[Uchida & Sakoe 03b] / Kyushu U	piecewise linear 2DW	0.9
[Uchida & Sakoe 02] / Kyushu U	deformation model	0.9
[Uchida & Sakoe 03b] / Kyushu U	eigen-deformations	0.8
[Uchida & Sakoe 02] / Kyushu U	deformation model, eigen-deformations	0.6
[Uchida & Sakoe 03a] / Kyushu U	eigen-deformations	0.5
[Keysers & Gollan ⁺ 04b] / i6	3-NN, IDM, local context	0.5

we quote the following description: “ ‘ETL character databases’ were collected as the common data to make it possible to compare the performance of off-line character recognition algorithms. Character images of the databases were got by observing OCR sheets or Kanji printed sheets with a scanner. All databases ETL1 - ETL9 are gray-valued image data.”

The ETL6A data is a subset of ETL6 and contains only the Latin characters. Examples are shown in Figure 3.6. The data set consists of 35,958 images of the 26 classes representing the uppercase Latin characters from ‘A’ to ‘Z’. The images are of size 64×63 pixels with 16 gray values. For our experiments, the images were scaled to 16×16 pixels using bilinear interpolation. Following [Uchida & Sakoe 03b], we use the first 600 images of each class as training data and the next 500 images as test data. Thus the size of the complete training set is 15,600 and the size of the test set is 13,000. Table 3.8 shows the available results for the ETL6A data.

All reference results available for this data set are due to the work of S. Uchida and colleagues. The best result so far was achieved using a method that the authors call ‘eigen-deformations’ [Uchida & Sakoe 03a]. Their work on various image deformation models was a basis and inspiration for many of the algorithms investigated in this thesis (cp. Chapter 6). Especially the two-dimensional warping algorithm is directly based on their work [Uchida & Sakoe 98].

3.2 Medical images

Another large area of image classification research is the medical domain. Also here, digital cameras and sensors and therefore digital image data become more and more ubiquitous, including for example microscope imagery. Furthermore, directly digital imaging modalities



Figure 3.7: One image from each of the six IRMA-1,617 classes: ‘abdomen’, ‘skull’, ‘chest’, ‘limbs’, ‘breast’, and ‘spine’.

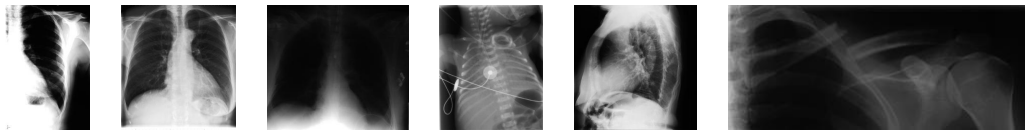


Figure 3.8: Several images from class ‘chest’ from the IRMA-1,617 database.

in radiology are used more and more widely. In several application areas, it is necessary to perform a classification of the taken images into one of several classes as an intermediate step or as a result. In the following sections we will introduce data sets of medical images that were used in experiments within this work.

3.2.1 The IRMA task

The IRMA database contains medical image data from the IRMA project (Image Retrieval in Medical Applications⁷) of the RWTH Aachen University. For an overview of the complete project, of which the classification stage discussed in this work is only a small fraction, see e.g. [Lehmann & Güld⁺ 04].

The database has evolved over time and now contains 10,000 images that are labeled with a detailed code that specifies body region, image modality, biosystem imaged, and imaging orientation [Lehmann & Schubert⁺ 03]. The first version of the database was available in 1999 and contained 1,617 medical radiographs [Dahmen & Theiner⁺ 00]. Most experiments on medical image categorization that are relevant to this work were performed with these data, which are associated with one of six class labels. Therefore, whenever the ‘IRMA database’ is mentioned without further specification within this document it refers to this first version containing 1,617 medical radiographs of six classes. For the first step in the retrieval process, i.e. classification or categorization, only six anatomic regions were distinguished, although more information for each image was available. Following this IRMA-1,617 data an additional test set of 332 images was made available, followed by an intermediate data set of about 4,000 images. This intermediate data set was then replaced by the final version in the beginning of 2005, which now contains the mentioned 10,000 images labeled with a detailed code. This final version of the IRMA database has also been publicly available since the beginning of 2005 because it was used as a part of the 2005 ImageCLEF workshop for the evaluation of image retrieval systems.⁸

Because many experiments within this work refer to the first IRMA database of 1,617 images, we will describe this database in more detail here. The radiograph images belong to

⁷<http://www.irma-project.org>

⁸<http://ir.shef.ac.uk/imageclef2005/>

one of the six classes abdomen, breast, chest, limbs, skull, and spine. The images originated from daily routine, are anonymized and secondary digital, that is they have been scanned from conventional film-based radiographs. All images were scanned using 256 gray levels, with the image sizes ranging from 142×233 pixels (e.g. a radiograph of a single finger) to $4,928 \times 4,008$ pixels (e.g. a chest radiograph). In the experiments, the images were adjusted to span the whole gray value range after a possible scaling. The distribution of the images reflects the distribution in the Department of Diagnostic Radiology of the University Clinic of the RWTH Aachen University and were labeled with one of the six classes by an expert. The quality of radiographs varies considerably and there is a great within-category variability (as caused by different doses of X-rays, varying orientations, images with and without pathologies or contrast agents, changing scriber position, etc.). Furthermore, there is a strong visual similarity between many images of the classes abdomen and spine. Therefore, the classification problem can be considered hard. The corpus consists of 110 abdomen, 706 limbs, 103 breast, 110 skull, 410 chest, and 178 spine radiographs, summing up to the total of 1,617 images. Furthermore, a smaller set of 332 images exists for testing purposes.

Figure 3.7 shows example images from the database which represent the different classes. The database contains a wide variation of images: Figure 3.8 shows different images of the class ‘chest’, giving an idea of the high intra-class variability. The original images are of varying sizes but were scaled to a common height of 32 pixels for the classification experiments, keeping the aspect ratio (denoted by ‘ $X \times 32$ ’). To be able to compare the results with some older results, for some experiments we also used a version that consists of images scaled to 32×32 pixels. This rescaling did not affect the recognition rate significantly [Dahmen & Theiner⁺ 00]. In some experiments the aspect ratio of the original images was used as an additional feature because this number supplies additional information about the image class.

Because there are only 1,617 images available, a leaving-one-out approach is used, thus the database serves as training and test set, where each image is classified while using the remaining 1,616 as training set. After parameter adjustment, a classifier can be evaluated on a new set of 332 additional radiographs, although most error rates for comparison are available for the original leaving-one-out approach.

For the IRMA-1,617 database, reference results for comparison are available only from within the IRMA project so far, since it is not in widespread use but originated from the project at the RWTH Aachen University. Because the database has been made available in the beginning of 2005, more reference results of competing approaches will probably be available in the future for the IRMA-10,000 data. First results for the IRMA-10,000 data are described in Section 3.2.1. Table 3.9 shows a summary of the results available for the IRMA-1,617 data.

The use of conventional approaches to segment these data usually fails which emphasizes the effectiveness of the appearance-based approach here. E.g. Bredno and colleagues applied an active shape approach to the categorization problem [Bredno & Brandt⁺ 00]. For form based image retrieval they extracted the outline of the shapes using balloon models and extracted invariant signatures from the outlines for classification. Using a 1-NN classifier the best error rate for leaving-one-out of 51.1% was achieved using invariant moments. On the 496 images where the outline detection was subjectively successful the error rate achieved was 34.9%.

The use of cooccurrence matrices [Haralick & Shanmugam⁺ 73] is often considered to be helpful for content-based medical image retrieval. However, experiments in this setup

Table 3.9: Leaving-one-out error rates for the IRMA-1,617 task [%]. Error rates for experiments performed at RWTH-i6 are given for the images scaled to a common height of 32 pixels, unless marked with a *, which indicates the data were scaled to 32×32 pixels.

reference	method	ER
–	Euclidean 1-NN *	18.2
–	Euclidean 1-NN	15.8
[Bredno & Brandt ⁺ 00]	active shape models	51.1
–	C4.5 decision tree	31.7
[Dahmen 01]	cooccurrence matrix features	29.0
[Deselaers 03]	1-NN, invariant feature histogram	22.6
[Deselaers 03]	1-NN, Tamura feature histogram	19.3
[Keysers 00]	kernel densities *	16.4
–	1-NN, normalized cross correlation *	14.8
[Keysers 00]	kernel densities, thresholding *	14.2
–	1-NN, normalized cross correlation, shifts *	13.3
[Paredes & Keysers ⁺ 02]	local patches	13.0
[Paredes & Keysers ⁺ 02]	local patches, thresholding	9.7
[Paredes & Keysers ⁺ 02]	local patches, histograms	9.3
[Keysers 00]	kernel densities, thresh., IDM *	9.0
[Keysers 00]	kernel densities, thresh., tangent distance, IDM *	8.0
[Kölsch & Keysers ⁺ 04]	local patches, tangent distance, kernel dens.	7.4
[Keysers & Gollan ⁺ 04a]	1-NN, IDM, local context, thresholding	6.6
[Keysers & Gollan ⁺ 04a]	1-NN, P2DHMM, local context, thresh.	5.7
[Keysers & Gollan ⁺ 04a]	1-NN, P2DHMDM, local context, thresh.	5.3

do not support this thesis [Dahmen 01]. In two experiments, global cooccurrence matrices for feature analysis within a synergetic classifier and within a kernel density-based classifier were used. In both cases, it was not possible to obtain classification error rates below 29%.

Figure 3.9 shows examples of a basic 1-NN classifier using Euclidean distance on the IRMA database with 32×32 sized images. With this basic setting an error rate of 18.2% is obtained, which can be substantially reduced using the models of variability discussed in this work.

The IRMA approach

To present the rationale behind the IRMA approach, we give a short introduction to the IRMA approach to medical image retrieval here. The importance of digital image retrieval techniques increases in the emerging fields of medical imaging and picture archiving and communication systems. Up to now, textual index entries are necessary to retrieve medical images from a hospital archive, even if the archive is digital imaging and communications in medicine (DICOM)-compliant. Currently, a lot of research is done in the field of image retrieval, but the majority of today’s content-based image retrieval (CBIR) approaches are intended for browsing large databases of arbitrary content, e.g. collected from the World Wide Web. Usually, the features used for indexing characterize the entire image rather than image regions or objects and one of the most effective features of such systems is color [Deselaers & Keysers⁺ 04b]. Unfortunately, color-based features are not suitable for

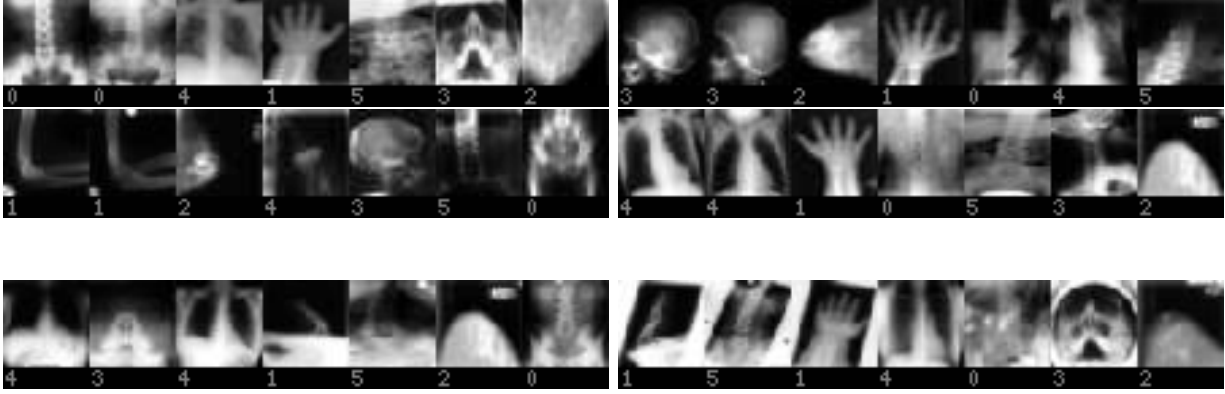


Figure 3.9: Examples of nearest neighbor recognition on the IRMA database using 32×32 images. First image: test pattern, following: best references from each class in descending order. Top: correct classification, bottom: incorrect classification. (class numbers: 0=‘abdomen’, 1=‘limbs’, 2=‘breast’, 3=‘skull’, 4=‘chest’, 5=‘spine’)

the majority of medical images, which are usually grayscale images. Resulting from the variety of images, common CBIR systems have only a rudimentary understanding of image content, with little or no distinction between important and negligible features or between different anatomical or biological objects in the image. But queries of diagnostic relevance include searching for organs, their relative locations, and other distinct features such as morphological appearances. Therefore, common CBIR systems cannot guarantee a meaningful query completion when used in a medical context. In contrast, the IRMA system is being developed for use in daily clinical routine [Lehmann & Wein⁺ 00].

The IRMA concept is based on a separation of the following seven steps to enable complex image content understanding: categorization of the entire image, registration with respect to prototypes, extraction and query-dependent selection of local features, hierarchical blob representation including object identification and finally, image retrieval. Various imaging techniques require adapted image processing methods. For example, ultrasonic images of vessels must be processed in a different manner than skeletal radiographs. Thus, if a radiologist is searching the database for all radiographs showing a pulmonal tumor, the IRMA system only processes radiographs which have a sufficiently high posterior probability for the class ‘chest’. Therefore, the categorization step — which is the only one discussed in this work — not only reduces the computational complexity of an IRMA query, it will also reduce the ‘false-alarm’-rate of the system, improving its precision. Based on global features, the IRMA approach distinguishes four major categories: image modality (physical), body orientation (technical), anatomic region (anatomical), and biological system (functional). These categories build subclasses resulting in hierarchically structured categories [Lehmann & Wein⁺ 00].

Modern modalities allow submission of textual information about the examination. However, medical staff often does not enter appropriate or sufficient data into the systems, as a study showed quantitatively (only one out of four examined modalities included the correct DICOM-header information and even in this case the information was incorrect in 15.5% of the examined cases [Kohnen & Schubert⁺ 01]). In a well-managed DICOM-compliant

picture archiving and communication system linked to a hospital information system, text-based retrieval will give excellent results. But in many cases automatic indexing by image content is still a necessary component to provide sufficient information, where these two methods should not be viewed as mutually exclusive but as synergetic tools.

Care must be taken with (possibly false) local decisions because of the interdependence of the steps. To improve the possibility of obtaining the overall best possible query result, it is necessary to work with a variety of different hypotheses throughout these steps. This holistic approach has shown superior performance over local decisions in applications such as speech recognition [Ney & Ortmanns 00]. An example of modeling of vague knowledge is the computation of posterior probabilities during the categorization step, which avoids the hard choice of a possibly false image category. Thus, it is not necessary to correct a false decision later, but instead the entire process works on multiple hypotheses at the same time. It is helpful to consider classification methods for CBIR because classification and image retrieval aim at similar objectives, which has been pointed out before (e.g. [Liu & Dellaert 98]).

The size of the used database with 10,000 images may seem small in comparison to large-scale databases with millions of entries that need to be handled in real-world medical image databases. Nevertheless, the IRMA database is the first database of this size containing medical images from daily routine that are labeled by an expert, allowing a detailed evaluation of the classification performance. On larger databases it will be necessary to reduce the computational load using known algorithms for the access of large databases.

ImageCLEF 2005 / IRMA

The IRMA database used in the ImageCLEF 2005 evaluation total consists of 10,000 images labeled with the detailed IRMA code. The data set is partitioned into 9,000 training images and 1,000 test images. For the ImageCLEF 2005 task, the images were subdivided into 57 classes. The images were scaled to a maximum width or height of 512 pixels and use 256 gray values.

In Table 3.10 an overview of the results of the 2005 ImageCLEF ‘Automatic Annotation Task’ is given. For each group, only the best and the worst result among the submissions is included. The complete table is available online⁹. In total, 26 groups registered for participation in the automatic annotation task. From these 26 groups, 12 groups submitted 41 runs, each group had at least two different submissions, the maximum number of submissions per group was 7. In the following, a very short description of the methods that were used by the groups given.

CEA: CEA, France, submitted three runs. In each run different feature vectors were used and classified using a k -Nearest Neighbor classifier (k was either 3 or 9). In one run the images were projected along horizontal and vertical axes to obtain a feature histogram. For the second run histograms of local edge pattern features and color features were created, and for the third run quantified colors were used.

Concordia U: The CINDI group from Concordia University in Montreal, Canada used multi-class SVMs (one-vs-one) and a 170 dimensional feature vector consisting of color moments, color histograms, cooccurrence texture features, shape moment, and edge histograms.

U Geneva: The medGIFT group from Geneva, Switzerland used various different settings for gray-levels and Gabor filters in their medGIFT image retrieval system.

⁹http://www-i6.informatik.rwth-aachen.de/deselaers/imageclef05_aat_results.html

Table 3.10: Overview of results achieved in the 2005 ImageCLEF ‘Automatic Annotation Task’ evaluation.

group		method or label	ER[%]
RWTH-i6	DE	IDM, local context	12.6
RWTH-MI	DE	IDM + Correlation + Tamura	13.3
RWTH-i6	DE	patches, discriminative training	13.9
U Liège	BE	patches, boosting	14.1
RWTH-MI	DE	IDM + Correlation + Tamura	14.6
U Liège	BE	patches, trees	14.7
U Geneva	CH	GIFT5NN-8g	20.6
Infocomm	SG	4-I2R-sg	20.6
U Madrid	ES	GIFT + decision table	21.4
NTU	TW	gray value block + nearest neighbor	21.7
Infocomm	SG	texture features + SVM	21.7
U Geneva	CH	medGIFT retrieval	22.1
U Madrid	ES	GIFT + nearest neighbor	22.3
NTU	TW	gray value blocks	22.5
NCTU	TW	SVM to learn image characteristics	24.7
–	–	32×32, 1-NN	36.8
CEA	FR	image projection, 3-NN	36.9
Mt.Holyoke	US	Gabor Energy features	37.8
Mt.Holyoke	US	Gabor Energy features	40.3
Concordia U	CA	SVM, mixed image feature vectors	43.3
CEA	FR	quantified colors, 9 nearest neighbor	46.0
U Montreal	CA	feature combinations	55.7
NCTU	TW	SVM to learn image characteristics	61.5
U Montreal	CA	texture features	73.3

Infocomm: The group from Infocomm Institute, Singapore, used three kinds of 16×16 low-resolution-map-features: initial gray values, anisotropy and contrast. To avoid over-fitting, for each of 57 classes a separate training set was selected and about 6,800 training images were chosen out of the given 9,000 images. Support Vector Machines with RBF kernels were used for classification.

U Madrid: The Miracle Group from UPM Madrid, Spain uses GIFT and a decision table majority classifier to calculate the relevance of each individual result. Additionally in one run a k -nearest neighbor classifier with $k = 8$ and attribute normalization is used.

U Montreal: The group from University of Montreal, Canada submitted seven runs, which differ in the features used. The participants estimated, which classes are best represented by which features and combined appropriate features.

Mt.Holyoke: For the submission from Mount Holyoke College, MA, USA, Gabor energy features were extracted from the images and two different cross-media relevance models were used to classify the data.

NCTU : The NCTU-DBLAB group from National Chiao Tung University, Taiwan, used a support vector machine (SVM) to learn image feature characteristics. Based on the SVM model, several image features were used to predict the class of the test images.

NTU: The Group from National Taiwan University used mean gray values of blocks as features and different classifiers for their submissions.

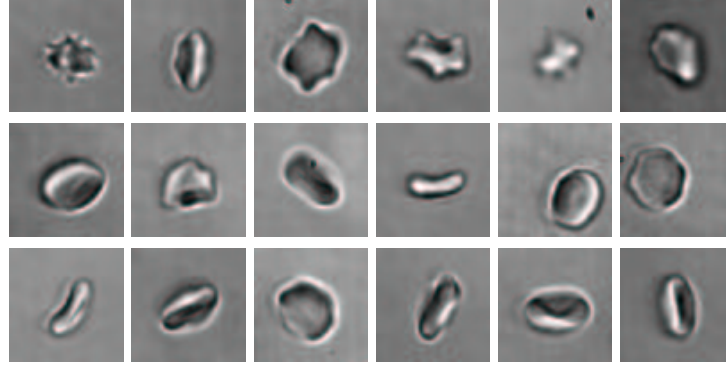


Figure 3.10: Images of red blood cells, top to bottom: stomatocytes, discocytes, echinocytes.

RWTH-i6: The Lehrstuhl für Informatik VI from RWTH Aachen University, Germany, had two submissions. One used the simple zero-order image distortion model taking into account local context. The other submission used a maximum entropy classifier and histograms of image patches as features.

RWTH-MI: The Medical Informatics group from RWTH Aachen University, Germany, used global texture features and two distance measures for down-scaled representations, which preserve spatial information and are robust with respect to global transformations like translation, intensity variations, and local deformations. The weighting parameters for combining the single classifiers were guessed for the first submission and trained on a random 8000-1000 partitioning of the training set for the second submission.

U Liège: The University of Liège method is based on random sub-windows and decision trees. During the training phase, a large number of multi-size sub-windows are randomly extracted from training images. Then, a decision tree model is automatically built (using Extra-Trees and/or Tree Boosting), based on size-normalized versions of the sub-windows, and operating directly on their pixel values. Recognition of a new image similarly entails the random extraction of sub-windows, the application of the model to these, and the aggregation of sub-window predictions.

3.2.2 The red blood cell task

The red blood cells (RBC) task is a database of 5,062 images that were labeled by an expert as either ‘stomatocyte’, ‘echinocyte’ or ‘discocyte’. Each cell is represented by a 128×128 pixels sized gray scale image. Examples are shown in Figure 3.10. The images were taken in a capillary where the RBC could show their native shapes without applied forces during sedimentation [Schönfeld & Grebe 89]. The dataset was not divided into a single training and test set. Instead, a cross-validation approach is applied in the experiments. The data are split into ten subsets. Each data set is then used for testing while the remaining nine sets are used for training, with the overall error rate being the mean over all subset error rates. Note that although all images are used as test and training images, the according training and test sets are strictly disjoint in all cases. For some experiments, the data was even used applying the leaving-one-out approach. I.e., when classifying an image we use the remaining 5,061 images as training data. This approach still strictly separates training and test data, but fully uses all available information. The latter approach was used for the experiments employing kernel density and nearest neighbor classifiers in this

Table 3.11: Results for the red blood cells corpus, error rates [%].

reference	method	ER
[Dahmen & Hektor ⁺ 00]	human performance	>20.0
[Dahmen & Hektor ⁺ 00]	Gaussian mixture densities	31.0
–	Euclidean 1-NN	21.4
[Keysers & Dahmen ⁺ 01a]	kernel densities	19.6
[Dahmen & Hektor ⁺ 00]	GMD, RST-invariant features	18.8
[Kölsch & Keysers ⁺ 04]	local patches, direct voting	17.7
[Kölsch & Keysers ⁺ 04]	local patches, KD	17.2
[Keysers & Dahmen ⁺ 01a]	KD, tangent distance, virtual data	16.3
[Dahmen & Hektor ⁺ 00]	GMD, RST-invariant features, LDA	15.3
[Kölsch & Keysers ⁺ 04]	local patches, tangent distance	13.5

work. [Dahmen & Hektor⁺ 00] preferred the ten-fold cross-validation instead due to the larger training requirements of a classifier based on Gaussian mixture densities and linear discriminant analysis.

As for the motivation of this RBC task, it should be noted that in standard tests, drugs that induce shape changes to red blood cells are often used to examine whether the cell membrane still acts in a well known way. This is done by comparing induced shape changes with the known behavior on drugs [Deuticke & Grebe⁺ 90]. This comparison is usually performed by a human expert and therefore time and cost consuming, which underlines the need for automatic classification. On the one hand, the high human error rate is caused by the hardness of the problem (especially the classes stomatocyte and discocyte are often hard to distinguish). On the other hand, manually classifying RBC is often done directly at the microscope, where decreasing alertness leads to a significant increase in mislabelings.

3.3 General images and complex scenes

3.3.1 The COIL-20 task

The Columbia University Object Image Library (COIL-20, [Murase & Nayar 95]) consists of images taken of 20 different 3D-objects viewed from 72 points of view taken at intervals of five degrees 3D-rotation¹⁰. Each image contains a single object (which is subject to different lighting conditions) placed in front of (almost) homogeneous black background and is given in 256 gray levels. There are 1,440 reference images of size 128×128 pixels available (called ‘processed’ data), as well as 360 test images of size 448×416 pixels (called ‘unprocessed’ data). The two corpora differ in the lighting conditions (because of the processing) and the size of the object in the image. Although the test images belong to only five classes, the problem is still treated as a 20-class problem in the experiments. To strictly separate training and test images, we use the odd angles of the ‘processed’ corpus for training and the even angles of the ‘unprocessed’ corpus for testing. This procedure ensures at least 5 degrees difference in 3D position and poses the additional difficulty of differing light conditions. Thus, a number of 720 reference images and 180 test images remains. Other authors use a splitting of the ‘processed’ corpus into train and test, but in this case even a Euclidean nearest neighbor classifier leads to a 0% error rate. Because of the fact that there are only

¹⁰<http://www.cs.columbia.edu/CAVE/research/softlib/coil-20.html>



Figure 3.11: One view of each of the 20 COIL-20 objects from the ‘processed’ corpus and one image from the ‘unprocessed’ corpus.

72 reference images available per class, a nearest neighbor-based classifier was used in the COIL-20 experiments throughout this work.

Example images taken from the COIL-20 training (processed) corpus and one image of the test (unprocessed) corpus are shown in Figure 3.11.

Concerning the state-of-the-art it should be noted that only few authors report error rates for the whole data set. In fact, most authors use COIL-20 in a modified version, for instance to investigate on the behavior of a recognition system in presence of inhomogeneous backgrounds and the like, because it is very easy to modify the COIL-20 data. Therefore, a direct comparison of different COIL-20 results is hard. Nevertheless it was chosen for some experiments on object localization and object recognition. Interesting publications using COIL-20 are for instance:

- In [Murase & Nayar 95], the authors implement a real-time segmentation-based recognition system for the COIL-20 data, reporting an error rate of 0% (using 720 unavailable test scenes which differ from the 360 mentioned above).
- In [Baker & Nayar 96], the authors present a recognition system optimized for fast recognition of COIL-20 objects. The experiments were conducted using only the 1,440 reference images (processed image set), which were split into two disjoint subsets. For this particular setting, the authors report an error rate of 0%.
- In [Pösl 98], the authors use a small subset of the available COIL-20 images for experiments dealing with inhomogeneous backgrounds and localization of known objects. In the experiments conducted, either the class of the object was known and the task is to detect it in the scene, or the position of the object is known and the according class label is to be determined. Furthermore, as only a subset of the images was used, no error rates for the complete COIL-20 data are given.
- The holistic model as presented in Chapter 8 and discussed in [Keysers & Motter⁺ 03] results in an error rate of 0% without tuning using a Gaussian background model with mean zero to represent the background. This result is not surprising due to the

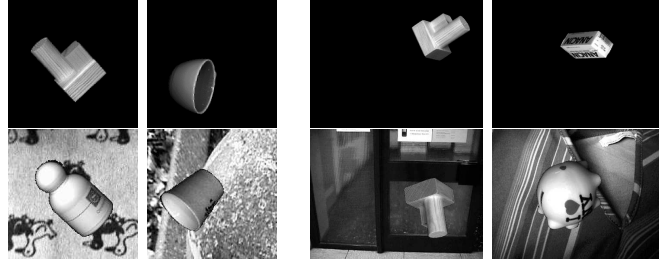


Figure 3.12: Example images from the COIL-i6 data set. Top row: transformation, bottom row: transformation and inhomogeneous background.

homogeneous black background; however, it should be noted that a simple classifier that does not take into account the background is not sufficient to achieve this result.

3.3.2 The COIL-RWTH task

As the COIL-20 database only contains images with homogeneous black background, segmentation of the object from the background is a feasible approach to classification. On the other hand, for real-world images segmentation poses a serious problem. (Although many application areas exist, where a homogeneous or static background can be assumed and existing methods provide acceptable solutions.) Therefore, a new dataset was created based on the objects from the COIL-20 database and a set of new background images. The goal was to create tasks of increasing difficulty to extend the COIL-20 task that can be solved perfectly by existing methods. Each test image carries meta-information about the used transformation parameters for the object images, allowing to separate the effects of different transformations.

We created two corpora that differ in the background used: The COIL-RWTH-1 corpus contains objects placed on a homogeneous black background, whereas the COIL-RWTH-2 corpus contains the objects in front of inhomogeneous real-world background images that were kept separate for training and test images and vary in resolution. The two training and test sets are based on the COIL-20 sets as described above. The training images are of size 192×192 and the size of the test images is 448×336 . In all sets, we applied the following uniformly distributed random transformations to the object images: translation, 360 degree 2D-rotation, and 60–100% scaling with fixed aspect ratio. The data set is publicly available at <http://www-i6.informatik.rwth-aachen.de/~keyzers/COIL-RWTH/>. Figure 3.12 shows example images from the created data set. We do not present error rates for this corpus here, because so far only error rates from within our group are available, which will be discussed along with the experiments performed in Chapter 8.

3.3.3 The Erlangen task

The Erlangen task is an object recognition corpus from the Chair for Pattern Recognition of the University Erlangen-Nürnberg. It contains six tasks in total. In [Reinhold & Paulus⁺ 01] the authors used two databases of images containing five different objects (two different toy cars, two different match boxes, and one decorative box). All images are of size 256×256 . The first of the databases contains images taken with one illumination while in the second

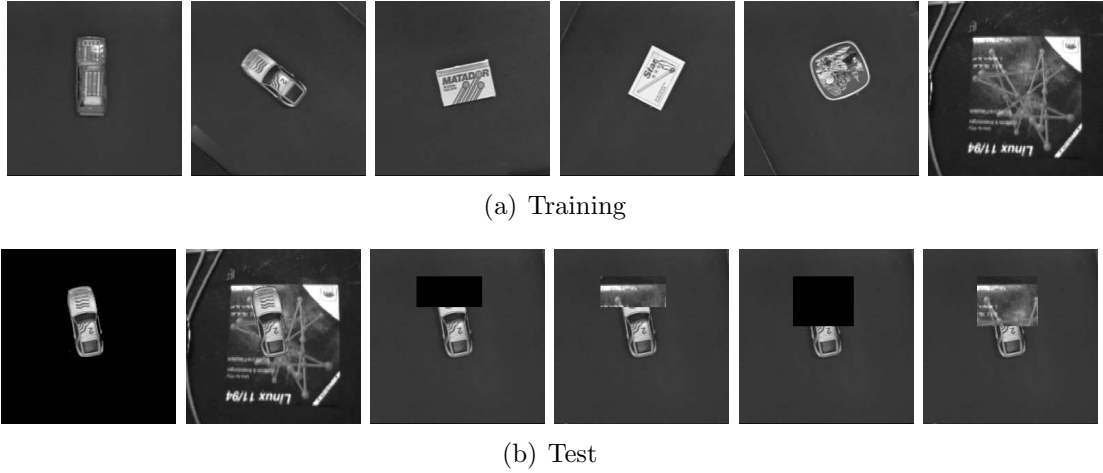


Figure 3.13: (a) The training images and the background of the Erlangen corpus and (b) examples of different test images.

Table 3.12: Error rates as (a) reported by [Reinhold & Paulus⁺ 01] for the Erlangen task and (b) presented in this work.

	condition	25% occlusion	50% occlusion	heterogeneous background
(a)	1 illumination	0.0	2.3	0.0
	2 illuminations	0.0	4.8	0.0
	condition	25% occlusion	50% occlusion	heterogeneous background
(b)	1 illumination	0.0	—	—
	2 illuminations	0.0	0.6	0.0

case the objects are illuminated with two light sources. Each of the training sets contains 18 images per object, taken at different 2D rotation angles on a homogeneous background. Another 17 images per object at rotation angles not occurring in the training set are in the test sets. For each database, three different test sets exist, one with heterogeneous background, and two with two different levels of occlusion. Note that background and occlusions were added artificially to the images. Note also that the background is identical in all of the images and it does not occur in the training images as background (although one image containing only the background exists). The background resolution differed from the resolution of the object images before the creation of the images. This fact might be advantageous when using features based on Gabor filters as in [Reinhold & Paulus⁺ 01]. The objects are shown in Figure 3.13(a) and some examples of the test conditions are displayed in Figure 3.13(b). The results on this corpus, as published in [Reinhold & Paulus⁺ 01], are reproduced in Table 3.12. In this work we report error rates of 0.0% for the case ‘one illumination, 25% occlusion’ using the holistic model in Chapter 8 [Keysers & Motter⁺ 03] and of 0.6% for the case ‘two illuminations, 50% occlusion’ using the patch-based approach as presented in Chapter 7 [Kölsch & Keysers⁺ 04].

3.3.4 The Caltech task

Fergus and colleagues use different datasets for unsupervised object training and recognition of objects [Fergus & Perona⁺ 03]. The task is to determine whether an object is present



Figure 3.14: Examples from the Caltech task: airplanes, faces, motorbikes, and background.

Table 3.13: Reported error rates for the Caltech task.

method		airpl.	faces	motorb.
Euclidean distance (32×32)	[Deselaers & Keysers ⁺ 04a]	24.0	15.0	17.4
statistical model	[Weber & Welling ⁺ 00]	32.0	6.0	16.0
statistical model	[Fergus & Perona ⁺ 03]	9.8	3.6	7.5
discrim. salient patches, SVM	[Gao & Vasconcelos 05]	7.0	2.8	3.8
segmentation	[Fussenegger & Opelt ⁺ 04]	2.2	0.1	10.4
Tamura texture features	[Deselaers & Keysers ⁺ 04a]	1.6	3.9	7.4
Tamura texture regions	[Deselaers, personal comm. 05]	5.4	12.9	3.6
texture feature combination	[Deselaers & Keysers ⁺ 04a]	0.8	1.6	8.5
patches, discrim. train., unscaled	[Deselaers & Keysers ⁺ 05a]	1.4	1.8	2.4
patches, discrim. train., scaled	[Deselaers & Keysers ⁺ 05a]	3.8	7.1	2.5
+ multi-scale	[Deselaers & Keysers ⁺ 05b]	1.1	5.0	1.9
+ brightness-norm	[Deselaers & Keysers ⁺ 05b]	1.4	3.7	1.1

in a given image or not. For this purpose, several sets of images containing certain objects (airplanes, human faces, and motorbikes) and a set of background images not containing any of these objects are available¹¹, see also [Weber & Welling⁺ 00]. The images are of various sizes and for the experiments they were converted to gray level images. The airplanes and the motorbikes task consist of 800 training and 800 test images each, the faces task consists of 436 training and 434 test images. For each of these tasks, half of the given images contain the object of interest and the other half does not. An example image of each set is shown in Figure 3.14. Reported error rates for the task are presented in Table 3.13.

The Caltech data is possibly the most widely used task to evaluate the performance of systems for the detection of general object categories in images in the presence of background. (For specialized tasks, like the detection of human faces, there are other data sets that are used as references.) However, one possible problem associated with this task is that it is possible to achieve very good results using global texture features that are also used in content-based image retrieval [Deselaers & Keysers⁺ 04a]. For these features it is quite obvious that they do not describe the object in the image. Instead, these results suggest that it is possible to distinguish between the two classes by only describing global texture properties, which are correlated with the class indices. For example, many of the airplanes are pictured with the sky in the background or at an airport, which are textures that do not occur for the background class. These results suggest that the task may be too simple to compare real object learning algorithms, at least if only the error rate is used as a criterion.

Observing the very good results using the global Tamura texture features [Deselaers & Keysers⁺ 04a], we performed a small experiment to investigate if performance gains could be obtained by treating the images as consisting of two regions, foreground and

¹¹<http://www.robots.ox.ac.uk/~vgg/data>

background. In this experiment, we considered 11 possible sizes for the foreground region that was always centered in the image. Instead of using the distance that compares the global texture descriptor of two images, we now used the minimum distance that results from comparing background to background and foreground to foreground region descriptors for all possible $11 \cdot 11 = 121$ size combinations. The foreground region distance was weighted with a factor of 0.8 and the background region with a factor of 0.2, which gave the best result on the average. The results are shown in the row labeled ‘Tamura texture regions’ in Table 3.13. We can observe that the error rate is reduced considerably for the ‘motorbikes’ task, while it is increased for the remaining tasks.

4 Pattern recognition for image classification

There are four laws. The third of them, the Second Law, was recognized first; the first, the Zeroth Law, was formulated last; the First Law was second; The Third Law might not even be a law in the same sense as the others.

– P.W. Atkins (about the laws of thermodynamics)

This chapter introduces the basic concepts of classification used in this work and state-of-the-art methods for image classification. Of course we do not try to cover the subject of statistical pattern recognition completely. Such in-depth introductions can be found in [Duda & Hart⁺ 01a, Fukunaga 90, Hastie & Tibshirani⁺ 01]. Nevertheless, we want to present the basic concepts necessary to understand the methods discussed in the following chapters. Some of these methods are closely related to the concepts discussed in this chapter, as e.g. the maximum entropy linear discriminant analysis is to the conventional linear discriminant analysis.

The concepts we are working with are also known as ‘statistical pattern recognition’ because we deal with data that are the results of some measurement and therefore subject to stochastic processes as e.g. image noise. This must be taken into account when the data is modeled. The stochastic nature of measurements naturally leads us to the use of statistical models that provide a well-founded basis for the discussion of uncertainty. The statistical method has been very successful in the domain of speech recognition and natural language processing. Most of the work for this thesis has been carried out working in the surrounding of people very knowledgeable in this domain, which has certainly influenced the approaches taken.

In this thesis we only consider recognition problems with well-defined classes, i.e. the decision function returns one of a finite set of previously defined classes. Questions to be answered include the following types:

- “Which digit is represented in this image?”
- “Which of six defined regions of the human body does this radiography show?”
- “Is there a motorcycle visible in this image or not?”

The performance of a classifier is usually measured by its error rate on a particular data set. Thus, a certain number of observations is classified and the error rate is defined as the ratio between the number of misclassifications and the total number of trials performed. In some applications, as for example image retrieval, other performance measures are also used, but these usually correlate strongly with the error rate of a suitably defined classifier [Deselaers & Keysers⁺ 04a]. Note that in some applications wrong decisions cause different costs, which must then be taken into account. Note furthermore that in some applications

it is more important to determine good estimates of the probability of a sample to belong to the different classes than to reduce the error rate. This is e.g. the case when a sequence of characters is to be classified by combining the individual character classification results and taking into account the likelihood of different sequences.

Because we are interested in the question which classifiers perform well on the data we are interested in, we want to assess their relative performance. When we want to compare two different classifiers we may do so by using the standard methods of statistical testing theory. For example, we may use a t-test to determine the significance of an improvement in the error rate. It should be noted that these tests are usually designed for the case in which the different methods are tested on two different test sets. This is e.g. the case in most medical experiments where two drugs are compared by observing the effects on two sets of patients. Now in most cases in pattern recognition two methods are compared using the same test set. In this case the traditional tests yield a too conservative estimate of the improvement. We may instead use a bootstrap sampling technique to determine the probability of improvement of classifier A versus classifier B and not reject the hypothesis that A is better than B if this probability is larger than e.g. 0.95 [Bisani & Ney 04].

4.1 Basic structure of a classifier

The problem to be solved by pattern recognition algorithms is the following: From a signal or a set of measurements (e.g. from an image) a vector of features $x \in \mathbb{R}^D$ is extracted. Given this feature vector x that belongs to one of the classes $k = 1, \dots, K$, a decision function (or decision rule) is to be constructed which determines the class the original signal belongs to.

Thus, a decision function

$$\begin{aligned} r : \mathbb{R}^D &\longrightarrow \{1, \dots, K\} \\ x &\longmapsto r(x), \end{aligned}$$

must be determined. In many cases, this is done using a discriminant function $g(x, k)$:

$$r : x \longmapsto \operatorname{argmax}_{k \in \{1, \dots, K\}} \{g(x, k)\}$$

where the wanted behavior of the discriminant function usually is

$$\begin{aligned} g(x, k) &\longmapsto 1 && \text{for the 'right' class} \\ g(x, k) &\longmapsto 0 && \text{for the 'false' class} \end{aligned}$$

The discriminant function can be chosen to have different functional forms, one of which – the one resulting from the statistical point of view – is described in more detail in the following section. Other functional forms include for example neural networks, polynomial functions or other function approximators that are trained to yield a regression of the above behavior on the training data. For example, multi-class logistic regression has an interesting interpretation as a discriminant function that is discussed in Section 5.3. Sometimes the concept of regression is directly inherent in a computer vision task, where we can then use the connection between classification and regression to achieve the specific goal. An application of this approach to the problem of determining the appropriate dose for an X-ray system given the image is described in [Keysers & Celik⁺ 02].

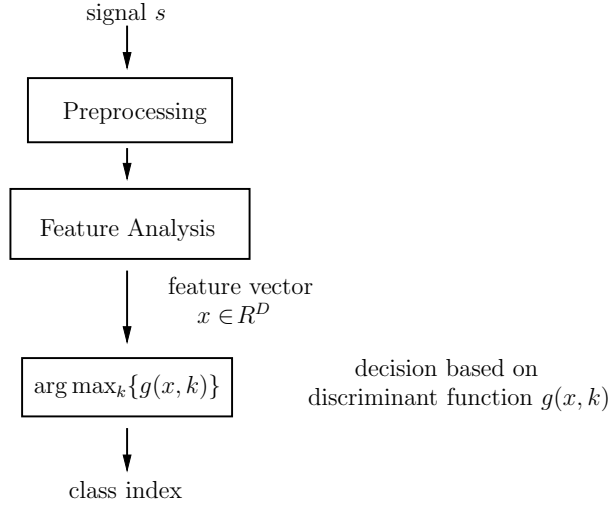


Figure 4.1: Typical structure of a recognition system.

Figure 4.1 illustrates the basic structure of a classifier. Usually, the feature analysis step is preceded by a preprocessing step. In image object recognition for instance, this could be a grayscale normalization or a slant correction of a written character.

The approach for object recognition and scene analysis followed in this work is based on the so-called appearance-based paradigm. In the appearance-based approach, the features used for classification are the pixel intensity values themselves (after a possible preprocessing like e.g. brightness normalization). “The appearance-based approaches to vision problems have recently received renewed attention in the vision community due to their ability to deal with combined effects of shape, reflectance properties, pose in the scene, and illumination conditions.” [Leonardis & Bischof 00]. Another advantage of this approach is that a segmentation of the object in the image is unnecessary. Therefore, errors in this segmentation step cannot occur. Although it is well-known that segmentation is a difficult process and can lead to errors it is used in a variety of different recognition applications. (Of course there are settings, in which a segmentation can be done with almost no errors, as for example in the case of a uniform background in front of which the object is present.) In some experiments described briefly later in this work it has been shown that the appearance-based approach is competitive even for domains such as gesture recognition, in which this paradigm is a novelty.

The appearance-based approach is not to be seen as a contrast to the use of other features, such as texture features, but as contrasted to approaches that first try to segment the image or that only compute global descriptors such as moments of the image grayvalues. We consider the appearance-based approach to include the use of features derived from the image such as the spatial derivative as computed by a Sobel-filter.

In this work an image x is considered a real valued function on a discrete image grid consisting of pixel locations from $\mathcal{I} \times \mathcal{J} = \{1, \dots, I\} \times \{1, \dots, J\}$, that is $x \in \mathbb{R}^{I \times J}$. On the other hand an image can be considered a simple feature vector with one dimensional indices and dimension $D = I \cdot J$. The graylevel value of an individual pixel at pixel position (i, j) will generally be denoted by x_{ij} .

4.2 Bayes' decision rule

In statistical pattern recognition, Bayes' decision rule is employed, which uses a specific model for $g(x, k)$. In that particular case, the class k is chosen which maximizes the posterior probability $p(k|x)$ given an observation x to be classified:

$$\begin{aligned} r(x) &= \operatorname{argmax}_k \{p(k|x)\} \\ &= \operatorname{argmax}_k \left\{ \frac{p(k) \cdot p(x|k)}{\sum_{k'=1}^K p(k') \cdot p(x|k')} \right\} \end{aligned} \quad (4.1)$$

As the denominator of Equation (4.1) does not depend on k , it can be neglected for classification purposes, arriving at

$$r(x) = \operatorname{argmax}_k \{p(k) \cdot p(x|k)\}, \quad (4.2)$$

where $p(k)$ is the prior probability of class k and $p(x|k)$ is the class conditional probability for the observation x given class k . Note that we do not distinguish in notation between probability distributions like $p(x|k)$ and true probability functions like $p(k)$ here. We also do not separate the random variables and their outcomes in the notation. We believe that the notation nevertheless is sufficiently clear.

It can be shown that the Bayes rule is optimal with respect to the expected number of errors in case the true distributions $p(k)$ and $p(x|k)$ are known [Duda & Hart⁺ 01a]. Note that this implies the assumption of a cost function assigning cost one to a misclassification and cost zero to a correct classification. It does not hold for the case in which some errors involve higher losses than others. For instance, a false-positive cancer detection in a medical application could be a 'cheap' misclassification (as the following examinations will show that the patient does not suffer from cancer), whereas a false negative result should result in high costs (as the patient is regarded to be healthy, delaying the necessary cancer therapy).

Since the true distributions are usually unknown, we must choose suitable models for $p(k)$ and $p(x|k)$ (or directly for $p(k|x)$) in order to use Bayes rule in real world applications. The free parameters of these models are then estimated during the training phase. Throughout this work, the training phase is considered to be supervised. That is, we are given training data as a set of labeled pairs (x_n, k_n) , $n = 1, \dots, N$ where x_n is a feature vector belonging to class k_n . This training data is then used to estimate the free model parameters. Sometimes we might prefer to write the training data as x_{kn} , $k = 1, \dots, K$, $n = 1, \dots, N_k$, where $N = \sum_k N_k$ and the first index denotes the class and the second index counts the samples of each class. I.e., N_k represents the number of training samples of class k , and x_{kn} denotes the n -th reference pattern of class k .

The a priori distribution is usually modeled by relative frequencies $p(k) = N_k/N$ or (e.g. in the case of digit recognition) it is often set to $p(k) = \frac{1}{K}$, i.e. it is considered to be uniform.

Models for $p(x|k)$

For the class conditional distributions we will consider mainly:

- unimodal (or single) Gaussian densities:

$$p(x|k) = \mathcal{N}(x|\mu_k, \Sigma) = \det(2\pi\Sigma)^{-\frac{1}{2}} \cdot \exp \left[-\frac{1}{2}(x - \mu_k)^T \Sigma^{-1} (x - \mu_k) \right]$$

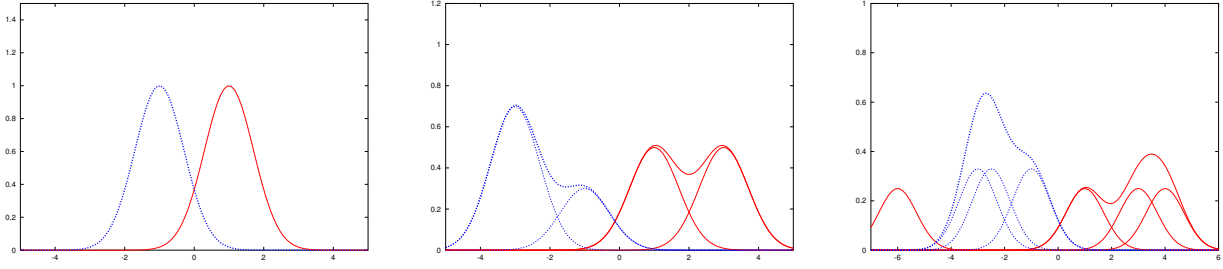


Figure 4.2: This figure uses a one-dimensional example with two classes to illustrate the three cases for the class-conditional distribution $p(x|k)$ using single Gaussian densities (left), Gaussian mixture densities (center, note the different weight of the densities), and Gaussian kernel densities (right, all training samples contribute with an equal weight).

- Gaussian mixture densities:

$$p(x|k) = \sum_{i=1}^{I_k} p(i|k) \cdot \mathcal{N}(x|\mu_{ki}, \Sigma)$$

- Gaussian kernel densities:

$$p(x|k) = \frac{1}{N_k} \sum_{n=1}^{N_k} \mathcal{N}(x|x_{kn}, \Sigma) \quad (4.3)$$

Here, the $\mu_k, \mu_{ki} \in \mathbb{R}^D$ are the mean vectors and the covariance matrix $\Sigma \in \mathbb{R}^{D \times D}$ is symmetric and positive definite. Note that the kernel density case is included in the mixture density case for equal mixture weight and identification of the mean vectors with the training samples.

Most of the theoretical results are derived for the case of Gaussian densities. In most cases, this does not impose any restrictions, as these results can be transferred to mixture or kernel densities, which can model any density function up to arbitrary precision. For example, the probabilistic interpretation of tangent distance can be transferred easily to kernel densities. One possible exception here is the discussion of the maximum entropy framework, where this transfer is not as straightforward.

For mixture densities, we assume Σ to be identical for all classes, i.e. we use variance pooling over classes. For some tasks (especially for larger number of dimensions) we also use pooling over dimensions, i.e. $\Sigma = \alpha \sigma^2 I$ with a factor α to determine the width of the density [Keysers & Macherey⁺ 04]. The relationship of the kernel density to the nearest neighbor classifier is discussed briefly further below.

4.3 Other classification approaches

The statistical pattern recognition approach is one of the three main approaches besides the empirical nonlinear approach for discriminant functions using artificial neural nets (ANN) and the support vector machine (SVM) approach based on statistical learning theory. This

distinction between approaches is somewhat arbitrary, since e.g. a support vector machine can be seen in the context of statistical pattern recognition. Furthermore, it can be shown that with respect to the squared error, the global optimum of an ANN is reached if the discriminant function equals the a posteriori probability density function [Ney 95]. There also exists a variety of methods based on rules or decision trees.

Artificial neural nets originated from trying to mimic the operation of the interconnected neural cells in the human brain. An artificial neural net usually consists of multiple layers of connected nodes, called neurons. It can be shown that one hidden layer (i.e. a net with input-, output- and one additional layer) is sufficient to model an arbitrary function. At each node a weighted sum of all input signals is computed. The output of a node is then computed to be a non-linear function of this weighted sum (usually, a sigmoid function is used). In many cases, given observations $x \in \mathbb{R}^D$ coming from K classes, the input layer of an artificial neural net consists of D neurons and the output layer of K neurons. The output neuron with maximum output (or activation) then determines the class which an observation is assigned to. Once the topology of the net has been chosen (number of layers, number of nodes, which neurons are connected etc.), the training problem is to choose the required weight coefficients in such a way that the net explains the available training data as accurately as possible, where usually a mean squared error criterion is used. One of the best known training procedure for artificial neural nets is the error-backpropagation method. Advantages of ANNs include ease of use and high performance in many applications, disadvantages include the lack of criteria to choose the topology of the net and the difficulty of interpreting the resulting behavior of the net after training. For a detailed introduction to ANN see e.g. [Bishop 95].

A way to formalize learning a classification function from examples is statistical learning theory [Vapnik 95, Vapnik 98, Vapnik 99]. One central point of the analysis of learning algorithms is the so called Vapnik-Chervonenkis dimension, equal to the maximum number of vectors from two classes which can be separated in all possible ways using (discriminant) functions of this set. This number is related to such notions as generalization ability, minimum description length and overfitting [Vapnik 99]. One basic result is that there exists a tradeoff between the quality of approximation and the complexity of the approximating function. From statistical learning theory the support vector machine (SVM) evolved, which uses optimal separating hyper-planes in high-dimensional feature spaces. It effectively transform patterns into high-dimensional space, constructs a hyper-plane for separation of classes and thus allows algorithmic control of the Vapnik-Chervonenkis-dimension. One important aspect in this transformation is that it does not have to be carried out explicitly. It suffices to be able to compute the scalar product in that space, which is done using a kernel function. One finding is that only few training examples are effectively used in constructing the hyper-planes, which are called support vectors and are characteristic for the discrimination problem. Note that the set of support vectors often comprises a significant fraction of the original training data. Classification can then be done by comparison with the support vectors. SVMs can also be equipped with transformation invariance in various ways [Schölkopf & Simard⁺ 98, Haasdonk & Keysers 02]. The idea of the support vector machine has been extended to the probabilistic relevance vector machine [Tipping 00].

Support vector machines, methods like k -NN, and kernel densities are usually considered memory-based techniques, because (a subset of) the training samples is memorized and compared to the observation during classification. In contrast to this, methods like ANNs are regarded as learned function techniques, since the training data are used to determine the free parameters of a discriminant function. As in most cases, such a distinction is somewhat arbi-

trary, because the memorized examples can be considered parameters of a complex function.

As a drawback of memory-based methods it is sometimes seen that they are time consuming because the test pattern has to be compared to all stored references. This problem is of much lower importance in many application today than in the past because the available computing power and memory capacity has grown significantly. Furthermore, as computers continue to become faster, this steadily becomes less of a drawback; Moore's law is considered to remain in effect for the next generations of computers. To overcome this problem in cases where necessary, a number of solutions have been proposed, including hierarchies of distances or models for representing large subsets of prototypes [Simard & Le Cun⁺ 98a, Hastie & Simard⁺ 95]. One can also use methods such as partial distance calculation, hierarchical structuring of the training vectors, or related approaches.

The choice of the model or classifier to use is in general somewhat arbitrary, but an empirical analysis [Sohn 99] shows that the accuracy of different algorithms depends on the data characteristics. For example k -NN performance decreases as the relative number of feature variables to the training cases increases. A recent study on the performance of different classifiers on data sets of varying characteristics has also led to interesting results in this direction [Bernado-Mansilla & Ho 04]. On the other hand, it is possible to show that for each regularity that a given machine can learn there exists another regularity for that the machine does the opposite, that is it generalizes worse than a random classifier. This statement is sometimes referred to as the 'no free lunch'-theorem.

4.4 Parameter estimation

In in almost all cases of a (statistical) classification system we have a set of free parameters that need to be determined before the actual classification can be performed. We will briefly introduce some techniques that play a role in the experiments that were performed with Gaussian models.

4.4.1 Maximum likelihood estimation

As described above, algorithms for the classification of observations $x \in \mathbb{R}^D$ into one of the classes $k \in \{1, \dots, K\}$ usually estimate some of their parameters in the training phase from a set of labeled training data $\{(x_n, k_n)\}$, $n = 1, \dots, N$. One widely used method to determine parameters from a set of given data is maximum likelihood estimation. (This term usually refers to the likelihood of the class-conditional density, and we will use it with this meaning, although it can also refer to the likelihood of the posterior.) Consider a density function $p_\Lambda(x|k)$ that depends on a parameter set Λ . The likelihood function is then given by

$$\Lambda \mapsto \prod_{n=1}^N p_\Lambda(x_n|k_n). \quad (4.4)$$

respectively the log-likelihood function is

$$\Lambda \mapsto \sum_{n=1}^N \log p_\Lambda(x_n|k_n). \quad (4.5)$$

Then the maximum likelihood estimator $\hat{\Lambda}$ is the set of parameters that maximizes the (log-)likelihood:

$$\hat{\Lambda} := \operatorname{argmax}_{\Lambda} \prod_{n=1}^N p_{\Lambda}(x_n | k_n) = \operatorname{argmax}_{\Lambda} \sum_{n=1}^N \log p_{\Lambda}(x_n | k_n). \quad (4.6)$$

For non-discriminative approaches the parameters for each task can be determined separately, which results in the total log-likelihood for the complete training data to separate for the classes:

$$\Lambda \mapsto \sum_{k=1}^K \sum_{n=1}^{N_k} \log p_{\Lambda}(x_{kn} | k). \quad (4.7)$$

4.4.2 Discriminative training

In maximum likelihood training, the training procedure takes into account only the data from one class at a time. In contrast to this, in discriminative training all of the competing classes can be considered at the same time. In the latter case the process is called discriminative. As discriminative training puts more emphasis on the decision boundaries, it often leads to better classification accuracy.

If we denote by Λ the set of free parameters of the distribution, the maximum likelihood approach consists of choosing the parameters $\hat{\Lambda}$ maximizing the log-likelihood on the training data as described in Equation (4.6). Alternatively, we can maximize the log-likelihood of the class posteriors,

$$\hat{\Lambda} = \operatorname{argmax}_{\Lambda} \sum_{n=1}^N \log p_{\Lambda}(k_n | x_n), \quad (4.8)$$

which is also called discriminative training, because the information of out-of-class data is used. This criterion is often referred to as maximum mutual information (MMI) criterion in speech recognition, information theory and image object recognition [Dahmen & Schlüter⁺ 99, Normandin 96].

If we rewrite the posterior probabilities using Bayes theorem, we arrive at a different formulation of the MMI criterion:

$$\sum_{n=1}^N \log \frac{p(k_n) \cdot p(x_n | k_n, \lambda)}{\sum_{k=1}^K p(k) \cdot p(x_n | k, \lambda)}, \quad (4.9)$$

where the prior probabilities $p(k)$ are assumed to be given. A maximization of the MMI criterion defined above therefore tries to simultaneously maximize the class conditional probabilities of the given training samples and to minimize a weighted sum over the class conditional probabilities of all competing classes. Thus, the MMI criterion optimizes the class separability.

4.4.3 The expectation-maximization algorithm

The Expectation-Maximization (EM) algorithm is an iterative method used to estimate the parameters of a probability density function in the presence of hidden variables. These

hidden variables are usually modeled as probability distributions themselves and a marginalization is performed. A very nice overview of applications of the EM algorithm in computer vision is given in [Forsyth & Ponce 03]. For example consider the task of image segmentation using a probabilistic model of pixel attributes. Here, the hidden variable is the assignment of each pixel to one of the segments the image is supposed to consist of. Given the assignment, it would be simple to estimate the distribution for each segment. Similarly, given the distribution for each segment, it is straightforward to determine the assignment of pixels to segments. Now the idea of the EM algorithm is to iterate these two procedures starting with some appropriate initial assignment and update assignment and distribution models in each step until convergence.

The application of the EM algorithm to mixture densities as it was also used for some experiments in this work is described in detail e.g. in [Dahmen 01]. Here the hidden variable is the assignment of an observation or a training sample to one of the densities in the mixture. This assignment is treated as a probability distribution over the mixture components. If we assign each observation completely to one component, we use the so-called maximum approximation. In the experiments, the number of densities to be trained per mixture as well as their initial parameters are defined by repeatedly splitting mixture components, i.e. a Linde-Buzo-Gray [Linde & Buzo⁺ 80] method is used. Note that choosing the number of mixture components in this case is a problem, because the log-likelihood of the model keeps improving while the number of densities is increased, which may lead to overfitting, i.e. lower generalization performance. Methods like the minimum description length principle can be used to overcome this effect [Hastie & Tibshirani⁺ 01].

4.5 Dimensionality reduction

A typical problem for statistical classifiers based on Gaussian mixture densities or kernel densities is the estimation of covariance matrices. In case of the USPS task, with feature vectors $x \in \mathbb{R}^{256}$, a single covariance matrix requires (due to symmetry) the estimation of $256 \cdot (256 + 1)/2 = 32,896$ parameters. Given 7,291 training samples with a dimensionality of 256, we need to estimate these parameters from $7,291 \cdot 256 = 1,866,496$ measurements. Although this seems reasonable, it can be observed in a variety of settings that a reduction in the number of parameters to be estimated can be beneficial for classification performance. A common approach to reduce the number of parameters is the use of variance pooling

- class-specific variance pooling:
estimate only a single Σ_k for each class k , i.e. $\Sigma_{ki} = \Sigma_k \forall i = 1, \dots, I_k$
- global variance pooling:
estimate only a single Σ , i.e. $\Sigma_{ki} = \Sigma \forall k = 1, \dots, K$ and $\forall i = 1, \dots, I_k$

in combination with diagonal covariance matrices.

Another way to overcome the difficulties with the estimation of covariance matrices is the use of feature reduction. The aim of feature reduction is to capture the essential information (for discrimination) of the high dimensional feature vector in a smaller number of features, usually by means of a linear transformation of the feature space, but nonlinear methods are also used. In the following sections, two methods that are frequently used are presented. Feature reduction also reduces the variance of the classifier (by increasing its

bias). For a discussion of the bias/variance trade-off see e.g. [Hastie & Tibshirani⁺ 01]. In applications of statistical learning and data mining, where features of different semantics are available, it is also common to perform feature selection, i.e. to disregard some of the inputs, but this approach will not be considered here.

Note that feature reduction always involves a loss of information. It can be shown that the information gained for classification from an additional feature is always positive [Fukunaga 90]. Yet, in practical applications, this theoretical loss introduced by feature reduction is often compensated by a more reliable parameter estimation in the reduced feature space. Furthermore, if an additional feature is affected by a high level of noise, it may only lead to improvements for classification in the case of large amounts of training data, which are usually not available. Therefore the classifier may generalize better if fewer features are used. This effect is also related to the so-called curse of dimensionality, which describes the effect of data sparseness in high-dimensional spaces. (For example, if we want to estimate a distribution on a set of n binary features by using a minimum number of samples for each possible feature combination, we need on the order of 2^n samples to accomplish this.) For more discussion on the curse of dimensionality the reader may refer to [Hastie & Tibshirani⁺ 01].

4.5.1 Principal components analysis

The principal components analysis (PCA) is a linear transformation aimed at minimizing the representation error in the reduced feature spaces. It is sometimes also called Karhunen-Loève transformation. After calculating the empirical covariance matrix Σ , it is diagonalized using an eigenvector decomposition with eigenvectors v_1, \dots, v_D and corresponding eigenvalues $\lambda_1, \dots, \lambda_D$. We assume the eigenvalues are sorted in decreasing order, i.e. $\lambda_d \geq \lambda_{d+1}, d = 1, \dots, D - 1$. This decomposition can numerically also be determined using e.g. the more general singular value decomposition [Press & Teukolsky⁺ 92]. Then Σ can be written as

$$\begin{aligned}
\Sigma &= \sum_{d=1}^D \lambda_d v_d v_d^T \\
&= [v_1 \cdots v_D] \text{diag}(\lambda_1, \dots, \lambda_D) [v_1 \cdots v_D]^T \\
&= [v_1 \cdots v_D] (\text{diag}(\lambda_1, \dots, \lambda_D))^{\frac{1}{2}} \left([v_1 \cdots v_D] (\text{diag}(\lambda_1, \dots, \lambda_D))^{\frac{1}{2}} \right)^T \\
&= \Sigma^{\frac{1}{2}} (\Sigma^{\frac{1}{2}})^T
\end{aligned} \tag{4.10}$$

where the last steps are given in order to help understand the considerations of following chapters, in which $\Sigma^{-\frac{1}{2}}$ is used, being the inverse of $\Sigma^{\frac{1}{2}}$. ($\Sigma^{-\frac{1}{2}}$ is also the transformation matrix of the whitening transformation. After application of the whitening transformation the covariance matrix in the transformed space is equal to the identity matrix and the distribution is called ‘white’ (compare [Fukunaga 90, pp. 26ff]).) The eigenvectors v_1, \dots, v_d (for some d fixed or to determine) corresponding to the largest eigenvalues are also referred to as principal components. Now the PCA consists of representing each vector by its projection onto the principal components, which is a linear transformation $x \in \mathbb{R}^D \mapsto \hat{x} \in \mathbb{R}^d$ with the matrix representation of the transformation being $[v_1 \cdots v_d]$, which has the property that the expected squared error $E\{\|x - \hat{x}\|^2\}$ is the smallest within all linear transformations to d dimensions.

Note that the PCA discards the directions of small variance. Doing so, it is usually expected that the transformation captures the ‘most relevant’ part of the information contained in the vectors x . This point of view of information based on magnitude of variance and minimal reconstruction error may not be suitable for classification purposes, because it does not take into account the class information. In fact, in some pattern recognition applications, a PCA transformation is used that discards the first few eigenvectors (for example the first three in [Martinez & Kak 01]). This approach may be interpreted in the following way: the first large eigenvectors may capture variability in the data that is not relevant for classification because it is common to all classes. For example, in many image classification settings, the first eigenvector corresponds to the brightness variability of the images, and it may be beneficial to reduce the impact of brightness changes on the feature vectors. Eigenvectors with small corresponding eigenvalues on the other hand are often attributed to noise components and therefore discarded. If the first PCA components are discarded in a class-specific or density-specific setup, doing so bears some similarity to the tangent vector approach described in Section 5.2. In both cases, the part of the distance that is in the direction of the PCA vectors or tangent vectors does not contribute to the overall distance that is then used for classification.

Further information on the PCA and the whitening transformation can be found in [Fukunaga 90]. Note that no class information is used when computing the PCA. Thus, although it is often used in pattern recognition tasks, in general nothing can be said about the discriminative power of the computed features.

4.5.2 Linear discriminant analysis

The linear discriminant analysis (LDA), also called Fisher’s LDA, takes into account the class information within the feature reduction process and aims at maximizing the separability of the classes in the transformed feature space [Duda & Hart 73, pp. 118ff]. Since this is what is usually wanted in pattern recognition, the LDA has advantages over the PCA in some applications.

A new and different form of linear discriminant analysis based on the maximum entropy framework has been derived in the course of this work. It is called maximum entropy linear discriminant analysis (MELDA) and is presented in Section 5.4.

One of the assumptions made to derive the LDA is that the class conditional distributions are Gaussian. Under this assumption, the LDA tries to simultaneously maximize the distances between the class centers μ_k and to minimize the distances within each class. This can be achieved by first determining within-class and between-class scatter matrices. The optimization problem then leads to a generalized eigenvalue problem. Another method leading to the same result is to employ a whitening transformation on the within-class covariance matrix and then (using the fact that the within-class scatter matrix is the identity matrix) use the subspace spanned by the vectors $\mu_k - \mu$ where μ is the total mean vector of all samples.

As the overall mean vector μ is a linear combination of the class-specific mean vectors μ_k , the dimensionality of the obtained subspace is at most $K - 1$, which can be too small for some applications. A method to circumvent this problem is to create so-called ‘pseudo-classes’ using a clustering algorithm and then use the LDA within the new set of data with $K' > K$ (pseudo-) classes yielding at most a $K' - 1$ -dimensional feature space. This is done by performing a cluster analysis on the available data and by interpreting each of the

resulting clusters as a pseudo-class. For instance, for the task of digit recognition ($K = 10$), four pseudo-classes can be created per class, yielding a reduced feature space of $K' - 1 = 39$ dimensions.

The LDA can be computed as follows. In a first step, we compute the within-class-scatter matrix S_w and the between-class-scatter matrix S_b :

$$S_w = \sum_{k=1}^K \sum_{n=1}^{N_k} (x_{kn} - \mu_k) \cdot (x_{kn} - \mu_k)^T \quad (4.11)$$

$$S_b = \sum_{k=1}^K N_k \cdot (\mu_k - \mu) \cdot (\mu_k - \mu)^T \quad (4.12)$$

and compute the eigenvectors and eigenvalues of the matrix $S_w^{-1} \cdot S_b$. In a second step, compute the projection of the data onto the subspace spanned by the first d principal components of $S_w^{-1} \cdot S_b$. To avoid the inversion of S_w , the LDA can also be computed by solving a generalized eigenvalue problem in S_w and S_b [Duda & Hart 73].

The criterion underlying the LDA can be formally stated in various ways, two of which follow here. We are looking for a projection matrix $V \in \mathbb{R}^{D \times (K-1)}$ maximizing between class distances with respect to within class distances. Two equivalent ways to formalize this goal are the following:

$$\max_V \text{Tr}(V^T S_b V) \text{ with } V^T S_w V = I \text{ resp. } \min_V \text{Tr}(V^T S_w V) \text{ with } V^T S_b V = I \quad (4.13)$$

In the first formulation it is easy to see that after a whitening transformation (i.e. with $S_w = I$) we are looking for a set of orthonormal vectors that span the same subspace from which S_b is constructed, which is the outer product of the vectors $(\mu_k - \mu)$. Therefore, if $S_w = I$, the LDA projection matrix V consists of the $(K - 1)$ -dimensional orthonormal basis of $\text{span}(\mu_1 - \mu, \dots, \mu_K - \mu)$.

4.5.3 Penalized linear discriminant analysis

For images, the correlation between the values of neighboring pixels often leads to negatively correlated coefficients in the projection vectors of the LDA projection matrix V , which then have a so-called ‘salt and pepper’ structure [Hastie & Tibshirani⁺ 01]. Examples of these vectors are shown in Figure 4.4 for images from the USPS data set (described in Section 3.1.1).

To counteract this effect we can include an additional smoothness criterion for the projection vectors of the LDA, i.e. negatively correlated coefficients in V are penalized. Thus we arrive at the penalized linear discriminant analysis (PDA) [Hastie & Buja⁺ 95]. The penalty takes the form

$$\min \text{Tr}(V^T \Omega V) \quad (4.14)$$

where $x^T \Omega x$ is large for coarse image vectors x , e.g. using the discrete second derivative or Laplacian. The complete criterion for the PDA then is (cp. (4.13)):

$$\min \text{Tr}(V^T (S_w + \gamma \Omega) V) \text{ with } V^T S_b V = I \quad (4.15)$$

with the relative PDA weight $\gamma \cdot \text{Tr}(\Omega) / \text{Tr}(S_w)$.

Table 4.1: USPS error rates [%] for penalized discriminant analysis and Gaussian single densities.

subspace dimensionality	LDA	PDA	relative improvement
9	11.5	11.1	3%
39	12.7	11.8	7%

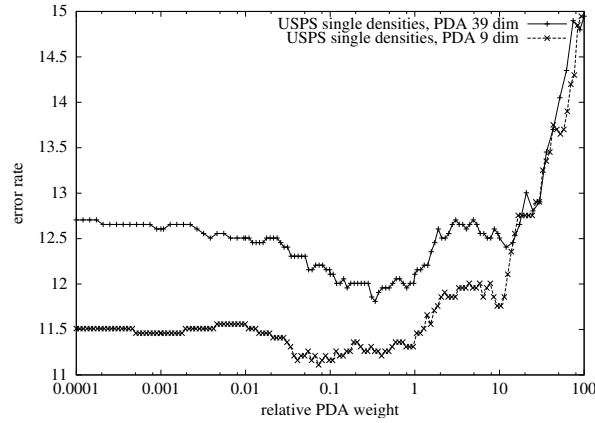


Figure 4.3: USPS error rates for penalized discriminant analysis with respect to relative PDA weight.

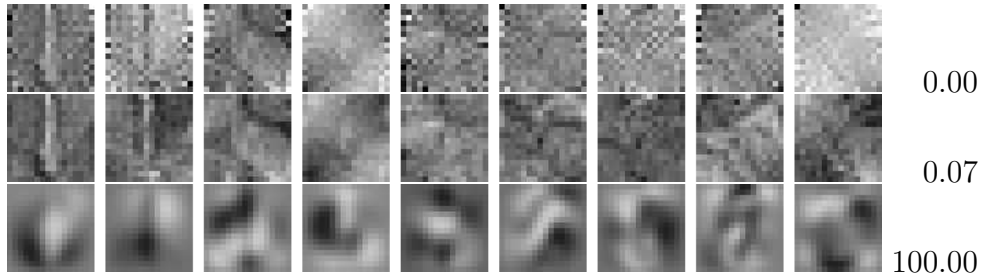


Figure 4.4: Projection vectors of the PDA for a subspace dimensionality of $K - 1 = 9$ and different relative PDA weights. The weight 0.0 corresponds to the conventional LDA, the weight 0.07 achieves the best error rates and with the weight 100.0 we can easily observe the effect of the coarseness penalty.

Hastie and colleagues report a relative improvement of 25% in error rate on a handwritten digit recognition task using the PDA. In experiments performed on the similar USPS task, we were only able to obtain an improvement of up to 7% relative starting from a relatively high overall error rate. The results are summarized in Table 4.1 and can be compared to those of Table 3.2 given earlier. Figure 4.3 shows the evolution of the error rates with respect to the relative PDA weight and Figure 4.4 shows example projection vectors for the PDA with different relative PDA weights. A relative weight of 0 corresponds to the conventional LDA.

4.6 Distance-based classification

In this section we quickly review a few topics relevant to the distance-based classification approaches adopted in several parts of this work. Note that we do not always use the term ‘distance measure’ in the mathematically strict sense, i.e. many of the dissimilarity measures we use are not strictly distance measures, because they e.g. do not fulfill the triangle inequality.

4.6.1 Distance functions and probability density functions

Since distance-based classifiers play an important role in this work, the connection to the statistical point of view is briefly considered here. Consider a Gaussian or normal distribution

$$p(x|k) = \mathcal{N}(x|\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} \exp \left(-\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \right) \quad (4.16)$$

and consider the discriminant function for a Bayesian classifier $g(x, k) = \log[p(k)p(x|k)]$. If the terms that are constant in k are dropped, we arrive at

$$g(x, k) = -\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) - \frac{1}{2} \log |\Sigma_k| + \log p(k) \quad (4.17)$$

The term $\alpha_k := -\frac{1}{2} \log |\Sigma_k| + \log p(k)$ does not depend on the feature vector x and is ignored for the next few steps. We now define

$$g(x, k) =: -d_k(x, \mu_k) \quad (4.18)$$

and thus with the so called Mahalanobis distance

$$d_k(x, \mu_k) = (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \quad (4.19)$$

the decision rule finally becomes

$$r(x) = \arg \min_k \{d_k(x, \mu_k)\} \quad (4.20)$$

which is called the nearest mean or nearest prototype decision rule. If we now reconsider the previously ignored term α_k , we see that the decision rule is altered by adding a constant offset to the distances of each hypothesized class, which is influenced by the covariance structure (small entries in the inverse covariance matrix yield smaller distances, which is counteracted by α_k) and the a priori probability of the class (classes with higher a priori probability result in smaller distances).

If we assume that each element of the training data set is used as a prototype, the Gaussian case corresponds to the kernel density classifier as described above. We can then use the so-called maximum approximation, meaning that we assume that the sum $\sum_{n=1}^{N_k} \mathcal{N}(x|x_{kn}, \Sigma)$ over all densities is dominated by one of the normal terms and it therefore suffices to use that largest term for the classification. The nearest prototype rule then becomes the nearest neighbor (NN) decision rule, which is also sometimes called minimum distance classifier:

$$r(x) = \arg \min_k \left\{ \min_{n=1, \dots, N_k} d_k(x, x_{kn}) \right\} \quad (4.21)$$

In many cases, experiments show that the maximum approximation made here does not decrease the performance of the resulting classifier much. But the NN classifier can be generalized to the k -nearest neighbor algorithm, which is more similar to the kernel density approach in that it uses more than one of the closest reference samples. (Please excuse the overloading of k in this part of the text. It should be clear from the context, whether k is meant to be a class index or the number of prototypes in the k -nearest neighbor classifier.) In the k -nearest neighbor classification, the classes of the k closest prototypes to the observation x are determined and the decision of the classifier is based a voting scheme, usually majority voting, where the class is chosen that most of the k closest prototypes belong to. It is also possible to use weights for the votes of the prototypes based on their distance to the observation. If the weight is chosen proportional to the exponential of the distance and k approaches N , the k -NN is equivalent to the kernel density classifier.

It can be shown that in the theoretical case of an infinite amount of training data the error rate for the NN classifier is at most twice the Bayes error rate. If the number of nearest prototypes k in the k -NN also approaches infinity, the error rate of the k -NN even approaches the Bayes error rate.

Many techniques have been developed to suitably reduce the number of reference vectors required, among them the editing or condensing techniques [Devijver & Kittler 82]. These techniques try to reduce the available references to those lying near class-borders in feature space. Thus, they are somewhat related to the idea of support vector machines. Yet, on today's state-of-the-art computers these drawbacks are somewhat alleviated and nearest neighbor techniques are applicable in many real-world problems. Due to its simplicity, a 1-nearest neighbor-based classifier is often used to produce baseline error rates to be compared with more sophisticated approaches.

In many experiments described in this work we adopt a distance-based classifier to investigate the influence of using different distance functions on the error rate. To do so, we often use a simple decision rule like the 1-NN or 3-NN rule and concentrate on the effects of the distance measures as described in Section 4.7.3.

4.6.2 Hierarchical filtering

Since some distance measures are computationally far more expensive than e.g. the Euclidean distance, we can use the a less costly distance as a pre-filter [Simard & Le Cun⁺ 93, Simard 94]. This method of hierarchical filtering is a special approach for distance-based classifiers where different distance measures with different reliability and computational costs are available. It consists of first computing the less costly distance and sorting out the most unlikely references. In a second step, the distances for the remaining samples are recomputed using the more expensive distance measure (e.g. the tangent distance), yielding better estimates of the respective distances.

For example, in the experiments performed with pre-filtering on the USPS database with about 7,000 training samples, it was observed that a Euclidean pre-filter which extracts 100-500 samples before calculation of more costly distance measures like the tangent distance or the two-dimensional distortion models already was sufficient in the sense that a larger set did not change classification results. The application of this method is especially important when using some computationally very demanding two-dimensional deformation models as described in Chapter 6.



Figure 4.5: Pattern to be classified (left), two prototypes. According to Euclidean distance the pattern to be classified is closer to the first prototype. A distance measure invariant to line thickness should find that the second, correct prototype is closer. (Compare [Simard & Le Cun⁺ 98a])

4.7 Invariance in classification

The main topic of this document is the modeling of variability for image recognition. We therefore give an overview of some widespread techniques to achieve invariance in classification in this section.

Our goal is to obtain a classifier that is invariant with respect to certain transformations of the data that are known to leave the class unchanged. This goal can be addressed in different stages of the classification process: in the preprocessing step the feature vectors can be normalized, during feature analysis we can extract invariant features and we can use invariant probability density functions, which are inherently related to invariant distance measures.

One example of the importance of invariance in image recognition is depicted in Figure 4.5. Here, an observed pattern is shown, which contains the object of a handwritten digit ‘7’. If it is compared with the two references shown on the right side, a classifier based on Euclidean distance would find that it is closer to the first reference, showing an image of the digit ‘9’, because the sum of squared grayvalue differences is smaller than the one for the second, correct reference. If the classifier used a distance measure invariant to the line thickness, it would find that the correct image is more similar to the observation and therefore correctly classify the given pattern. Note that if a sufficiently large set of training data is available, it would probably also contain versions of the digit ‘7’ with modified line thickness, such that the advantage of invariance would be reduced.

There exists a variety of techniques for solving the problem of invariant pattern recognition including integral transforms, construction of algebraic moments, invariant distance measures, and the use of structured neural networks, where in all cases it is usually assumed that the nature of the invariance is known a priori [Wood 96]. The last statement is quite essential in most approaches. In contrast to this restriction to domain knowledge, a method to estimate the derivatives of transformation from the given data is presented in Section 5.2.2.

A theoretical statement of invariance can be given as follows: Consider patterns as functions on some set, e.g. in image recognition $x : (i, j) \in I \times J \mapsto x_{ij}$, furthermore there exists a classification function which maps patterns onto class numbers, e.g. $r : x \mapsto 1, \dots, K$ and a transformation group \mathcal{G} which acts on the set the pattern is defined on and therefore on the pattern space, e.g. $\forall g \in \mathcal{G} : (gx)_{ij} = x_{g^{-1}(i,j)}$, and does not affect class membership. Thus the desired classification function should be invariant under the action of the group, that is $\forall g \in \mathcal{G} : r(gx) = r(x)$. That is, the patterns with the same invariant content form an equivalence class with respect to a group operation describing the geometric transform

[Burkhardt & Fenske⁺ 92]. In practice, in some cases we want to restrict the actions of the group, e.g. in digit recognition to distinguish between the digits ‘6’ and ‘9’. This is sometimes referred to as 6-9-problem. Other properties of interest in invariant classification include discrimination, computational complexity, ease and speed of training, generalization ability, flexibility and the possibility of transformation retrieval. Note that discrimination is an important aspect here, as for instance a mapping of any feature vector to a constant value yields a perfectly invariant mapping, which of course is useless for classification. In some cases one may want to distinguish between global and local invariances, depending on the context and the given data, but this distinction can be reduced to the assumption of different transformations which are present. One trivial solution to the problem of invariance in pattern recognition is employing brute force. In this context this means to compare all the possible transformations of the patterns and extract the optimal match.

In the following, different methods to deal with known invariances are presented. The distinction between the approaches is somewhat arbitrary, for example one can regard normalization as a process of invariant feature extraction (normalized images are of course invariant with respect to the chosen transformations) or one can define an invariant distance measure as the distance between the normalized images. A further (equally arbitrary) distinction can be made concerning the time step the invariant process takes place. Normalization and feature extraction usually are performed before the actual classification process, whereas invariant distance measures and classifier combination are methods used in later steps of the classification procedure.

4.7.1 Normalization

With the term normalization one usually refers to the construction of a canonical representation for each pattern with respect to the regarded transformations, in which the considered transformations are eliminated. These representations can then be compared without the influence of the differences of the transformations. For instance, to achieve invariance to additive illumination changes, it is sufficient to normalize all given images to have the same mean graylevel.

For example, we may use the following normalization procedure in order to achieve invariance with respect to rotation, translation, and scale for images (sometimes referred to as RST-invariance):

- compute the center of gravity and translate it to the origin (translation invariance)
- normalize for average radius (scale invariance)
- rotate such that the direction of maximum variance coincides with the x -axis (rotation invariance)

A drawback of such normalization procedures is the fact that they often depend on a segmentation of the objects contained in an image and that they may be very sensitive to noise. Furthermore, moment-based normalization steps (as the computation of the center of gravity in the above procedure) only yield meaningful results if the intra-class variability of the objects regarded is considerably small.

4.7.2 Invariant features

If we want to obtain a classification procedure that is invariant with respect to certain transformations, one approach is to calculate a set of features from the pattern that is not affected by these transformations but still contains all information relevant for classification. An overview over the use of invariant features is given in [Burkhardt & Siggelkow 01]. Such an ideal system of invariants should be able to distinguish with arbitrary precision between any two patterns not in the same orbit under \mathcal{G} . But in practice such a complete set of invariants for a given group unfortunately does not always exist.

Although the concept of invariant features is theoretically sound, no general strategy for feature extraction is known. In many cases, however, it is possible to trace back this redundancy to the action of a group \mathcal{G} . Yet, it must be considered that an invariant feature space does not exist for all kinds of transformation. In [Schulz-Mirbach 92] the nonexistence of such a space for the dilations and any group containing the dilations as a subgroup is proven. This can be illustrated by regarding the scaling transformation. If features are required to be invariant with respect to scaling, all images should lead to the same features as a single point.

A number of performance aspects for invariant features is presented in [Burkhardt & Fenske⁺ 92], which include completeness (ability to discriminate between all possible images), robustness (tolerate deterministic and stochastic errors), continuity (clustering, metric), and computational complexity.

A common problem of invariant feature extraction methods is that in many cases a significant part of the information contained in the original images is lost. For example, in most Fourier-based methods, the phase information of the Fourier spectrum is discarded. Using the invariant moments as proposed by [Hu 62], all the information contained in the regarded object is reduced to seven real-valued moments, which obviously implies a considerable loss of information. Thus, it is not guaranteed that invariant features are discriminative features at the same time. As an example, the mapping of each image to a constant value results in a perfectly invariant, yet at the same time completely useless feature. We find a similar statement in [Breuel 93]: “The driving principle behind feature-based recognition is data reduction. That is, variation that is inessential to the identity of a character is to be discarded during the feature extraction process. Unfortunately, it seems that any part of the input data describing a character can be essential for determining its identity and no information should be discarded before the classification process.” In spite of these problems, invariant features may be very useful for data that allows a completely invariant representation, as for example images of red blood cells [Dahmen & Hektor⁺ 00].

Complete invariant features that only eliminate the degrees of freedom of the respective transformations are desirable. Examples of such features are translation invariant features based on monomials [Burkhardt & Fenske⁺ 92]. In practical situations, these approaches are often complemented by other methods, as for example histograms of rotation invariant integrated monomials for content-based image retrieval [Siggelkow 02, Deselaers & Keysers⁺ 04b] or the use of partial invariance for handwritten digit recognition [Haasdonk & Halawani⁺ 04].

Another class of invariant features is based on the Fourier transform (FT). Here, the invariance of the squared magnitude of the FT spectrum (power spectrum) under translation of the pattern is the property that is used.

If more than only translation-invariance is desired, this can be achieved using variants

of the FT, e.g. the Mellin transform. This a Fourier transform evaluated over an exponential scale, which is invariant under the scaling transformation. If aspects of the Fourier and Mellin transform are combined in two steps along with a transformation to polar coordinates of an image (resulting in a circular Fourier, radial Mellin transform), one can achieve invariance with respect to rotation, scaling, and translation simultaneously.

Another application of the FT is the extraction of Fourier descriptors for binary images. They can be obtained by parameterizing the object boundary and analyzing the Fourier transform of the resulting boundary function [Burkhardt & Fenske⁺ 92]. These Fourier descriptors are invariant with respect to translation and rotation and can be enhanced for affine invariance. The Fourier descriptors for shape can be generalized to grayscale objects (given a separation from the background) by not only parameterizing the object boundary but also the grayvalue distribution. [Burkhardt & Fenske⁺ 92] state that the performance of affine invariant gray level Fourier descriptors is superior to that of affine invariant moments, because they are less sensitive to noise in real applications.

Algebraic invariants, or moment invariants, that can be used as invariant features are obtained by taking quotients and powers of moments. A moment is a weighted sum of the pattern x_{ij} over the whole input field, with weights equal to some polynomial in i, j .

[Hu 62] proposed seven polynomial combinations of basic moments as features that are translation, scale, and also rotation invariant. These invariant features seem to work well only on binarized patterns in the absence of distortion and noise, which is reflected in extremely poor performance for example on the USPS digit recognition task [Perrey 00]. This is consistent with the observation that regular moments are highly noise-sensitive [Wood 96].

4.7.3 Invariant distance measures

While normalization and the extraction of invariant features aim at the elimination of the transformations present in the data before the actual classification process, invariance can also be incorporated directly into the classifier. This can be done by using invariant distance measures, which is the approach used in large parts of this work. An invariant distance measure would ideally have the property that the distance between two patterns is always equal to the minimum distance between the ‘best matching’ transformed instances of those patterns.

The most common distance measure encountered is the (squared) Euclidean distance, which is also inherent in the normal distribution (with the identity matrix as covariance matrix). There are a variety of other distance measures used in pattern recognition, like the Minkowski metrics or l_p -norms. As a similarity measure also the the dot or scalar product of two vectors is often used, which is also called the correlation. The dot product $x^T \cdot \mu = \sum_{d=1}^D x_d \mu_d$ is related to the angle θ between two vectors by $\cos \theta = x^T \cdot \mu / (\|x\| \|\mu\|)$ where the cosine of the angle is also called normalized dot product. A connection to the Euclidean distance is given by the relation

$$\|x - \mu\|^2 = \|x\|^2 - 2x^T \cdot \mu + \|\mu\|^2. \quad (4.22)$$

If the two vectors have a fixed length (e.g. norm 1), this shows the direct equivalence of Euclidean distance and dot product.

The relation (4.22) is also helpful for pattern matching in larger images, that is, if the best fitting match x (a part of a larger image) to a reference μ is desired. In that case the Euclidean distance can be decomposed into a term independent of the position ($\|\mu\|^2$), a

term easily calculated for each position of the smaller template in the image ($\|x\|^2$, only the sum of squares of the border needs to be considered when stepping through the image) and a convolution ($x^T \cdot \mu$) which can be efficiently calculated using the fast Fourier transform. This approach was used for the speed-up of the holistic search as described in Chapter 8 and in [Keysers & Motter⁺ 03]

These basic distance measures can be very sensitive with respect to variations in the images like affine transformations. The main part of this thesis is dedicated to the effect of different models of variability on the recognition performance of distance-based classifiers.

We will use different invariant distance measures of the general form:

$$d(x, x') = \min_{\alpha \in \mathcal{M}} \left\{ d'(x, t(x', \alpha)) \right\} \quad (4.23)$$

where d' is a simple distance measure, usually the Euclidean distance. Then the invariant distance measure d results from a minimization over all considered transformations t with parameters α in a set of allowable parameters \mathcal{M} , which e.g. restricts the rotation angle, or the maximum displacement of a pixel position. Note that this formulation is asymmetric. We usually take the point of view that the transformations are applied to the reference image and thus the test image (or observation) should be best ‘explained’ by the transformed reference image. This is not a hard restriction, though: e.g. the tangent distance is often applied with tangent vectors calculated from both the test and the reference image.

In some cases, it is beneficial to add a penalty or regularization term to the distance function that depends on the deviation of the transformation from the identity mapping such that

$$d(x, x') = \min_{\alpha \in \mathcal{M}} \left\{ d'(x, t(x', \alpha)) + f(\alpha) \right\}. \quad (4.24)$$

Here, for simple formulations of t , we could have e.g. $f(\alpha) = c \cdot \|\alpha\|^2$. However, in almost all of the experiments we performed, no improvement in classification performance was obtained by using such a regularization term.

Note that the minimization involved in the distance computation may be of different degrees of difficulty. For example the minimization in the context of the tangent distance (as discussed in Chapter 5) is computationally simple, whereas the complexities for the pixel-to-pixel deformation models that are discussed in Chapter 6 can range from very low to exponential depending on the model used.

4.7.4 Virtual data

For most pattern recognition applications, the size of the training data set has a strong influence on the classification results. It seems obvious that a classifier, in particular one based on statistics, should perform better with an increasing number of training samples. This idea is often found to be true in experimental results.

Unfortunately, in many applications training data is scarce and in most settings it involves a certain cost to acquire new data, especially if it needs to be labeled with the correct class labels manually.

One approach to alleviate this problem is to use a priori or domain knowledge for invariance, for example represented by tangent vectors, which will be introduced in detail in Section 5.2 along with the concept of tangent distance. On this aspect, Simard and colleagues comment that “using tangent distance or tangent propagation is like having a much larger database” [Simard & Le Cun⁺ 98a].

With respect to the impact of the training set size in optical character recognition, one can find the following statement in [Smith & Bourgoin⁺ 94]: “For every tenfold increase in database size the error rate is cut by half or more though the performance seems to be leveling off slightly for the larger database sizes.” And furthermore “there is good reason to believe that performance will continue to improve as the training database grows even larger. In some ways, this is an obvious result. If the database is large enough it will eventually saturate the space of all possible bitmaps and the system could only fall short of perfect performance due to errors or noise in the training database.” From this the authors even deduce that “researchers might better spend their time collecting data than writing code.” (We believe that this statement is not true, as you can infer e.g. from the evolution of achieved error rates on the MNIST corpus (compare Section 3.1.2), which have reached new minima every few years. You might argue that this evolution is due to some extent to effects of training on the testing data and that you are likely to achieve lower error rates even if you only randomly change your classifiers, but we believe that there is also a better understanding of the classification techniques and the image variability involved.)

Naturally, larger training set sizes bring along higher computational demands. For example, a kernel density or nearest neighbor classifier with the naive implementation has a runtime in testing proportional to the number of training samples used. This effect can be reduced by various techniques as the use of hierarchical distance measures, pruning, or approximate nearest neighbor search techniques. Also for support vector machines, large amounts of training data induce computational demands, especially high in the training phase, but also the number of support vectors determined will usually grow with the number of training samples.

One simple and common way to increase the amount of training data is to use the available domain knowledge to create so-called virtual training data by applying transformations that do not change the class membership to the given training data and enlarge the training set with these new examples. The brute force method to do so would be to produce all possible transformations in order to achieve complete invariance, but this is not feasible in most practical settings. Therefore, one usually restricts the multiplication of the data to a few variants of the transformations. The creation of virtual data is a fairly general technique that is well-established for different pattern recognition tasks [Niyogi & Girosi⁺ 98]. Two possible drawbacks of this method are that the user must choose the magnitude of the transformation parameters and the number of instances to be generated beforehand and that the generated data is highly correlated [Simard & Victorri⁺ 92].

Note that invariant distance measures can be viewed as involving an implicit creation of virtual data: the invariant distance measure chooses the best fitting pattern among all transformed training patterns $t(x', \alpha)$; the same result could be obtained by explicitly enriching the training set with all these transformations, which may be (possibly infinitely) many.

One example of the successful application of virtual training data are the experiments performed by Drucker and colleagues on the MNIST handwritten digits task. Using extensive virtual training data creation (multiplying the available 60,000 images to some million training examples) in combination with a boosted artificial neural net, the authors reported the error rate of 0.7% on that particular task in 1993 [Drucker & Schapire⁺ 93], which was the best known error rate for a long time.

Virtual data has also led to very good results for support vector machines [DeCoste & Schölkopf 02]. In an SVM, often only the resulting support vectors are mul-

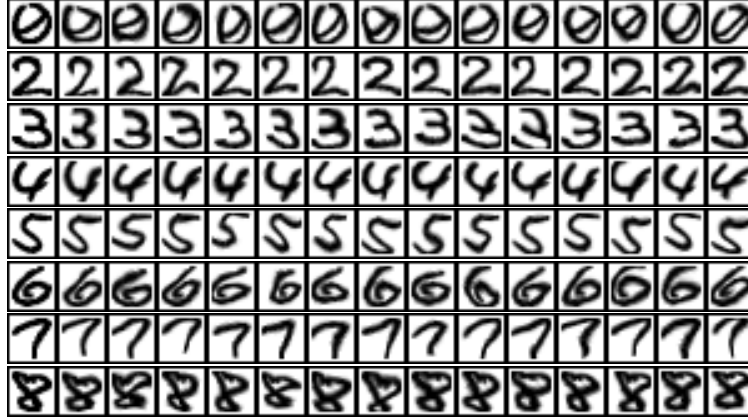


Figure 4.6: Examples of 2D smoothed random distortions of USPS images following the ideas presented in [Simard & Steinkraus⁺ 03].

multiplied to yield so-called virtual support vectors, which then are used to re-train the SVM. This approach has the benefit that only training samples that lie close to the decision boundary are multiplied. These virtual samples are more likely to be beneficial to the training process than those that are in regions of the input space where the classification is very clear for one class.

The creation of virtual training data may also be done implicitly as for example during the training cycles of a neural network. Simard used this idea of implicit data multiplication by a factor of about 1,000 in [Simard & Steinkraus⁺ 03] and achieved the currently best error rate on one of the standard handwritten character recognition benchmark data sets, the MNIST data set as described in Section 3.1.2. The method proceeds as follows: in each training cycle for a neural network, each training sample is fed into the network in a slightly distorted way, based on a random two-dimensional distortion. This distortion is calculated by choosing a random 2-dimensional displacement vector in a suitable interval for each pixel and then smoothing the resulting displacement field by using a Gaussian filter. Example images that result from this technique are shown in Figure 4.6. The interesting aspect of this technique is that it is not necessary to store all 60 million distorted training samples but still each of these implicit training items influences the training.

As it is possible to use the knowledge about invariance for the training data by creating virtual data this is the case for the test data as well. Here the interpretation is not as straightforward as for the training data case, but inspired by methods for classifier combination [Kittler 98] one can arrive at the following solution called virtual test sample method (VTS) [Dahmen & Keysers⁺ 01a]:

When classifying a given pattern, transformed versions of the pattern are generated (using the a priori knowledge about the data) and independently classified by the same classifier. The overall decision is then obtained by combining the individual results using the sum rule, which usually outperforms other classifier combination schemes [Kittler 98], i.e.

$$p(x|k) = \sum_{\alpha} p(x, \alpha|k) \quad (4.25)$$

where α denotes the used transformation parameters. Note that in the case of VTS, the motivation for the sum rule differs from that proposed by Kittler. To justify the sum rule in the

case of using multiple classifiers to classify a single test pattern, he assumed that the posterior probabilities computed by the respective classifiers do not differ much from the prior probabilities. In contrast to this, using multiple test patterns and a single classifier, the sum rule simply follows from the fact that the transformations considered are mutually exclusive, if we assume that the respective prior probabilities are equal (e.g. the prior probability for a right shift should be the same as for a left shift, which seems reasonable). More detailed discussions of this method can be found in [Dahmen & Keysers⁺ 00a, Dahmen & Keysers⁺ 01a]. A similar approach has also been presented in [Ha & Bunke 97].

5 Gaussian and related models

Die Theorie zieht die Praxis an, wie der Magnet das Eisen.

– C.F. Gauß

Gaussian models are widely used in various applications of pattern recognition. They have a number of interesting properties, for example they are relatively simple to handle analytically, they are computationally easy to use both regarding the estimation of their parameters and the evaluation of the density, and they have the added advantage of playing a major role in the central limit theorem.

In this chapter we describe — after a short general introduction — the two concepts that are related to Gaussian models and are subjects of this document: the tangent distance approach for modeling image variability and the maximum entropy framework for discriminative training.

5.1 Gaussian models

One effective method to describe class conditional probability densities is to assume that the data is distributed according to a linear mixture of multivariate Gaussian distributions, thus allowing multimodal distributions. This assumption does not impose any restriction on the modeling power, since the resulting Gaussian mixture density (GMD) can approximate any density function with arbitrary precision.

First, consider a unimodal Gaussian distribution

$$\begin{aligned} p(x|k) &= \mathcal{N}(x|\mu_k, \Sigma_k) \\ &= \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} \exp \left(-\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \right) \end{aligned} \quad (5.1)$$

with the according maximum likelihood estimates

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N_k} x_{nk}, \quad (5.2)$$

$$\hat{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N_k} (x_{nk} - \mu_k)(x_{nk} - \mu_k)^T. \quad (5.3)$$

Since in most of the experiments the setting $\Sigma_k = \sigma_k^2 I$ was used, here the maximum likelihood estimator for σ_k^2 is given (as one easily verifies by differentiating the log-likelihood)

$$\hat{\sigma}_k^2 = \frac{1}{DN_k} \sum_{n=1}^{N_k} (x_{nk} - \mu_k)^T (x_{nk} - \mu_k) = \frac{1}{D} \text{trace}(\hat{\Sigma}_k) \quad (5.4)$$

This means that the estimator equals the arithmetic mean of the diagonal elements of the empirical covariance matrix. One reason justifying this approach is that in an appearance-based approach all variables correspond to measurements of the same kind, i.e. pixel intensities. Now, a Gaussian mixture is a linear combination of Gaussians

$$p(x|k) = \sum_{i=1}^{I_k} c_{ki} \cdot \mathcal{N}(x|\mu_{ki}, \Sigma_{ki}), \quad c_{ki} > 0, \quad \sum_{i=1}^{I_k} c_{ki} = 1 \quad (5.5)$$

with mixture weights c_{ki} and component densities $\mathcal{N}(x|\mu_{ki}, \Sigma_{ki})$. The maximum likelihood estimators for the parameters cannot be determined explicitly any more, but there exists an iterative algorithm which can be used for this purpose called EM-algorithm (Expectation-Maximization) [Dempster & Laird⁺ 77, Ney 99]. A classifier using GMD is also called radial basis function classifier (RBF) and produces the same type of decision rules as a support vector machine with Gaussian kernel [Vapnik 98]. For a short introduction to image object recognition using GMD see [Dahmen & Beulen⁺ 00]. The use of GMD based classifiers has proven to be effective for image object recognition in various settings, and is a widely used method in speech recognition.

The description of the class conditional probability density function by kernel densities (KD) (also called Parzen windows or Parzen densities) can be seen as extreme case of GMD where each reference serves as a center of its ‘own’ (usually, but not necessarily normal) distribution. That is, each training sample x_n defines a single density (e.g. Gaussian $\mathcal{N}(x|x_n, \Sigma_{x_n})$ with covariance matrix Σ_{x_n}), that is the sample itself is interpreted as mean vector. Although in general Σ_{x_n} may depend on the sample x_n , it is usually chosen to be equal for all considered x_n . Thus, kernel densities are the extreme case of a mixture density model. The method belongs to the class of so-called nonparametric procedures (as for example the nearest neighbor method) that can be used without assuming that the form of the underlying density is known [Duda & Hart 73, p. 85]. Since all the training patterns are kept and compared to the observation, this method is also closely related to the (k -)nearest neighbor (NN) technique. A good informal description in the context of digit recognition can be found in [Hinton & Dayan⁺ 97]. Since each training sample defines its own density center, the covariance matrix must be chosen by other methods than maximum likelihood, because ML estimation leads to zero variances in this case. Often, the neighborhoods are assumed to take the same ellipsoidal shape as the underlying distribution [Fukunaga 90, p.267].

Starting with a kernel function $\varphi_k(x)$ that is itself a probability density function usually centered around zero (possibly depending on the class k), the kernel density approximation of the class conditional probability density function is

$$p(x|k) = \frac{1}{N_k} \sum_{n=1}^{N_k} \varphi_k(x - x_{nk}) \quad (5.6)$$

and using a Gaussian kernel this becomes

$$\begin{aligned} p(x|k) &= \frac{1}{N_k} \sum_{n=1}^{N_k} \mathcal{N}(x|x_{nk}, \Sigma_{x_{nk}}) \\ &= \frac{1}{N_k} \sum_{n=1}^{N_k} \frac{1}{\sqrt{(2\pi)^D |\Sigma_{x_{nk}}|}} \exp \left(-\frac{1}{2} (x - x_{nk})^T \Sigma_{x_{nk}}^{-1} (x - x_{nk}) \right) \end{aligned} \quad (5.7)$$

Inserting this into Bayes' rule together with the ML-estimation $p(k) = \frac{N_k}{N}$ yields the decision rule

$$\begin{aligned}
 r(x) &= \operatorname{argmax}_k \{p(k)p(x|k)\} \\
 &= \operatorname{argmax}_k \left\{ \frac{N_k}{N} \frac{1}{N_k} \sum_{n=1}^{N_k} \frac{1}{\sqrt{(2\pi)^D |\Sigma_{x_{nk}}|}} \exp \left(-\frac{1}{2} (x - x_{nk})^T \Sigma_{x_{nk}}^{-1} (x - x_{nk}) \right) \right\} \\
 &= \operatorname{argmax}_k \left\{ \frac{1}{\sqrt{|\Sigma_{x_{nk}}|}} \sum_{n=1}^{N_k} \exp \left(-\frac{1}{2} d_{nk}(x_{nk}, x) \right) \right\}
 \end{aligned} \tag{5.8}$$

where $d_{nk}(x_{nk}, x)$ represents the Mahalanobis distance of x to x_{nk} . Now the KD-based classifier can be used with different other distance measures. For example the squared Euclidean distance could be used or distance measures that are invariant with respect to some transformation as e.g. the tangent distance which will be introduced in the following. Consider for example the setting of $\Sigma_{x_{nk}} = \sigma_k^2 I$, which was used in the experiments with Euclidean distance. Then the decision rule becomes

$$r(x) = \operatorname{argmax}_k \left\{ \frac{1}{\sigma_k^D} \sum_{n=1}^{N_k} \exp \left(-\frac{1}{2\sigma_k^2} \|x - x_{nk}\|^2 \right) \right\} \tag{5.9}$$

To compensate for the fact that variances are often underestimated using a limited amount of training data, one can multiply the variances σ_k by a constant factor greater than one.

5.2 Tangent vectors in Gaussian models

In this section, we study the use of linear representations for the variability in a statistical framework. This approach is based on the use of tangent vectors, that have been successfully used for the recognition of handwritten digits within distance-based classifiers. We start with a discussion of the so-called tangent distance.

The use of tangent vectors in this context has its origin in the regularization of neural networks for character recognition [Simard & Le Cun⁺ 92, Simard & Victorri⁺ 92]. Here, the tangent vectors denote those directions of the input space, along which the output of the neural network should not change. This idea was then successfully used in distance-based classifiers [Simard & Le Cun⁺ 93]. The basic idea behind the tangent distance (TD) is to approximate image transformations in feature space by a linear subspace, and then use the distance between these subspaces as invariant distance measure for classification.

The tangent distance is discussed in this chapter, because it can easily be integrated into Gaussian models and a theoretical analysis shows that it is equivalent to using a covariance matrix of special structure in a Gaussian model. For a discussion of other ways to structure the covariance matrix see [Dahmen & Keyers⁺ 00b].

We have observed that using the tangent distance not only improves the recognition performance in the original setting of handwritten digit recognition as shown by Simard and colleagues, but we can report various additional results:

- very good results for the classification of medical images [Keyers & Dahmen⁺ 03],
- strong improvements for the application to gesture and sign language recognition [Zahedi & Keyers⁺ 05a, Dreuw & Keyers⁺ 05],

- integration of the tangent vector approach into a statistical framework and
- derivation of the resulting distribution analytically [Keysers & Dahmen⁺ 00a], which allows the
- estimation of tangent vectors from training data that is useful in image classification but can also be applied to other domains like speech recognition, showing significant improvements [Keysers & Macherey⁺ 04].

We integrate the tangent method into a statistical framework for classification analytically and practically. The resulting consistent statistical framework derived allows us to use tangent vectors that are the derivatives of specified transformations as well as to determine the tangent vectors from the given training data in terms of a maximum likelihood estimation. This facilitates the use of the tangent vector method for tasks where meaningful transformations of the feature vectors are not easily obtained, like e.g. the transformation effects on the feature vectors of a speech signal used in automatic speech recognition. The use of tangent vectors in this framework improves classification results on real-world pattern recognition tasks.

5.2.1 Overview of tangent distance

In some application areas, transformations that leave the class membership unchanged are known a priori, like e.g. small affine transformations in the case of character recognition. We want the classifier to be invariant with respect to these transformations. Let $\tilde{x}(\alpha)$ denote a transformation of x depending on a parameter L -tuple $\alpha \in \mathbb{R}^L$. The set of all transformed patterns typically has highly nonlinear characteristics in pattern space. To obtain a tractable representation, we consider a linear approximation of the transformation using a Taylor expansion around $\alpha = 0$:

$$\tilde{x}(\alpha) = x + \sum_{l=1}^L \alpha_l v_l + \sum_{l=1}^L \mathcal{O}(\alpha_l^2),$$

and then neglecting the terms of second order and higher. Here, the partial derivatives of the transformation \tilde{x} with respect to the parameters α_l ($l = 1, \dots, L$) are called the tangent vectors $v_l = \partial \tilde{x}(\alpha) / \partial \alpha_l|_{\alpha_l=0}$, as they span the tangent subspace of the set of all transformed patterns at the point x . These derivatives can be efficiently calculated e.g. using differences between slightly transformed patterns [Keysers 00]. The software used to determine the tangent vectors and to calculate the tangent distance is available for download¹. Figure 5.1 shows examples using an image of a handwritten digit and approximations of transformations. Figure 5.2 shows examples using a medical radiograph and approximations of transformations. These examples illustrate the advantage of using the linear approximation, as the depicted patterns (and those which result from a combination of the transformations) all lie in the same linear subspace and can therefore be represented by one prototype and the corresponding tangent vectors. We thus have a concise representation of the variability, where the degree of transformation is represented by a parameter vector α . This representation can be integrated into the probabilistic framework as presented in the following section.

To determine the tangent vectors $\{v_l\}$, we can use three alternatives:

¹<http://www-i6.informatik.rwth-aachen.de/~keysers/td/>



Figure 5.1: Example of first-order approximation of affine transformations and line thickness for an image from the USPS data. Left to right: original image, \pm horizontal translation, \pm vertical translation, \pm rotation, \pm scale, \pm axis deformation, \pm diagonal deformation, \pm line thickness.

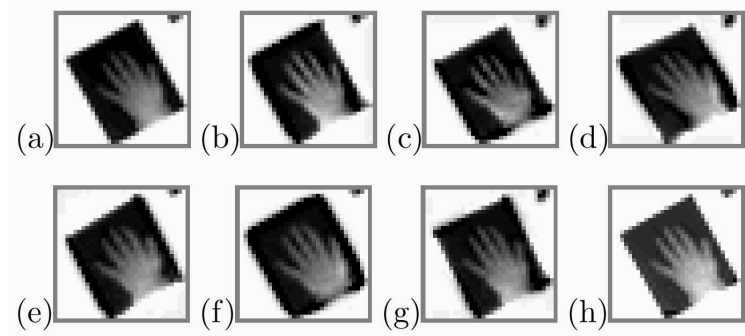


Figure 5.2: Example images generated using linear approximations of affine transforms and image brightness for an image from the IRMA data. (a) original image, (b) left shift, (c) down shift (d) hyperbolic diagonal deformation, (e) hyperbolic axis deformation, (f) scaling, (g) right rotation, (h) increased brightness

- (v1) compute the derivatives for the reference vector μ ($v_l = \partial \tilde{\mu}(\alpha) / \partial \alpha_l |_{\alpha_l=0}$),
- (v2) compute the derivatives for the observation vector x ($v_l = \partial \tilde{x}(\alpha) / \partial \alpha_l |_{\alpha_l=0}$),
- (v3) estimate the derivatives from the training data,

where (v1) and (v2) require prior knowledge about the transformations. How to apply (v3) shall become clear with the integration of the following statistical framework, which facilitates the estimation as a maximum likelihood solution.

We briefly present the distance formulation that results from the tangent vector approach, i.e. the original tangent distance. A more in-depth discussion can be found e.g. in [Keysers 00]. Assume we want to calculate a distance between x and μ that is invariant with respect to a transformation t . The distance-based approach starts with the observation, that ideally we would want to regard the distance between the sets $\mathcal{M}_x = \{t(x, \alpha)\}$ and $\mathcal{M}_\mu = \{t(\mu, \alpha)\}$, which are manifolds in pattern space. Since this distance is hard to compute in most cases, we approximate the manifolds by their tangent subspaces $\hat{\mathcal{M}}_x$ and/or $\hat{\mathcal{M}}_\mu$ that is spanned by the tangent vectors. The distance between these subspaces is easily computed as the solution to a linear least squares problem. Figure 5.3 illustrates these concepts for the one-sided tangent distance with tangent vectors computed for the reference.

So far, we have not discussed the computational complexity of the tangent method. Due to the structure of the resulting model, the computational cost of the distance calculation is increased approximately by a factor of $(L + 1)$, in comparison with the model that corresponds to the Euclidean distance or to Mahalanobis distance with diagonal covariance

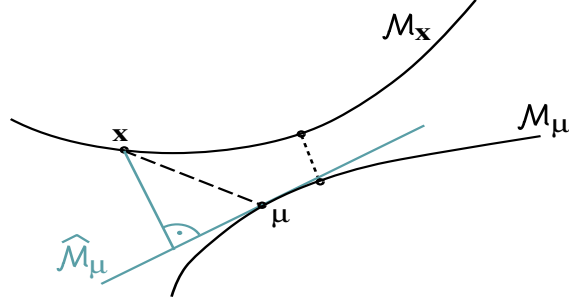


Figure 5.3: Illustration of the Euclidean distance between an observation x and a reference μ (dashed line) in comparison to the distance between the corresponding manifolds (dotted line). The tangent approximation of the manifold of the reference and the corresponding (one-sided) tangent distance is depicted by the light gray lines [Keysers & Macherey⁺ 01].

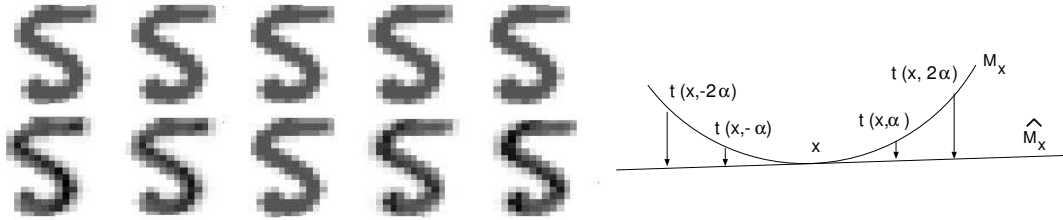


Figure 5.4: Images obtained by shifting a digit and by finding the closest point in the tangent space, original image in the middle. The upper row shows the shifted images with the closest tangent approximation in the lower row. Schematic illustration on the right. The transformation t is a horizontal shift here and α corresponds to the displacement of one pixel

matrices. If full covariance matrices are used, the tangent vector approach does not increase the computational complexity independent of L , because the structure can be incorporated into the covariance matrix.

Some examples of the linear approximation in comparison to the transformation approximated are given in Figure 5.4, which shows images of the digit ‘5’ obtained by shifting the original image and finding the closest corresponding image in the tangent subspace for translation. On the right a schematic illustration is given. One can see that the approximated image corresponds well to the shifted image for shifts with a displacement of one pixel (second and fourth column), but the linear tangent subspace cannot describe larger shifts well (see outer columns, the images are almost identical to the ones obtained for one pixel shifts). Note that this result corresponds to the resolution of the difference operator that was used, which is a slightly extended Sobel operator [Keysers 00] and thus tuned to differences that occur on the scale of about one pixel in each direction.

5.2.2 Integration into the probabilistic framework

In adaptive pattern recognition, the distribution models are assumed to depend on an unknown adaptation parameter vector α , e.g. for rotation and scaling in image recognition. The Bayesian approach to adaptation consists in integrating out the unknown parameter, which is possible in this context [Keysers & Macherey⁺ 04]. We consider the case where the observations x have a Gaussian distribution with expectation μ_k and covariance matrix Σ . The extension to Gaussian mixtures or kernel densities is straightforward using maximum approximation or the expectation-maximization algorithm. The starting point is the integration

$$p(x|k) = \int p(x, \alpha|k) d\alpha = \int p(\alpha|k) \cdot p(x|k, \alpha) d\alpha \stackrel{\text{model}}{=} \int p(\alpha) \cdot p(x|k, \alpha) d\alpha,$$

where the distribution of the adaptation parameter set α is assumed to be independent of k . This distribution is assumed to be Gaussian with zero mean and covariance matrix equal to a multiple of the identity matrix:

$$p(\alpha) = \mathcal{N}(\alpha|0, \gamma^2 I), \quad (5.10)$$

where γ is a hyper-parameter describing the standard deviation of the transformation parameters. The distribution of x is assumed to be Gaussian for these considerations to simplify the analytical derivation. This assumption does not imply a loss of generality as the expectation-maximization algorithm allows us to transfer the results to Gaussian mixtures or kernels, which can model arbitrarily complex distributions and are successfully used in different applications (e.g. being the standard in speech recognition).

The distribution of class k is modified for adaptation based on the first-order approximation of the transformation given by the tangent vectors $\{v_{kl}\}$:

$$\begin{aligned} p(x|k, \alpha) &= \mathcal{N}(x|\tilde{\mu}_k(\alpha), \Sigma) \\ \tilde{\mu}_k(\alpha) &= \mu_k + \sum_{l=1}^L \alpha_l v_{kl} \\ v_{kl}^T \Sigma^{-1} v_{km} &= \delta_{lm} \end{aligned} \quad (5.11)$$

where $\delta_{lm} := 1$ if $l = m$ and 0 otherwise, denotes the Kronecker delta. To simplify the mathematical representation, the tangent vectors are assumed to be orthonormal with respect to the global covariance matrix Σ . This does not imply a loss of generality as only the spanned subspace determines the variation modeled and it is always possible to achieve this condition using e.g. a singular value decomposition. It is then possible to perform the integration (5.10) analytically by combining the exponents of the Gaussian density functions into one term of quadratic order in α using (5.10) and (5.11) and transforming this into one Gaussian density function [Keysers & Dahmen⁺ 00a]. The detailed calculation is presented below. As result, we obtain the exact closed-form solution for the probability density function of the observations:

$$\begin{aligned} p(x|k) &= \mathcal{N}(x|\mu_k, \tilde{\Sigma}_k) \\ \tilde{\Sigma}_k &:= \Sigma + \gamma^2 \sum_{l=1}^L v_{kl} v_{kl}^T \\ \tilde{\Sigma}_k^{-1} &= \Sigma^{-1} - \frac{1}{1 + \frac{1}{\gamma^2}} \cdot \Sigma^{-1} \sum_{l=1}^L v_{kl} v_{kl}^T \Sigma^{-1} \end{aligned} \quad (5.12)$$

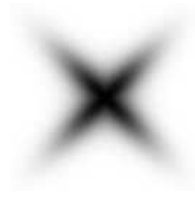


Figure 5.5: The resulting density for case (v2) in a 2D example with $\Sigma = I$, $\gamma = 2$, $L = 1$ and $v_1 = \frac{1}{\|x\|} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x$.

Thus, the incorporation of tangent vectors only affects the covariance matrix which can be interpreted as imposing a structure on Σ [Keyzers & Dahmen⁺ 00a]. Note that this result does not hold for the case (v2) above, using the derivatives of the observation. In this case, the resulting distribution is not necessarily Gaussian. Figure 5.5 shows an example of a resulting density that is not Gaussian. Note furthermore that $\det(\tilde{\Sigma}_k) = (1 + \gamma^2)^L \cdot \det(\Sigma)$ (cp. [Fukunaga 90, pp. 38ff.]) which is independent of the tangent vectors and can therefore be dropped in the maximum likelihood estimation (Section 5.2.3).

To view the results in terms of distances, consider the exponent in $\mathcal{N}(x|\mu_k, \tilde{\Sigma}_k)$ with a covariance matrix $\Sigma = \sigma^2 I$ which is assumed to be white except for a constant factor. This is the case for example after application of a global whitening transform of the data. We furthermore assume $\gamma \rightarrow \infty$ here, which lets the factor $\frac{1}{1+\frac{1}{\gamma^2}}$ approach one:

$$\begin{aligned} (x - \mu_k)^T \tilde{\Sigma}_k^{-1} (x - \mu_k) &= (x - \mu_k)^T \Sigma^{-1} (x - \mu_k) - \sum_{l=1}^L [(x - \mu_k)^T \Sigma^{-1} v_{kl}]^2 \\ &= \frac{1}{\sigma^2} \left[(x - \mu_k)^T (x - \mu_k) - \sum_{l=1}^L [(x - \mu_k)^T v_{kl}]^2 \right] \end{aligned} \quad (5.13)$$

The resulting exponent (5.13) turns out to be a modified Euclidean distance. It shows that variations along the directions of the tangent vectors are not (or less) important for classification. Note that the exponent leads to the conventional Mahalanobis distance for $\gamma \rightarrow 0$ and to the tangent distance for $\gamma \rightarrow \infty$.

Although the results presented here may not be new to researchers well familiar with linear Gaussian models, we include the following detailed calculations showing how to arrive at the final distribution model (5.12) starting from the basic tangent model and the assumptions (5.10) and (5.11). Since we only consider one class here, we suppress the class index k .

Starting with

$$\begin{aligned} p(x) &= \int p(x, \alpha) d\alpha = \int p(\alpha) \cdot p(x|\alpha) d\alpha \\ &= \int \mathcal{N}(\alpha|0, \gamma^2 I) \cdot \mathcal{N}(x|\mu + \sum_{l=1}^L \alpha_l v_l, \Sigma) d\alpha \end{aligned}$$

it suffices to consider the joint exponent

$$\begin{aligned}
 & \frac{1}{\gamma^2} \sum_l \alpha_l^2 + (\mu + \sum_l \alpha_l v_l - x)^T \Sigma^{-1} (\mu + \sum_l \alpha_l v_l - x) \\
 &= \frac{1}{\gamma^2} \sum_l \alpha_l^2 + (x - \mu)^T \Sigma^{-1} (x - \mu) + \sum_{l,l'} \alpha_l \alpha_{l'} \underbrace{v_l^T \Sigma^{-1} v_{l'}}_{=\delta_{ll'}} - 2(x - \mu)^T \Sigma^{-1} \sum_l \alpha_l v_l \\
 &= (x - \mu)^T \Sigma^{-1} (x - \mu) + \left(1 + \frac{1}{\gamma^2}\right) \sum_l \alpha_l^2 - 2(x - \mu)^T \Sigma^{-1} \sum_l \alpha_l v_l \\
 &= (x - \mu)^T \Sigma^{-1} (x - \mu) + \left(1 + \frac{1}{\gamma^2}\right) \sum_l \left[\alpha_l - \frac{(x - \mu)^T \Sigma^{-1} v_l}{1 + \frac{1}{\gamma^2}} \right]^2 \\
 &\quad - \frac{1}{1 + \frac{1}{\gamma^2}} \cdot \sum_l [(x - \mu)^T \Sigma^{-1} v_l]^2
 \end{aligned}$$

Then, integration over α yields (except for a constant factor) the exponent

$$\begin{aligned}
 & (x - \mu)^T \Sigma^{-1} (x - \mu) - \frac{1}{1 + \frac{1}{\gamma^2}} \cdot \sum_l [(x - \mu)^T \Sigma^{-1} v_l]^2 \\
 &=: (x - \mu)^T \tilde{\Sigma}^{-1} (x - \mu) \\
 &\text{with } \tilde{\Sigma}^{-1} := \Sigma^{-1} - \frac{1}{1 + \frac{1}{\gamma^2}} \cdot \Sigma^{-1} \sum_{l=1}^L v_l v_l^T \Sigma^{-1}
 \end{aligned}$$

The distribution is obtained by re-normalization:

$$p(x) = \mathcal{N}(x | \mu, \tilde{\Sigma}) \quad \text{where} \quad \tilde{\Sigma} = \Sigma + \gamma^2 \sum_{l=1}^L v_l v_l^T$$

Exactly the same result with a similar calculation is also obtained by using maximum approximation, i.e. by only considering the optimal value of α . The relation $\tilde{\Sigma} \cdot \tilde{\Sigma}^{-1} = I$ is easily checked using the orthonormality of the tangent vectors with respect to Σ .

5.2.3 Estimation of tangent vectors

To relax the constraint that the transformations must be known a priori, the tangent vectors can be estimated from the training data. This estimation can be formulated as a maximum likelihood approach within the presented framework. Let the training data be given by $x_{kn}, n = 1, \dots, N_k$ training patterns of $k = 1, \dots, K$ classes. We consider the single Gaussian model (5.12) with known class means μ_k and global covariance matrix Σ . For the derivation, we assume that the number L of tangent vectors is known.

We consider the log-likelihood as a function of the unknown tangent vectors $\{v_{kl}\}$:

$$\begin{aligned}
 F(\{v_{kl}\}) &:= \sum_{k=1}^K \sum_{n=1}^{N_k} \log \mathcal{N}(x_{nk} | \mu_k, \tilde{\Sigma}_k) \\
 &= \frac{1}{1 + \frac{1}{\gamma^2}} \cdot \sum_{k=1}^K \sum_{l=1}^L v_{kl}^T \Sigma^{-1} S_k \Sigma^{-1} v_{kl} + \text{const}
 \end{aligned} \tag{5.14}$$

with the class dependent scatter matrix

$$S_k = \sum_{n=1}^{N_k} (x_{nk} - \mu_k)(x_{nk} - \mu_k)^T.$$

Taking into account the constraints of orthonormality of the tangent vectors with respect to Σ^{-1} , we obtain the following result (cp. [Fukunaga 90, pp. 400ff.]): the class specific tangent vectors $\{v_{kl}\}$ maximizing (5.14) have to be chosen such that the vectors $\{\Sigma^{-1/2}v_{kl}\}$ are the eigenvectors with the largest corresponding eigenvalues of the matrix $\Sigma^{-1/2}S_k(\Sigma^{-1/2})^T$ (the dominant eigenvectors or principal components).

Using this model is equivalent to performing a global whitening transformation of the feature space (i.e. right-multiplication by $\Sigma^{-1/2}$ of all data) and then using the L principal components as tangent vectors for each class. This reduces the effect of those directions of class specific variability that contribute the most to the variance.

In summary, the use of estimated tangent vectors in Gaussian models consists of the following steps for each class k :

- compute the empirical mean vector μ_k
- compute the scatter matrix S_k
- compute $\{\Sigma^{-1/2}v_{kl}\}$ as eigenvectors with largest eigenvalues of $\Sigma^{-1/2}S_k(\Sigma^{-1/2})^T$

5.2.4 Discussion

The presented tangent model is related to previous work in two fields: On the one hand, tangent vectors have been used in distance-based classifiers, where the resulting distance measure is called tangent distance. On the other hand, the resulting distributions take the form of linear Gaussian models.

Tangent distance has been successfully applied in image object recognition during the last years [Keysers & Macherey⁺ 01, Simard & Le Cun⁺ 93, Simard & Le Cun⁺ 98b] and also has been included in textbooks [Bishop 95, pp. 320ff.], [Duda & Hart⁺ 01b, pp. 188ff.], [Hastie & Tibshirani⁺ 01, pp. 423ff.] as it combines intuitive understanding and effective modeling of variability, leading to reduction of classification errors.

Note that tangent distance can also be used as a distance within other classifiers, for example within support vector machines [Haasdonk & Keysers 02]. Similarly, tangent distance can be embedded into different algorithms that rely on distances to determine the dissimilarity of patterns, for example into an editing approach that reduces a specific training data set [Paredes & Vidal⁺ 02].

It might be interesting to further investigate the possibility of integrating the tangent vector approach into other classifiers like e.g. polynomial classifiers. As these also use a matrix representation of the variability in the data, it is analytically easy to integrate the variability described by tangent vectors into polynomial classifiers. This can be done by regarding the result of assuming a probability distribution along the tangent directions as in the probabilistic interpretation of tangent distance. Then, we can determine the expected value of the matrices used within the polynomial classifier and derive a closed form solution that is similar to the resulting covariance matrix for the tangent vectors in the Gaussian model. First informal experiments did not lead to significant improvements following this approach on a handwritten digit classification task.

The subject of linear subspaces for pattern classification is treated in different contexts with different names, including principal component analysis or Karhunen-Loève transform, factor analysis, sensible principal component analysis [Roweis 98], local principal component analysis [Kambhatla & Leen 97], tangent distance [Simard & Le Cun⁺ 93], constraint tangent distance [Schwenk & Milgram 96], nearest feature line method [Li & Lu 99], manifold learning [Bregler & Omohundro 95], mixture of linear experts [Hinton & Revow⁺ 95], locally linear models [Hinton & Dayan⁺ 97], Eigenfaces [Turk & Pentland 91], Fisher-faces [Belhumeur & Hespanha⁺ 97], etc.

Recently, [Roweis & Ghahramani 99] presented a unified view of linear Gaussian models including (sensible) principal component analysis, factor analysis and mixtures of Gaussians with the respective expectation-maximization algorithms. Here, we connect the use of tangent vectors to these models and describe a framework suitable for classification. The main addition is the treatment of the global noise covariance which is identical in the class-specific models, implying different restrictions on the covariance matrices. We consider this connection between tangent vectors and linear Gaussian models to be important, because the use of tangent vectors improves results on different classification tasks.

In the resulting model (5.12), the parameters α can be regarded as latent variables and it is therefore related to sensible principal component analysis [Roweis & Ghahramani 99] and probabilistic principal component analysis [Tipping & Bishop 99]. For the limiting case $\Sigma = I$, a similar result to the one presented here was derived in [Hastie & Simard 98]. The presented model with locally estimated scatter matrices is adaptive to specific local variability and therefore similar to the model presented in [Hastie & Tibshirani 96]. Note that the presented model assigns to the subspace components a weight γ which may differ from the corresponding eigenvalue. This is a main difference to subspace approximations to the full covariance matrix based on eigenvalue decomposition like e.g. [Meinicke & Ritter 99]. In the experiments, this weight was chosen to be larger than the eigenvalues (cp. Figure 5.6 (a)). In [Hastie & Simard⁺ 95] the authors present learning of prototypes in a distance-based formulation, where the solution is obtained by iterative optimization, using two-sided tangent distance. Some connections between tangent distance and linear models are already pointed out in [Hinton & Dayan⁺ 97], but here the authors report that they “found that the inclusion of tangent vectors did not substantially improve the performance.”

The maximum likelihood estimation of the tangent vectors seems to resemble conventional principal component analysis, which minimizes the reconstruction error. But here the projection vectors are chosen separately for each class. Furthermore, the model (5.12) disregards the specific variability of the patterns when determining the distance or the log-likelihood, respectively. That is, the tangent vectors span the subspace with *least* importance in the distance calculation here. In the limiting case of $\gamma \rightarrow \infty$, the effect is a class-dependent dimensionality reduction.

Note that the probabilistic interpretation of tangent distance can be used for a more reliable estimation of the parameters of a basic distribution by implicitly enriching the training set with infinitely many transformed patterns [Dahmen & Keyzers⁺ 01b].

Note also that there is substantial recent work on problems related to that of determining the number of tangent vectors L automatically [Bishop 99, Bishop & Winn 00, Everson & Roberts 00, Minka 00], which can alternatively be achieved using cross-validation.

The restrictions on the orthonormality of the tangent vectors with respect to the covariance matrix Σ (5.11) along with the restriction on the covariance structure of α (5.10) in fact do constitute a loss of generality. This restricts the subspace variability to be spher-

ical, which is a restriction similar to common restrictions imposed on covariance matrices in different pattern recognition aspects. Nevertheless, the extension to the general case is possible [Roweis & Ghahramani 99].

The maximum likelihood estimation of the tangent vectors yields results similar to conventional principal component analysis, where the principal components are chosen to minimize the reconstruction error. One main difference is that these components are chosen separately for each class here. Furthermore, the model (5.12) disregards the specific variability of the patterns when determining the distance or the log-likelihood, respectively. That is, the tangent vectors span the subspace with least importance in the distance calculation here. This difference is treated with the names ‘distance in feature space’ and ‘distance from feature space’, respectively, in [Moghaddam & Pentland 97], where the latter is pointed out to be more appropriate for classification. In the limiting case of $\gamma \rightarrow \infty$, the effect is a class-dependent dimensionality reduction. This principle bears a similarity to the idea of (linear) discriminant analysis (LDA, [Duda & Hart⁺ 01b, pp. 117ff.], [Hastie & Tibshirani⁺ 01, pp. 84ff.]), where a global transformation is sought, that minimizes inter-class distances with respect to intra-class distances. Here, this transformation is chosen separately for each class.

There exists an interesting connection between the concept of the manifold in the context of tangent distance and the concept of intrinsic dimensionality as presented by [Fukunaga 90, pp.280ff]. The following quotation expresses this connection: “Whenever we are confronted with high-dimensional data sets, it is usually advantageous for us to discover or impose some structure on the data. Therefore, we might assume that the generation of the data is governed by a certain number of underlying parameters. The minimum number of parameters required to account for the observed properties of the data, n_e , is called the *intrinsic* or *effective dimensionality* of the data sets, or, equivalently, the data generating process. [...] The geometric interpretation is that the entire data set lies on a topological hyper-surface of n_e -dimension.” The author goes on to state that a measure of the dimensionality is the number of dominant eigenvectors of the covariance matrix and that these form the effective subspace, but that this approach is only suitable for linear surfaces. For nonlinear surfaces the intrinsic dimensionality can be determined locally, similar to the local linearization of a nonlinear function. Therefore it is also called local dimensionality. This is closely connected to the considerations of methods to estimate the directions of variation are derived based on dominant eigenvectors of the (local) covariance matrix.

In [Kim & Kim⁺ 02] the authors evaluate a PCA mixture model for handwritten digit recognition and observe that using the reconstruction error outperforms using the membership value. Since the reconstruction error is identical to the tangent distance of estimated tangents this result underlines the adequacy of the tangent distance method using estimated tangents.

While the previously described methods are based on a single pattern from which a description of the manifold is derived, some methods have been proposed for description of the manifold from a set of patterns. For example Hinton and colleagues use a blended linear approximation to the manifold fitted with an EM-based algorithm [Hinton & Dayan⁺ 97]. This method can be viewed as a mixture density implementation of the approaches proposed in Section 5.2.2. A similar approach is taken by [Bregler & Omohundro 95], interpolating between specified images with manifold learning by inducing a smooth nonlinear constraint manifold from a set of examples from the manifold, while linear interpolation just averages the two pictures. The underlying principle of the approach is basic, i.e. a mixture model of local linear patches is fit to the data by clustering, PCA and EM. The final step of inter-

Table 5.1: Results for the USPS corpus for Gaussian models, error rates [%].

classifier	total # of dens.	without LDA $x \in \mathbb{R}^{16 \times 16}$		with LDA $x \in \mathbb{R}^{39}$
		$\Sigma = \sigma^2 I$	$\text{diag}(\Sigma)$	$\text{diag}(\Sigma)$
single Gaussian	10	18.6	19.5	12.8
Gaussian mixtures	$\sim 1,000$		8.0	6.7
+ virtual data	$\sim 10,000$		6.0	3.4
nearest neighbor	$\sim 7,300$	5.6	6.8	7.0
+ virtual data	$\sim 65,700$	4.3	5.3	3.6
Gaussian kernels	$\sim 7,300$	5.5	6.3	6.5
+ virtual data	$\sim 65,700$	4.2	5.1	3.4

polation is then achieved by (different methods) of projection into the manifold. Recently, [Fitzgibbon & Zisserman 03] presented the use of image manifolds for clustering the face images appearing in a movie, thus enabling the system to automatically summarize the cast of a film by invariant clustering of faces.

Already [Short & Fukunaga 81] present a local distance measure for nearest neighbor classification that adapts to the distribution of patterns in a local neighborhood of the test pattern. The authors argue that those direction in the local neighborhood that contribute strongest to a change in the density $p(k|x)$ should have the strongest weight in the optimal distance measure for nearest neighbor classification. This is related to the local estimation of tangent vectors and their use in the tangent distance for nearest neighbor (or kernel density) classification in the following way: The use of the estimated tangent vectors amounts to ignoring those directions in pattern space in which the local density changes least (those with the largest local variance components). Therefore, only the remaining directions with a stronger change in density are used for classification. This resembles the results presented by [Short & Fukunaga 81] qualitatively.

5.2.5 Experimental results

We performed experiments using the statistical approach in combination with tangent vectors for real-world classification tasks from two different domains, namely image object recognition and automatic speech recognition. Here we only present the results for the image-related parts. the results for the automatic speech recognition experiments can be found in [Macherey & Keysers⁺ 01, Keysers & Macherey⁺ 04]. We present results for the recognition of handwritten digits and of images of red blood cells.

USPS data

Results for the domain of image object recognition were obtained on the well-known USPS task (Section 3.1.1). Reported recognition error rates for this database are summarized in Table 3.2 on page 10. The comparably small training set makes the use of invariance methods especially helpful.

Table 5.1 shows a summary of results on the USPS database using the Gaussian models [Keysers & Macherey⁺ 04]. The non-Gaussian data is modeled well by the use of mixture

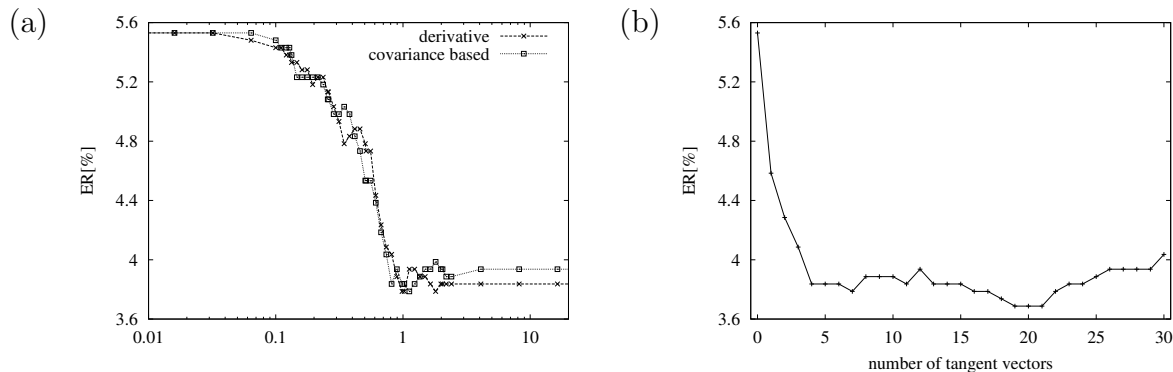


Figure 5.6: Error rate (ER) (a) as a function of tangent vector parameter standard deviation γ for $L = 7$ derivative and covariance-based tangent vectors, and (b) as a function of the number of covariance-based tangent vectors (both for USPS, kernel densities).

and kernel density models. Because of the good performance of the Gaussian kernel density model (4.3), all following experiments on USPS were based on this model, using $\Sigma = \alpha\sigma^2 I$.

In the experiments with Gaussian kernels and estimated covariance-based tangent vectors, we computed the local scatter matrix S_{kn} using the nearest neighbors of the same class for each training vector. The experiments showed that using about 30 neighbors provides a sufficient estimate of the local covariance structure.

Figure 5.6 (a) shows the error rate with respect to γ for derivative tangents of the references and the covariance based estimation of tangents using $L = 7$ each. It can be seen that, on this data, no significant improvement can be obtained by restricting the value of the hyper-parameter γ , which controls the possible values of the transformation vectors α . This effect is most likely due to the high dimensionality of the feature space in combination with a fixed range for meaningful feature values ('black' to 'white'). The strong non-linearity of the manifolds then makes undesired solutions with high values of the parameters α very unlikely. The following experiments were therefore performed using $\gamma \rightarrow \infty$.

Table 5.2 shows the effect of the different types of tangent vectors that can be calculated using the derivative of affine transformations and line thickness as proposed in [Simard & Le Cun⁺ 93]. Additional transformations (e.g. projective) were tested but did not yield any improvements over the set of the original seven vectors. We can observe that the effect of removing the tangent vector for line thickness is the largest. Almost identical results were obtained when judging the effect of using only one tangent vector, i.e. the effect was strongest for the line thickness tangent.

Another interesting factor with effect on the error rate is the number of tangent vectors used in the covariance-based approach. This dependency is depicted in Figure 5.6 (b). It can be observed that the first four tangent vectors lead to the largest reduction in error rate, while a minimum was reached for 20 tangent vectors per kernel density. The strong decrease in error rate shows that the presented method can be effectively used to learn the class specific variability on this dataset.

The effect of the three estimation methods (v1) to (v3) is indicated in Table 5.3. The results show that on this data, the covariance-based estimation of the tangent vectors (v3) lead to the same error rate as the use of the derivatives for μ (v1) using more parameters

Table 5.2: Effect of the type of tangent vector for derivatives of the observation (USPS corpus, kernel densities, error rates [%]).

type of tangent vector	ER [%]
without tangent vectors	5.5
all 7 tangent vectors	3.3
without line thickness	4.0
without vertical translation	3.8
without horizontal translation	3.6
without rotation	3.6
without scaling	3.6
without diagonal deformation	3.6
without axis deformation	3.4

Table 5.3: USPS error rates using kernel densities and tangent distance.

method	ER[%]
kernel densities baseline	5.5
+ tangent vectors (v3), $L = 7$	3.8
(v3), $L = 12$	3.7
(v1), $L = 7$	3.7
(v2), $L = 7$	3.3
(v1)+(v2), $L = 14$	3.0
+ virtual test data	2.6
+ virtual training data	2.4
+ classifier combination	2.2

(20 instead of 7 tangent vectors) or a slightly higher error rate using the same number of parameters. This result seems quite remarkable, as the estimation from the data alone was able to improve results as much as the use of additional domain knowledge about the data (invariance with respect to small affine transformations and line thickness). The use of derivatives of x (v2) and the combination of (v1) and (v2) lead to further improvements.

Table 5.3 also contains the results obtained using additional virtual data. The use of virtual test and training data (by shifting the images 1 pixel into 8 directions, keeping training and test set separated) increased the performance of the classifier further to 2.4%. The best result obtained using the presented approaches was with a combination of different classifiers (with varying parameters), where different test results were combined using the sum rule. This reduced the error rate further to 2.2%, although this last result must be considered as an effect of ‘training on the testing data’, as the best ensemble was chosen on the basis of the test results.

Interestingly, when using a single Gaussian density, i.e. one reference per class, the error rate on the USPS corpus could be reduced from 18.6% to 5.5% using $L = 12$ covariance-based tangent vectors. Using only $L = 7$ tangent vectors, the result of 6.4% outperforms the use of the derivative, here with 11.8% error rate. Here, the means of the single densities are very blurred, which is a disadvantage for the derivative tangent vectors.

Using all 7,291 training patterns in a kernel density-based classifier, the result obtained without tangent model was the same as for a single density model with 12 estimated tangents

Table 5.4: Summary of Results for the RBC data

method	ER[%]
1-NN	24.4
+ histogram equal.	21.4
+ kernel densities	19.6
+ tangent distance	17.8
+ virtual data	16.3

(5.5%). In this case, the single densities with estimated tangent subspace obtain the same result as the kernel density approach using about 50 times fewer parameters.

The application of the estimation of tangent vectors to automatic speech recognition is presented in more detail in [Macherey & Keysers⁺ 01, Keysers & Macherey⁺ 04]. Significant improvements on the SieTill corpus of spoken German digit strings could be obtained.

Red blood cells data

We also evaluated the performance of the invariant statistical classifier with respect to the classification of red blood cells (RBC) [Keysers & Dahmen⁺ 01a]. In this case, the classifier is based on distance functions invariant to affine transformations and additive brightness and on kernel densities.

Table 5.4 shows a summary of the obtained results to be compared to the results presented in Table 3.11. We started our experiments regarding the appearance-based method with a nearest neighbor (1-NN) classifier, which is often used as a baseline result. We found that applying a two-bin histogram equalization to the data during classification improved the result from 24.4% to 21.4%, diminishing different background graylevel intensities in the data. By using a kernel density-based Bayesian classifier the error rate could be further reduced to 19.6%. Finally, we added two ingredients to improve transformation tolerance, i.e. tangent distance and virtual data as described above. These led to the best observed error rate for the appearance-based approach of 16.3% error. The tangents used in these experiments were six for the affine transformations and one for additive brightness offsets.

By using a simple reject rule (reject, if the negative log-likelihood of the second best class is not at least $r\%$ larger than that of the best class) we could reduce the error rate to 15.5% at 1.4% reject for $r = 10$ and to 14.5% at 3.9% reject for $r = 12$. This is slightly inferior to the result of 13.6% error at 2.4% reject as reported in [Dahmen & Hektor⁺ 00]. We also performed a number of further experiments, which did not lead to improved recognition rates. Among these was the use of image normalization with respect to rotation and the use of gradient information as additional features.

The obtained results show that on this specific task – RBC classification – the appearance-based approach as described here does not lead to the best possible performance. This observation can be explained by regarding the different types of variability present in the data: the RBC images appear in rotations of all possible angles during sedimentation, while handwritten digits are only subject to small rotations and some other transformations with comparatively small extent. Thus, the information loss inherent in the extraction of invariant features is tolerable for RBC images but not for images of digits, while tangent distance is able to model small transformations in OCR, but does not perform as well for larger transformations. Nevertheless it can be observed that the employment of tangent distance

and virtual data improves classification significantly. The presented method has the advantage that less parameters need to be chosen, suggesting better generalization properties. Nevertheless, the appearance-based approach using local patches as described in Chapter 7 in combination with tangent distance led to an even better error rate.

5.2.6 Offline handwriting recognition using tangent vectors

The offline recognition of continuous handwritten text is an important research topic. Typical applications include reading of bank checks or customer forms. Most approaches in this area use hidden Markov models to model the variability in writing direction. Here we will briefly discuss the possible use of tangent vectors in hidden Markov models for offline handwriting recognition [Toselli & Juan⁺ 04]. The tangent vectors can be used to model additional small vertical changes that are orthogonal to the writing direction. Robustness with respect to stroke vertical variability is achieved by integrating tangent vectors into the emission densities of the hidden Markov models. Experimental results are reported on a syntax-constrained interpretation task which show the effectiveness of the proposed approaches in [Toselli & Juan⁺ 04].

The method as presented in [Toselli & Juan⁺ 04] uses extensive preprocessing of the input images and elaborate treatment of ascenders and descenders. However, vertical shift variability remains difficult to model in left-to-right one-dimensional HMMs. As an additional effective method for coping with this problem we can use tangent vectors, which are especially suitable for integration into Gaussian models as discussed above.

Here, we want the character HMMs to be robust with respect to small vertical shifts. This can be achieved by applying the following procedure to each Gaussian density $\mathcal{N}(\vec{\mu}, \vec{\Sigma})$ of each mixture of the trained HMMs:

- calculate the tangent vector \vec{v} as the vertical derivative of the mean vector $\vec{\mu}$ and normalize to $\|\vec{v}\| = 1$;
- modify the covariance matrix $\vec{\Sigma}$ by setting $\vec{\Sigma} \leftarrow \vec{\Sigma} + \gamma^2 \vec{v} \vec{v}^T$, where the factor γ controls the variance along the tangent vector direction.

The increased variance in the direction of the tangent vectors leads to emission densities which assign higher probability to slightly transformed feature vectors. This has the effect that the resulting model is more robust with respect to this transformation, in this case with respect to vertical variability.

Experiments were performed on a data set of handwritten Spanish text of currency amounts [Toselli & Juan⁺ 04]. The tangent vectors were incorporated into the trained HMMs in order to increase robustness of the approach with respect to changing vertical shift within each word. Using this method, the WER was reduced from 5.8% to 5.0%, which is a relative improvement of about 14%. The digit error rate could be reduced from 4.6% to 4.1%, which corresponds to a relative improvement of about 10%. We can thus observe considerable improvements due to the use of tangent vectors in the Gaussian emission densities of the hidden Markov models.

Figure 5.7 shows examples of recognized handwritten words from the data set of handwritten Spanish currency amounts. The recognition was performed with and without the use of tangent vectors and we can observe that the use of tangent vectors in the emission densities effectively compensates for vertical variability as was expected.

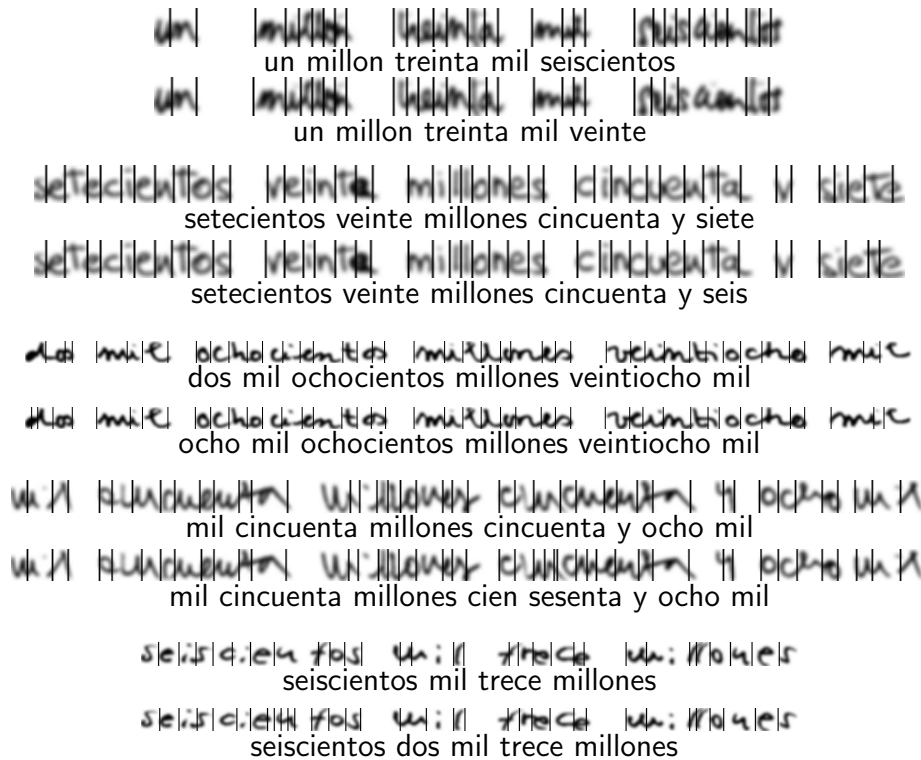


Figure 5.7: Examples of the application of tangent vectors for vertical shift in hidden Markov models for continuous text classification. Upper part of pair: with tangent vectors, lower part: without tangent vectors.

5.3 Maximum entropy models

In this section, we discuss the relation of Gaussian densities and statistical models for the class posterior probability based on the maximum entropy approach and log-linear models.

The principle of maximum entropy is a powerful framework that can be used to estimate class posterior probabilities for pattern recognition tasks. Here we show how this principle is related to the discriminative training of Gaussian mixture densities using the maximum mutual information criterion. This leads to a relaxation of the constraints on the covariance matrices to be positive (semi-)definite. Thus, we arrive at a conceptually simple model that allows us to estimate a large number of free parameters reliably. We compare the proposed method with other state-of-the-art approaches in experiments with the well-known US Postal Service handwritten digits recognition task.

The maximum entropy framework is based on principles applied in the natural sciences. It has been applied to the estimation of probability distributions [Jaynes 82] and to classification tasks such as natural language processing [Berger & Della Pietra⁺ 96] and text classification [Nigam & Lafferty⁺ 99].

The contributions of this part of the work are

- to show the relation between maximum entropy and Gaussian models,
- to present a framework that allows us to estimate a large number of parameters reliably, e.g. the entries of full class specific covariance matrices, and
- to show the applicability of the maximum entropy framework to image object recognition.

5.3.1 Introduction to the maximum entropy framework

An introduction to the criterion functions used in discriminative training is given in Section 4.4.2. The criterion used in the maximum entropy framework is often referred to as mutual information criterion in speech recognition, information theory and image object recognition [Dahmen & Schlüter⁺ 99, Normandin 96].

Consider a set of so-called feature functions $\{f_i\}, i = 1, \dots, I$ that are supposed to compute ‘useful’ information for classification:

$$f_i : \mathbb{R}^D \times \{1, \dots, K\} \longrightarrow \mathbb{R} : (x, k) \longmapsto f_i(x, k)$$

From the information in the training set, we can then compute the following numbers

$$F_i := \sum_n f_i(x_n, k_n) .$$

Now, the maximum entropy principle consists in maximizing

$$\max_{p(k|x)} \left\{ - \sum_n \sum_k p(k|x_n) \log p(k|x_n) \right\}$$

over all possible distributions with the requirements:

- *normalization constraint* for each observation x :

$$\sum_k p(k|x) = 1;$$

- *feature constraint* for each feature i :

$$\sum_n \sum_k p(k|x_n) f_i(x_n, k) = F_i.$$

It can be shown that the resulting distribution has the following log-linear or exponential functional form:

$$p_\Lambda(k|x) = \frac{\exp [\sum_i \lambda_i f_i(x, k)]}{\sum_{k'} \exp [\sum_i \lambda_i f_i(x, k')]}, \quad \Lambda = \{\lambda_i\}. \quad (5.15)$$

Interestingly, it can also be shown that the stated optimization problem is convex and has a unique global maximum. Furthermore, this unique solution is also the solution to the following dual problem: Maximize the log probability (4.8) on the training data using the model (5.15). Note that there may be more than one maximizing parameter set Λ , though. In this formulation of the problem, it is easier to see that there exists exactly one maximum, because (4.8) is a sum of convex functions and therefore also convex. A second desirable property of the discussed model is that effective algorithms are known that compute the global maximum of the log probability (4.8) given a training set. These algorithms fall into two categories: On the one hand, we have an algorithm known as generalized iterative scaling (GIS) [Darroch & Ratcliff 72] and related algorithms that can be proven to converge to the global maximum. On the other hand, due to the convex nature of the criterion (4.8), we can also use general optimization strategies as e.g. conjugate gradient

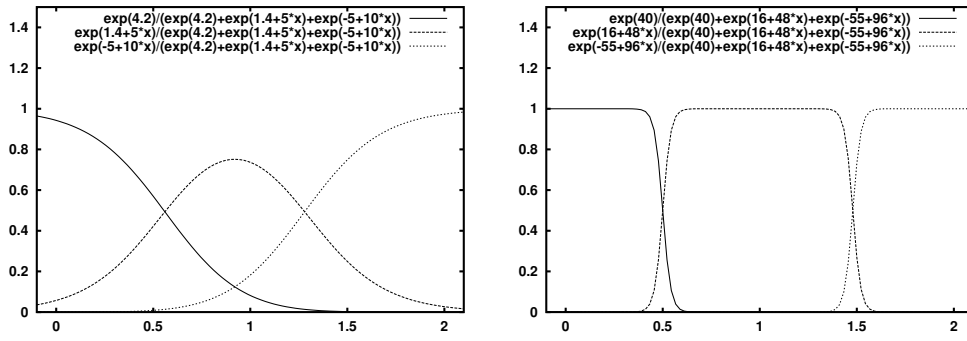


Figure 5.8: Posterior distributions using the log-linear or maximum entropy framework for the example problem. Left: after a few GIS iterations; right: after many GIS iterations.

methods [Press & Teukolsky⁺ 92, pp. 420ff.]. The resulting model (5.15) is also known as logistic regression for $K=2$ or as multi-class logistic regression for $K>2$.

The crucial problem in maximum entropy modeling is the choice of the appropriate feature functions $\{f_i\}$.

We illustrate the discussion of the maximum entropy framework with some very simple examples. Although these examples do not contribute anything new to the pattern recognition knowledge, they helped the author to better understand the principles behind the methods and are included in the hope the same may be true for some of the readers. The example consists of feature vectors $x \in \mathbb{R}$ coming from three classes:

$$p(k=0, x=0) = p(k=1, x=1) = p(k=2, x=2) = 1/3$$

We regard six feature functions in the example:

$$\begin{aligned} f_{k,2}(x, k') &= \delta(k, k') \\ f_{k,1}(x, k') &= \delta(k, k') x \end{aligned}$$

Figure 5.8 shows the resulting distributions after a few and many GIS iterations. We can observe that the speed of change of the posterior density at the class boundaries increases strongly.

The effect of too crisp posteriors can be counteracted by using (Gaussian) priors on the coefficients of the model. This has been tried in several informal experiments for this work, e.g. for the experiments using the USPS data, but the effects were non-conclusive, although other authors report improvements using such priors.

The second example also consists of feature vectors $x \in \mathbb{R}$, now from two classes. The distribution is depicted in Figure 5.9 along with the resulting posteriors for maximum likelihood and maximum entropy models. The first class has data points around 0 and 2, which are equally likely, the second class has data points around 3, and both classes have equal prior probabilities of $1/2$. The resulting posteriors for the maximum likelihood approach using Gaussians with pooled variance are given in red, and the posteriors resulting from the maximum entropy approach with the same first-order features as in the first example are given in blue. We can observe that the maximum likelihood decision boundary cuts through the second mode of the first class, while the discriminative maximum entropy approach shifts the decision boundary toward the second class, such that this does not happen. Obviously this example is contrived to show exactly this effect.

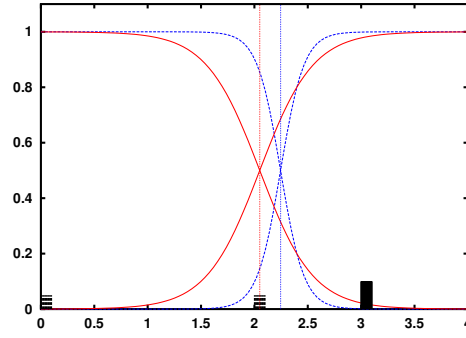


Figure 5.9: Posterior distributions for maximum likelihood (red) and maximum entropy training (blue) for the second example problem (two classes, distributions given as histogram boxes).

5.3.2 Connection between Gaussian and maximum entropy models

To discuss the relationship between the maximum entropy and the Gaussian case, we will regard Gaussian models for the class conditional distributions. The free parameters of these models are the class means μ_k and the class specific covariance matrices Σ_k . The conventional method for estimating these parameters is to maximize the log-likelihood on the training data, which yields the empirical mean and the empirical covariance matrix as solutions. Problems with this approach arise if the feature dimensionality is large with respect to the number of training samples. This is common e.g. in appearance-based image object recognition tasks, where each pixel value is considered a feature. The problems are that the large number of $K \cdot D \cdot (D + 1)/2$ parameters of the covariance matrices often cannot be estimated reliably using the usually small amount of training data available. Common methods for coping with this problem are to constrain the covariance matrices, e.g. to use diagonal covariance matrices, or to use pooling, i.e. to estimate only one covariance matrix Σ instead of K matrices.

We will now first derive the connection between the maximum entropy log-linear models and the Gaussian case. To do so, consider first-order feature functions for maximum entropy classification

$$\begin{aligned} f_{k,i}(x, k') &= \delta(k, k') x_i, \\ f_k(x, k') &= \delta(k, k'), \end{aligned}$$

where $\delta(k, k') := 1$ if $k = k'$, and 0 otherwise again denotes the Kronecker delta function. In the context of image recognition, we may call the functions $f_{k,i}$ appearance-based image features, as they represent the image pixel values. The duplication of the features for each class is necessary to distinguish the hypothesized classes. The functions f_k allow for a log-linear offset in the posterior probabilities. Now, using the properties of the Kronecker delta, the structure of the posterior probabilities becomes [Keysers & Och⁺ 02a]

$$\begin{aligned} p_\Lambda(k|x) &= \frac{\exp[\alpha_k + \sum \lambda_{k,i} x_i]}{\sum_{k'} \exp[\alpha_{k'} + \sum \lambda_{k',i} x_i]} \\ &= \frac{\exp[\alpha_k + \lambda_k^T x]}{\sum_{k'} \exp[\alpha_{k'} + \lambda_{k'}^T x]} \quad \Lambda = \{\lambda_{k,i}, \alpha_k\}, \end{aligned} \quad (5.16)$$

where α_k denotes the coefficient for the feature function f_k .

Now, consider a Gaussian model for $p(x|k)$ with pooled covariance matrix $\Sigma_k = \Sigma$. Using Bayes' rule, and the relation

$$\begin{aligned} \log N(x|\mu_k, \Sigma_k) &= -\frac{1}{2} \log \det(2\pi\Sigma_k) - \frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \\ &= -\frac{1}{2} \log \det(2\pi\Sigma_k) - \frac{1}{2} x^T \Sigma_k^{-1} x + \mu_k^T \Sigma_k^{-1} x - \frac{1}{2} \mu_k^T \Sigma_k^{-1} \mu_k, \end{aligned}$$

we can rewrite the class posterior probability (note that the terms that do not depend on the class k cancel in the fraction):

$$\begin{aligned} p(k|x) &= \frac{p(k) N(x|\mu_k, \Sigma)}{\sum_{k'} p(k') N(x|\mu_{k'}, \Sigma)} \\ &= \frac{\exp [(\log p(k) - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k) + (\mu_k^T \Sigma^{-1})x]}{\sum_{k'} \exp [(\log p(k') - \frac{1}{2} \mu_{k'}^T \Sigma^{-1} \mu_{k'}) + (\mu_{k'}^T \Sigma^{-1})x]} \\ &= \frac{\exp [\alpha_k + \lambda_k^T x]}{\sum_{k'} \exp [\alpha_{k'} + \lambda_{k'}^T x]} \end{aligned} \quad (5.17)$$

As result, we see that for unknown class priors $p(k)$ the resulting model (5.17) is identical to the maximum entropy model (5.16). We can conclude that the discriminative training criterion (4.8) for the Gaussian model with pooled covariance matrices results in exactly the same functional form as the maximum entropy model for first-order features. This allows us to use the well understood algorithms for maximum entropy estimation to estimate the parameters of a Gaussian model discriminatively.

Note that we can always find a Gaussian model with the same posterior distribution (but not necessarily the same prior probabilities) as given by a log-linear model. To do so, we need to determine μ_k and Σ such that the posterior distributions match those of the maximum entropy model with parameters α_1^K and λ_1^K . Therefore, we add $\alpha' := -\log \sum_k \exp(\alpha_k + \frac{1}{2} \lambda_k^T \lambda_k)$ to each α_k , which does not change the distribution but ensures that the resulting prior probabilities sum to unity. Then choose $\Sigma = I$, $\mu_k = \lambda_k$, and let $p(k) = \exp(\alpha_k + \frac{1}{2} \lambda_k^T \lambda_k + \alpha')$.

If we repeat the same argument as above for the case of Gaussian densities without pooling of the covariance matrices, we find that we can again establish a correspondence to a maximum entropy model:

$$\begin{aligned} p(k|x) &= \frac{p(k) N(x|\mu_k, \Sigma_k)}{\sum_{k'} p(k') N(x|\mu_{k'}, \Sigma_{k'})} \\ &= \frac{\exp [\alpha_k + \lambda_k^T x + x^T S_k x]}{\sum_{k'} \exp [\alpha_{k'} + \lambda_{k'}^T x + x^T S_{k'} x]} \end{aligned}$$

Here, the square matrix S_k corresponds to the negative of the inverse of the covariance matrix Σ_k . These parameters can be estimated using a maximum entropy model with the second-order feature functions

$$\begin{aligned} f_{k,i,j}(x, k') &= \delta(k, k') x_i x_j, \quad i \geq j, \\ f_{k,i}(x, k') &= \delta(k, k') x_i, \\ f_k(x, k') &= \delta(k, k'). \end{aligned}$$

One interesting consequence of using the corresponding maximum entropy model and estimation is that we implicitly relax the constraints on the covariance matrices to be positive

(semi-)definite. Therefore, the resulting model is not exactly equivalent to a Gaussian model. Note, though, that we can always add any matrix to all S_k without changing the distribution, as it cancels in the fraction. If we choose a multiple of the identity matrix with the multiple c larger than the smallest (negative) eigenvalue among all S_k , the resulting matrices $S'_k = S_k + cI$ will all be positive definite and thus Gaussian models can be found.

This result is in contrast to the approach taken in [Jaakkola & Meila⁺ 00], where the authors derive discriminative models for Gaussian densities based on priors of the parameters and the minimum relative entropy principle. Their solution results in discriminatively trained weights for the training data and therefore preserves the mentioned constraints.

5.3.3 Experimental results

We performed experiments on the USPS handwritten digits recognition task, as described along with known recognition rates from various methods in Section 3.1.1.

In a first experiment, we wanted to compare the result obtained with the maximum entropy framework that uses the MMI criterion to a direct implementation of that criterion as presented in [Dahmen & Schlüter⁺ 99]. Dahmen and colleagues report an error rate of 10.2% using Gaussian single densities, virtual data, 39-dimensional features based on an LDA after creation of pseudo-classes, and discriminative training with the MMI criterion. Using the same features and first-order feature functions, we obtained an error rate of 9.8%, which is almost identical to the result of Dahmen and colleagues, which was to be expected as the same functional form of the discriminant and the same criterion are used.

In most of the experiments performed we obtained better results using ‘feature normalization’. This means that we enforced for each observation during training and testing that the sum of all feature values is equal to one by scaling the feature values appropriately. Thus, we obtain new feature functions $\{\tilde{f}_i\}$:

$$\forall x, k, i : \tilde{f}_i(x, k) = \left(\sum_{i'} f_{i'}(x, k) \right)^{-1} \cdot f_i(x, k)$$

In the following, we only report results obtained using feature normalization. The parameters were trained using generalized iterative scaling [Darroch & Ratcliff 72].

Table 5.5 shows the main results obtained in comparison to other approaches along with the number of free parameters of the respective models.

The error rates show that we can already gain recognition accuracy by using the maximum entropy framework to only estimate the pooled covariance matrix of a Gaussian model, while fixing the mean vectors to their maximum likelihood values. Taking into account the class information in training using the maximum entropy framework increases the recognition accuracy for first-order features from 18.6% to 8.2% error rate using fewer parameters.

Figure 5.10 shows visualizations of the first-order feature weights that result from the training process. In most images it is possible to visually determine the image class that was trained. We can also observe that because of the discriminative nature of the training process, we obtain high weights in those areas of the image that are more likely to contain a dark pixel for that class than for other classes. For example, for the weights corresponding to the class ‘2’, we observe large weights in the upper left, lower left, and lower right corner. This is because on the average for image of handwritten twos, these regions are more likely to contain dark pixels than for the other classes because of the form of a handwritten ‘2’.

Table 5.5: Overview of the results obtained on the USPS corpus using maximum entropy modeling in comparison to other models (error rates, [%]). ML: maximum likelihood, MMI: maximum mutual information, *: with pooled diagonal covariance matrix.

model	training criterion	# parameters	ER[%]
Gaussian model*	ML	2 816	18.6
	Σ : MMI, μ_k : ML	2 816	14.2
maximum entropy, first-order features	MMI	2 570	8.2
second-order features	MMI	331 530	5.7
third-order features	MMI	759 090	5.6
nearest neighbor classifier		1 866 496	5.6

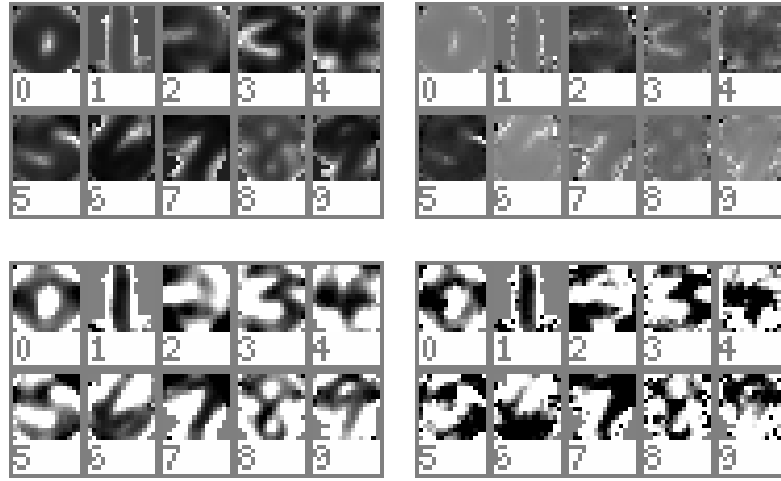


Figure 5.10: Visualization of maximum entropy coefficients for first order features on the USPS data set. Dark pixels correspond to large weights for the indicated class. Left: after a few iterations; right: after 300 iterations; top row: normalized over the complete range of the coefficients; bottom row: cut to the range of $[-5,5]$ and then normalized.

It can be observed that the maximum entropy models perform better for second-order features than for first-order features. This is in contrast to the experience gained with maximum likelihood estimation of Gaussian densities, where best results were obtained using pooled diagonal covariance matrices [Dahmen & Keysers⁺ 01b]. Note for example that the maximum likelihood estimation of class specific diagonal covariance matrices already imposes problems for the USPS data, because in some of the classes some of the dimensions have zero variance in the training data. This can be overcome e.g. by using interpolation with the identity matrix, but the maximum entropy framework offers an effective way to overcome these problems.

Using the equivalent of a full class specific covariance matrix, i.e. second-order features, the error rate of a ‘pseudo Gaussian’ model with 5.7% error rate approaches that of a nearest neighbor classifier, which has more than five times as many parameters.

Observing the reduction in error rate while going from first to second-order features, we

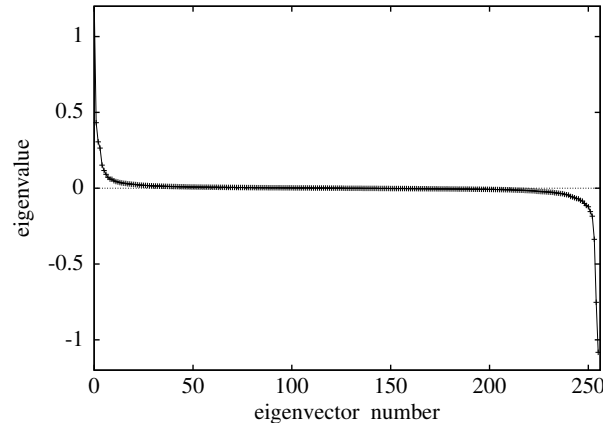


Figure 5.11: Eigenvalue distribution for the ‘covariance matrix’ of the class ‘5’, estimated using the maximum entropy approach.

also performed experiments using additional third-order features. In that case the feature functions are of the form

$$\begin{aligned}
 f_{k,i,j,m}(x, k') &= \delta(k, k') x_i x_j x_m, \quad i \geq j \geq m, \\
 f_{k,i,j}(x, k') &= \delta(k, k') x_i x_j, \quad i \geq j, \\
 f_{k,i}(x, k') &= \delta(k, k') x_i, \\
 f_k(x, k') &= \delta(k, k').
 \end{aligned}$$

The resulting error rate of 5.6% is only marginally better than for the second-order model and thus probably not worth the considerable additional computational effort. Note that the model now does not correspond to any Gaussian model any more. For more discussion, refer to the following Section 5.3.4.

Figure 5.11 shows the eigenvalues of the ‘covariance matrix’ of this ‘pseudo Gaussian’ model for the class ‘5’ ordered by size. It can be observed that about half of the eigenvalues are positive, while the other half is negative. The distribution of the negative eigenvalues seems to match the distribution of the positive eigenvalues. We can conclude that besides the typical important eigenvectors with large positive eigenvalues there are also important eigenvectors with large negative eigenvalues in this discriminative context. This means that the relaxation of the constraint on the covariance matrix to be positive (semi-)definite leads to discriminative models that are not exactly Gaussian any more.

Table 5.6 shows some error rates obtained by using various features and first-order maximum entropy training (thanks go to A. Hegerath for help with the experiments). The filter values of the used filters, which include the Sobel-filter, the Laplacian, and various Gabor filters, are squared before being used as features. If the filter values are taken directly, they have no effect on the resulting maximum entropy distribution (except for possible slight numerical differences, compare the discussion in the context of the maximum entropy linear discriminant analysis on page 92). This was also verified experimentally. We can observe that the Sobel features and the Gabor features lead to very good results. Using Sobel-filters in four directions, we can reach the same error rate as with the second-order features but using far fewer parameters (the second-order approach uses about $\frac{1}{2}D^2$ features, the

Table 5.6: Error rates for the maximum entropy method and different first-order features on the USPS and MNIST data sets.

features	USPS	MNIST
gray values	9.0	8.2
Sobel	6.1	3.7
gray values + Sobel	6.1	3.6
Laplace	20.9	18.6
gray values + Laplace	8.3	7.3
Sobel (4 directions)	5.6	3.0
Gabor (2 phases, 2 frequencies)	7.5	3.8
Gabor (4 phases, 3 frequencies)	7.0	3.3
gray + Gabor (4,3) + Sobel (4 dir.)	5.4	2.5

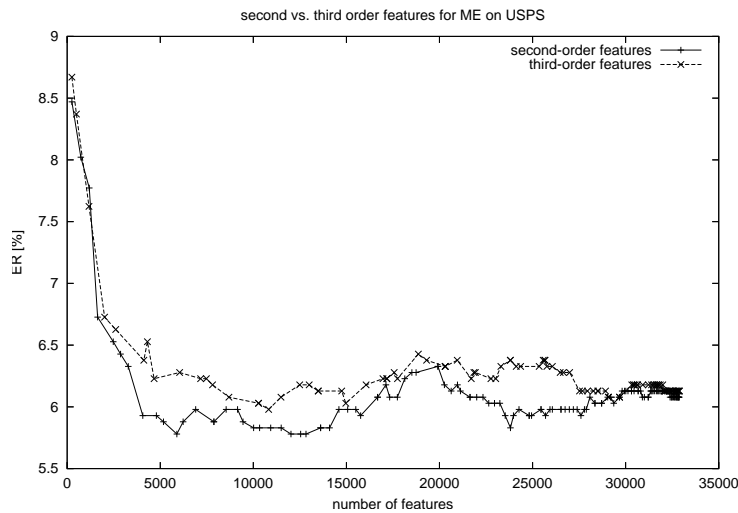


Figure 5.12: Comparison of second- and third-order features for the maximum entropy classifier on the USPS data.

Sobel-filter first-order approach only about $4D$). A system that uses these features and a first-order maximum entropy classifier is very fast and gives very reliable estimates of the posterior probabilities at the same time. It would therefore be ideal for high-speed classifiers as a pre-classifier using a rejection rule similar to ‘if $\max_k p(k|x) > p_{\min}$ classify to $\arg \max_k p(k|x)$, otherwise use a more sophisticated (and slower) classifier’. In informal experiments on the USPS corpus, we could achieve 0% error at 10.3% rejection rate using this rule with $p_{\min} = 0.5$ and the gray values as features. With the Sobel features and $p_{\min} = 0.45$ we could obtain an error rate of 0% at a rejection rate as low as 7.3%. This means that a computationally more expensive classifier would only have to be used in one tenth of the cases or less, lowering the overall computation times considerably.

5.3.4 Higher order feature functions

Figure 5.12 shows the error rate of the maximum entropy classifier on the USPS data set for varying numbers of features, comparing second-order features to third order features. The results were produced using as stopping criterion a (relatively large) fixed value for

Table 5.7: Corpus statistics for the three databases used in the experiments from the UCI and STATLOG repositories, respectively.

corpus name	MONK	DNA	LETTER
# classes	2	3	26
# features	17	180	16
# training samples	124	2 000	15 000
# test samples	432	1 186	5 000

the change in the log-posterior during the iterations of the GIS algorithm. The number of features was varied by including products of two and three pixels, respectively, of features that were increasingly farther away from each other in the image.

The jitter in the graph can be explained by the setup of the experiment, in which the GIS algorithm was terminated when the decrease of the criterion sank below a threshold that was chosen to be larger than for the other individual experiments, because of the large number of experiments and limited computation times.

In a key experiment, we used first-order and second-order features and added those third-order features for which the sum of the squared Euclidean distances between pairs of pixels was smaller than 30, which resulted in a total of 75,909 features, which implies a memory usage of 2.7GB. In this case about as many third-order features as second-order features were used. The resulting error rate was 5.6%, which is only marginally smaller than the error rate of 5.7% observed for the first- and second-order features alone.

Two interesting observations can be made about this result. The first is that the maximum entropy framework continues to show resistance to overfitting in this experiment. With more than 750,000 free parameters, the algorithm still arrives at reasonable error rates.

The second observation is related to the fact that an inclusion of linearly dependent new features does not change the maximizing distribution and therefore leaves the optimum criterion value in the maximum entropy framework unchanged (compare the discussion in the context of the maximum entropy linear discriminant analysis on page 92). This implies that there can be at most $N = 7,291$ features that can contribute to the criterion value, because you can define at most N linearly independent features on a training data set of N items. Looking at the graph in Figure 5.12, this number of 7,291 features seems reasonable, because it coincides with a leveling off of the reduction in error rate for growing number of features. (The point where the saturation of the feature space is reached does not necessarily have to be at N , because some features may already be linearly dependent such that further features can introduce new linearly independent components.) From this point of view the very small (and statistically not significant) reduction in error rate from 5.7% to 5.6% seems reasonable for the third order features and we cannot expect to reach larger reductions by including more features.

Another point of view for these connections is that if the subspace spanned by many (e.g. higher order) features is redundant in the sense of linear dependencies, you might as well eliminate these features for the maximum entropy approach without changing the results. The interesting question one should pose then becomes, which features you can expect to result in classifiers that generalize well for your data?

In experiments in which we wanted to assess the performance of the maximum entropy framework for tasks other than image recognition [Keysers & Paredes⁺ 03,

Table 5.8: Experimental results for the three UCI/STATLOG databases used with different orders of feature functions given as error rate (ER) in %. The number of parameters (#param.) refers to the total number of parameters needed to completely define the classifier. (cp. also Table 5.9)

ME-order	MONK		DNA		LETTER	
	ER[%]	#param.	ER[%]	#param.	ER[%]	#param.
first-order	28.9	36	5.6	543	22.5	442
second-order, $\text{diag}(\Sigma_k)$	28.9	70	5.6	1,083	18.6	858
second-order	0.2	308	5.1	48,873	13.5	3,562
third-order	4.4	1,362	–	–	8.1	14,586

Keyzers & Ney 04], we used three corpora from the UCI and STATLOG database, respectively [Merz & Murphy⁺ 97, Michie & Spiegelhalter⁺ 94]. The corpora were chosen to cover different properties with respect to the number of classes and features and with respect to the size. The statistics of the corpora are summarized in Table 5.7. The data sets from the UCI and STATLOG repositories are often used for comparison of general pattern recognition approaches.

MONK is an artificial decision task with categorical features also known as the monk’s problem. For the experiments, the categorical features were transformed into binary features. For the DNA task, the goal is to detect gene intron/exon and exon/intron boundaries given part of a DNA sequence. Also for this task, the categorical features were transformed into binary features. Finally, the LETTER corpus consists of printed characters that were preprocessed and a variety of different features was extracted.

Results on these data sets using higher-order features are included in Table 5.8. The row marked with ‘ $\text{diag}(\Sigma_k)$ ’ uses only those second-order features that are squares of features and thus corresponds to the use of diagonal, class-specific covariance matrices. Note that the data sets MONK and DNA only contain binary features, such that here the squares of features are identical to the features themselves and thus no change in error rate occurs. For the LETTER corpus, we can observe an improvement using the diagonal class-specific covariance matrices.

For the MONK corpus, the error rate increases for third-order features. This is obviously an effect of over-fitting to the training data. Recall that the MONK task is an artificial data set and it is likely that a few training samples exist that have a third-order correlation that does not occur in the test set. For the DNA corpus, third-order features were not tested because of the extreme requirements on main memory and run times. Interestingly, for the LETTER corpus, the performance continues to improve strongly for the use of third-order features. This behavior is consistent with the interpretation given before for the USPS corpus, because for the LETTER data we have 15,000 training samples and only 560 third-order features, such that the possible space of feature vectors that are linearly independent cannot be explored completely even using third-order features.

5.3.5 Comparison to weighted dissimilarity measures

In this section we compare the maximum entropy approach to class-dependent weighted dissimilarity measures for nearest neighbor classifiers on three databases from the UCI and STATLOG repositories [Keyzers & Paredes⁺ 03]. The two approaches are both discrimina-

tive but use different frameworks. The experiments show that the maximum entropy-based log-linear classifier performs better for the equivalent of a single prototype. On the other hand, using multiple prototypes the weighted dissimilarity measures outperforms the log-linear approach. This result suggests an extension of the log-linear method to multiple prototypes.

The use of weighted dissimilarity measures, where the weights may depend on the dimension and class and are trained according to a discriminative criterion, has shown high performance on various classification tasks [Paredes & Vidal 00]. Also for this method, a strong connection to the use of Gaussian densities can be observed if one prototype per class is used. For more than one prototype per class, the similarity leads to a mixture density approach. These connections to the Gaussian classifier are used to compare the two discriminative criteria.

Discriminative training was used in [Paredes & Vidal 00] to learn the weights of a weighted dissimilarity measure. This weighted measure was used in the nearest neighbor classification rule improving significantly the accuracy of the classifier in comparison to other distance measures, for which the parameters were not estimated using discriminative training.

Brief introduction to weighted dissimilarity measures

In [Paredes & Vidal 00], a class-dependent weighted dissimilarity measure for nearest neighbor classifiers was introduced. The squared distance is defined as

$$d^2(x, \mu) = \sum_d \left(\frac{x_d - \mu_d}{\sigma_{k_\mu d}} \right)^2, \quad \Lambda = \{\sigma_{kd}, \mu_d\},$$

where d denotes the dimension index and k_μ is the class the reference vector μ belongs to. The parameters Λ are estimated with respect to a discriminative training criterion that takes into account the out-of-class information and can be derived from the minimum classification error criterion:

$$\hat{\Lambda} = \underset{\Lambda}{\operatorname{argmin}} \sum_n \frac{\min_{\mu: k_\mu = k_n} d_\Lambda(x_n, \mu)}{\min_{\mu: k_\mu \neq k_n} d_\Lambda(x_n, \mu)} \quad (5.18)$$

In other words, the parameters are chosen to minimize the average ratio of the distance to the closest prototype of the same class with respect to the distance to the closest prototype of the competing classes.

To minimize the criterion, a gradient descent approach is used and a leaving one out estimation with the weighted measure is computed at each step of the gradient procedure. The weights selected by the algorithm are those weights with the best leaving one out estimation instead of the weights with the minimum criterion value. In the experiments, only the weights σ_{kd} were estimated according to the proposed criterion. The references μ_k were chosen as the means for the one-prototype approach. In the multiple-prototype approach the whole training set was used.

Also in this approach, we have a strong relation to Gaussian models. Consider the use of one prototype per class. The distance measure then is a class-dependent Mahalanobis distance with class-specific, diagonal covariance matrices

$$\Sigma_k = \operatorname{diag}(\sigma_{k1}^2, \dots, \sigma_{kD}^2).$$

Table 5.9: Experimental results for the three databases used with different settings of the algorithms given as error rate (ER) in %. The number of parameters (#param.) refers to the total number of parameters needed to completely define the classifier. Best other results from [Merz & Murphy⁺ 97, Michie & Spiegelhalter⁺ 94].

method	MONK		DNA		LETTER	
	ER[%]	#param.	ER[%]	#param.	ER[%]	#param.
single Gaussian	28.5	51	9.5	720	41.6	432
log-linear, first-order	28.9	36	5.6	543	22.5	442
second-order	0.2	308	5.1	48 873	13.5	3 562
weighted diss., one prot.	16.7	68	6.7	1 080	24.1	832
multiple prot.	0.0	2 142	4.7	360 540	3.3	240 416
best other	0.0	-	4.1	-	3.4	-

The decision rule is then equivalent to the use of single Gaussian models in combination with an additional factor to compensate for the missing normalization factor of the Gaussian. In the case of multiple prototypes per class, the equivalence is extensible to mixtures of Gaussian densities.

Connection between the two classifiers

As discussed in the two previous sections, the two approaches are equivalent to the use of discriminative training for single Gaussian densities with some additional restrictions. This implies that the main difference between the classifiers is the criterion that is used to choose the class boundaries:

- For Gaussian densities the criterion used is maximum likelihood (4.6). The decision boundary that results is linear for pooled covariance matrices or quadratic for class-specific covariance matrices.
- For the log-linear model the criterion used is maximum mutual information or maximum likelihood of the posterior (4.8). The decision boundary that results is linear for first-order feature functions or quadratic for second-order feature functions.
- For the weighted dissimilarity measures the criterion is a function of the intra-class distances versus inter-class distances (5.18). The decision boundary that results is quadratic for one prototype per class or piecewise quadratic for multiple prototypes per class.

Experimental results

The experiments were performed on three corpora from the UCI and STATLOG database, respectively [Merz & Murphy⁺ 97, Michie & Spiegelhalter⁺ 94], as described in Section 5.3.4. The statistics of the corpora are summarized in Table 5.7.

Table 5.9 shows a summary of the results obtained with the two methods. The figures show the following tendencies:

- Considering the four approaches that can be labeled ‘one-prototype’ (single Gaussian, both log-linear models and the one-prototype weighted dissimilarity measure), the

discriminative approaches generally perform better than the maximum likelihood-based approach (single Gaussian).

- For the two log-linear approaches, the second-order features perform better than the first-order features.
- On two of the three corpora, the log-linear classifier with first-order features performs better than the one-prototype weighted dissimilarity measure using a smaller number of parameters.
- On all of the corpora, the log-linear classifier with second-order features performs better than the one-prototype weighted dissimilarity measure, but using a larger number of parameters.
- The weighted dissimilarity measures using multiple prototypes outperforms the other regarded ('one-prototype') approaches on all tasks and is competitive with respect to the best known results on each task.

Note that second-order features perform better here although estimation of full, class-specific covariance matrices is problematic for many tasks. This indicates a high robustness of the maximum entropy log-linear approach. Note further that both the one-prototype weighted dissimilarity classifier and the log-linear model with second-order features lead to quadratic decision boundaries, but the former does not take into account bilinear terms of the features, which is the case for the second-order features.

The high error rate of the log-linear model with first-order features on the MONK corpus was analyzed in more detail. As this task only contains binary features, also the one-prototype weighted dissimilarity classifier leads to linear decision boundaries here ($x^2 = x \Leftrightarrow x \in \{0, 1\}$). Therefore it is possible to infer the parameters for the log-linear model from the training result of the weighted dissimilarity classifier. This showed that the log-likelihood of the posterior (4.8) on the training data is lower than that resulting from maximum entropy training, which is not surprising as exactly this quantity is the training criterion for the log-linear model. But interestingly the same result holds for the *test* data as well. That is, the maximum entropy training result has higher prediction accuracy on the average for the class posterior, but this does not result in better classification accuracy. This may indicate that on this corpus with very few samples the weighted dissimilarity technique is able to better adapt the decision boundary as it uses a criterion derived from the minimum classification error criterion.

A direct transfer of the maximum entropy framework to multiple prototypes is difficult, because the use of multiple prototypes leads to nonlinearities and the log-linear model cannot be directly applied any more.

The consistent improvements obtained with weighted dissimilarity measures and multiple prototypes in combination with the improvements obtained by using second-order features suggest possible improvements that could be expected from a combination of these two approaches.

5.3.6 Parameter estimation and heuristic speed-up

We briefly discuss a simple but effective method for speeding up the computations of the maximum entropy parameters using a heuristic extension of the GIS algorithm. The GIS

algorithm [Darroch & Ratchiff 72] proceeds as follows to determine the free parameters of the model. First, we choose an initial parameter set $\Lambda^{(0)} = \{\lambda_i^{(0)}\}$. Then, for each iteration $m = 1, \dots, M$ the parameters are updated according to

$$\begin{aligned}\lambda_i^{(m)} &= \lambda_i^{(m-1)} + \Delta\lambda_i^{(m)} \\ &= \lambda_i^{(m-1)} + \frac{1}{F} \log \frac{N_i}{Q_i^{(m)}} \\ Q_i^{(m)} &:= \sum_n \sum_k p_{\Lambda^{(m)}}(k|x_n) f_i(x_n, k)\end{aligned}$$

and F is a constant depending on the training data. This choice for $\Delta\lambda_i^{(m)}$ in the GIS algorithm ensures the convergence of the criterion on the training data. The computation is expensive as it requires one pass over the training data to determine the probabilities $p(k|x_n)$ for each class k , while summing values for each feature f_i . Furthermore, the convergence of the algorithm may take many iterations for complex distributions, resulting in a high computational cost.

Interestingly, it can be observed for different tasks that consecutive update vectors $\Delta\lambda^{(m)}$ and $\Delta\lambda^{(m+1)}$ tend to be similar to each other especially for increasing numbers of iterations [Keyzers & Och⁺ 02b]. This similarity can be measured by the cosine of the angle between two consecutive update vectors. Now, we can assume that in regions where the cosine is close to one (i.e. the vectors point into very similar directions in the vector space of possible parameter sets), the update vector $\Delta\lambda^{(m)}$ can be multiplied by a factor greater than one (where the factor depends on the similarity to the previous update). This yields a faster convergence of the algorithm (i.e. convergence within a smaller number of iterations). This procedure (‘efficient GIS’) implies that we cannot theoretically guarantee convergence of the algorithm any more, but experiments show possible speed-ups of 2 to 100 times faster convergence. Furthermore, we can ensure convergence of the algorithm by observing the log-probability on the training data in each iteration and falling back to the conventional update strategy if it decreases.

Note that there exists an enhanced version of the GIS algorithm known as improved iterative scaling [Della Pietra & Della Pietra⁺ 97] which in most cases converges faster. The speed-up method presented here may also be applied effectively to this improved version. This is especially true in the case where feature normalization (see below) is applied, as in that case GIS and improved iterative scaling are identical.

Figure 5.13 shows the log-probability on the training data for first-order features as a function of the number of iterations for standard GIS and heuristic speed-up GIS. It can be observed that on this data the proposed speed-up can save around 90% of training time.

5.4 Maximum entropy linear discriminant analysis

We discussed the relationship between the discriminative training of Gaussian models and the maximum entropy framework for log-linear models in the previous section. Observing that linear transforms leave the distributions resulting from the log-linear model unchanged, we now derive a discriminative linear feature reduction technique from the maximum entropy approach and compare it to the well-known linear discriminant analysis. From experiments

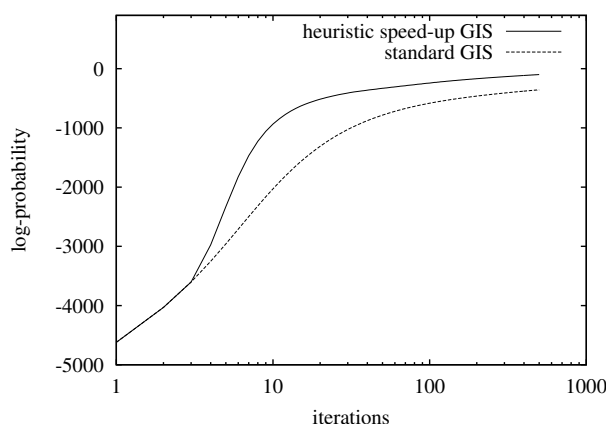


Figure 5.13: Log-probability on the training data (USPS, first-order features) as a function of the number of iterations for standard GIS and heuristic speed-up GIS.

on different corpora we observe that the new technique performs better than linear discriminant analysis if the dimensionality of the feature space is large with respect to the number of classes. The results presented here have been previously described in [Keysers & Ney 04].

5.4.1 Derivation of MELDA

Linear discriminant analysis (LDA, [Duda & Hart⁺ 01a, pp.117ff.]) is a widely used tool in pattern recognition. It is introduced as a linear feature reduction technique in Section 4.5.2. The derivation of LDA can be based on the assumption that the class conditional distributions are Gaussians. We have shown that there exists a strong relation between discriminative log-linear models with the appropriate choice of features and discriminative training of Gaussian models in the previous section. This connection leads to a counterpart of LDA in the context of log-linear models.

When using log-linear models for the class posterior in classification, it can be observed that the model distributions are not changed by any non-singular linear transformation of the feature space. Furthermore, no linear feature reduction can improve the log-likelihood of the posterior on the training data.

As these models (also called maximum entropy models) are successfully used in a wide range of pattern recognition applications, the question is raised how this framework can be used to estimate a discriminative linear feature reduction transformation. This section provides an answer to this question resulting in a maximum entropy linear discriminant analysis (MELDA). We show that the solution has two properties:

- The transformation preserves exactly that linear subspace of the original feature space that is orthogonal to the class boundaries chosen by the maximum entropy training.
- The solution follows from an appropriate choice for the degrees of freedom in the maximum entropy solution for the transformation matrix when considering the connection between maximum entropy training and Gaussian models.

We discuss experiments comparing LDA and MELDA that suggest that LDA leads to better results for tasks with lower dimensionality of the feature space, whereas MELDA performs better on tasks with high dimensionality. This result may be helpful when a feature reduction technique for tasks with comparatively few classes with respect to the number of features is needed. Moreover, classification results can be improved by first introducing artificial new features and then applying MELDA to the larger feature space.

We summarize one of the possible formulations of LDA here (again) to compare it to the MELDA approach we derive. LDA aims at minimizing intra-class variance with respect to inter-class variance using a linear transformation $\tilde{x} = Ax$, i.e. minimize

$$\min_A \sum_n \|A(x_n - \mu_{k_n})\|^2, \quad \text{under condition} \quad \sum_{n,k} \|A(x_n - \mu_k)\|^2 = c \quad (5.19)$$

One computational method to determine the LDA matrix A is to compute the within class scatter matrix W and the between class scatter matrix B and then solve the generalized eigenvalue problem $BA^T = \lambda WA^T$. This requires the computation of the $D(D+1)/2$ independent entries of the scatter matrices, which can be computationally inefficient in the case $D > N$, i.e. if there are more features than training samples. In this case, we can reduce the size of the scatter matrices by applying a singular value decomposition to the data, computing a projected representation of the data in $\text{span}(\{x_n\})$, computing the LDA in this vector space with lower dimensionality and then reversing the projection.

To derive the maximum entropy LDA, first note that in the log-linear model any non-singular linear transformation of the feature space leaves the maximum class posterior distribution unchanged. Consider a transformation $\tilde{x} = Ax$:

$$\begin{aligned} p_{\tilde{\Lambda}}(k|\tilde{x}) &= \frac{\exp[\tilde{\alpha}_k + \tilde{\lambda}_k^T A x]}{\sum_{k'} \exp[\tilde{\alpha}_{k'} + \tilde{\lambda}_{k'}^T A x]} \\ &= \frac{\exp[\alpha_k + \lambda_k^T x]}{\sum_{k'} \exp[\alpha_{k'} + \lambda_{k'}^T x]} = p_{\Lambda}(k|x) \end{aligned} \quad (5.20)$$

with $\lambda_k^T = \tilde{\lambda}_k^T A$ and $\alpha_k = \tilde{\alpha}_k$. From the uniqueness of the maximum entropy distribution it follows immediately that the distribution is not changed. (Note that the uniqueness applies to the distribution, not to the parameter values.)

Now, even if A does not have full rank, all solutions $\tilde{\Lambda}$ have at least one corresponding solution Λ , i.e. the criterion (4.8) can never be improved by applying a linear transformation to the feature space. This observation motivates the following approach: First compute the parameters in the original feature space, then choose the linear transformation accordingly.

Assume $D \geq K$. We want to estimate a transformation matrix $\tilde{x} = Ax$ with $A \in \mathbb{R}^{(K-1) \times D}$ maximizing the log-likelihood of the posterior of the log-linear model $p_{\tilde{\Lambda}}(k|\tilde{x})$.

The geometric interpretation of this result is the following. The functional form (5.16) of the log-linear model implies that in the computation of $p_{\Lambda}(k|x)$ those components of x that are orthogonal to all of the difference vectors $\{\lambda_k - \lambda_{k'}\}$ do not change the result. This implies that only the projections of x onto the subspace

$$\text{span}(\{\lambda_k - \lambda_{k'}\}) = \text{span}(\{\lambda_2 - \lambda_1, \lambda_3 - \lambda_1, \dots, \lambda_K - \lambda_1\}) = \text{span}(A)$$

influence the posterior $p_{\Lambda}(k|x)$. Therefore, defining the transformation matrix as

$$A = ((\lambda_2 - \lambda_1), (\lambda_3 - \lambda_1), \dots, (\lambda_K - \lambda_1))^T \quad (5.21)$$

achieves the required result. The transformation retains those parts of the feature space orthogonal to the linear class boundaries chosen by maximum entropy training.

To observe the second property of the chosen transformation, we consider the relationship between the Gaussian model and the log-linear model again. In the Gaussian model we know that if $\tilde{x} = Ax$ then $\tilde{\mu} = A\mu$ and $\tilde{\Sigma} = A\Sigma A^T$. From the equivalence with the Gaussian posterior in (5.17) we see that for the estimated parameters we have:

$$\lambda_k^T = \tilde{\lambda}_k^T A = \tilde{\mu}_k^T \tilde{\Sigma}^{-1} A \quad (5.22)$$

This equation is under-determined for A if only the parameter vectors $\{\lambda_k\}$ are known. After the transformation, we choose the mean vectors to be the null vector and the $K - 1$ unit vectors, respectively, and the covariance matrix to be the identity matrix:

$$\begin{aligned} \tilde{\mu}_1 &= (0, 0, \dots, 0)^T \\ \tilde{\mu}_2 &= (1, 0, \dots, 0)^T \\ \tilde{\mu}_3 &= (0, 1, \dots, 0)^T \\ &\vdots \\ \tilde{\mu}_K &= (0, 0, \dots, 1)^T \end{aligned} \quad \tilde{\Sigma} = I_{K-1}$$

This restriction leaves (5.22) satisfiable, as we have additional degrees of freedom in the log-linear model and we can always transform the maximum entropy solution by setting $\lambda_k \leftarrow \lambda_k - \lambda_1$, forcing $\lambda_1 = 0$. Thus, we obtain the same solution for the transformation matrix A as given in (5.21).

A further connection between LDA and MELDA can be observed if we derive an expression similar to (5.19) for MELDA. Consider again a Gaussian model as in (5.17) and (5.20) where we now neglect the class priors $p(k)$ and assume $\Sigma = I$. We obtain the expression:

$$\begin{aligned} \operatorname{argmax}_A \sum_n \log p_{\tilde{\Lambda}}(k_n | \tilde{x}_n) = \\ \operatorname{argmin}_A \sum_n \frac{1}{2} \|A(x_n - \mu_{k_n})\|^2 + \log \sum_k \exp \left[-\frac{1}{2} \|A(x_n - \mu_k)\|^2 \right] \end{aligned}$$

In this formulation we do not need an additional constraint and we observe that in the sum over all classes the distances to the closer competing class are (exponentially) more ‘important’ than classes with means far away. This is not the case for the conventional LDA and emphasizes the discrimination between directly competing classes.

5.4.2 Experimental results

In a first experiment, we compared LDA to the maximum entropy approach directly. We observed the error rates on the USPS task using single Gaussians with pooled diagonal covariance matrix, 9-fold virtual training data and an LDA estimated on 40 clusters or pseudo-classes yielding a 39-dimensional feature space. The resulting error rates are shown in Table 5.10. We can observe that the LDA improves the result for the maximum likelihood estimation, but not for the discriminative MMI estimation using the maximum entropy framework. This effect shows that LDA is implicit in the maximum entropy model because of the invariance with respect to affine transformations of the features as discussed above. This implies as stated above that the application of LDA can never improve the maximum

Table 5.10: Results of first comparison between LDA and maximum entropy (error rates on USPS [%]).

criterion	LDA	
	yes	no
ML	13.1	22.3
MMI (MaxEnt)	9.9	8.7

Table 5.11: Summary of the data statistics for the data sets used in the experiments with the MELDA.

name	K	D	$D/(K-1)$	$N(\text{train})$	$N(\text{test})$
MONK	2	17	17	124	432
MONK ²	2	153	153	124	432
DNA	3	180	90	2 000	1 186
DNA ²	3	16 290	8 145	2 000	1 186
LETTER ²	26	136	5	15 000	5 000
USPS	10	256	28	7 291	2 007
USPS ²	10	4 930	548	7 291	2 007

entropy estimation as measured by the criterion used, which is reflected in the error rates here.

The further experiments use the data sets MONK, DNA, and LETTER, as described in Section 5.3.4 and the USPS corpus as described in Section 3.1.1. The statistics of the corpora are summarized in Table 5.11, where K is the number of classes, D is the number of features, $D/(K-1)$ is the factor of feature reduction for LDA and MELDA, and N is the number of samples. From each of the corpora we created a “squared” version (indicated by a superscript 2) by using all feature products $x_i x_j, i \geq j$ as additional features. This procedure was based on the finding that the performance of the log-linear classifier generally improves with larger number of features [Keysers & Paredes⁺ 03]. The squared corpora have $D(D+1)/2$ features with the exception of the USPS corpus. Here, a subset of the product features was chosen based on pixel neighborhoods because of memory limitations for the LDA computation. For the LETTER corpus, we only consider its squared version LETTER², since the original corpus has $D = 16 < 26 = K$. and therefore a dimension reduction to a value different from $K-1$ would have had to be performed. We did not consider this experiment, as we used a consistent reduction to $K-1$ features for all transformations.

The results of our experiments are summarized in Table 5.12. On each of the corpora (names abbreviated by first letter), we compare the error rates of three different classifiers for each of the feature reduction (FR) methods (including no feature reduction). The three classifiers are the single Gaussian (SG, using pooled diagonal covariance matrices), the nearest neighbor (NN, using Euclidean distance), and the maximum entropy (ME) classifier. Note that in the lines with no feature reduction, the number of features used is larger by a factor of up to 8,145.

Table 5.12: Experimental results: error rates [%]. FR: Feature reduction method. Cl: Classifier (SG: single Gaussian with pooled diagonal covariance matrix, NN: nearest Neighbor with Euclidean distance, ME: maximum entropy). Corpus names abbreviated by first letter.

FR	Cl.	M	M ²	D	D ²	L ²	U	U ²
NONE	SG	28.5	22.7	9.9	11.2	44.2	19.4	25.0
	NN	21.3	21.3	23.4	33.1	4.7	5.6	7.2
	ME	28.7	0.9	6.2	5.1	9.5	8.8	7.2
LDA	SG	26.6	30.8	6.7	52.4	18.4	11.5	22.2
	NN	27.5	28.5	6.2	52.2	4.5	10.9	22.9
	ME	26.6	30.8	4.4	53.3	13.9	11.0	22.9
MELDA	SG	25.0	0.9	9.1	7.5	42.0	32.4	27.4
	NN	26.2	0.9	6.0	5.3	5.9	14.7	12.0
	ME	28.7	0.9	6.2	5.1	9.5	8.8	7.2

As expected, the LDA performs better for the Gaussian classifier and MELDA performs better for the maximum entropy classifier as both methods are especially suited for these cases. Nevertheless in both these cases the general tendency is preserved: the relative performance of MELDA with respect to LDA increases with the feature reduction factor $D/(K-1)$. To illustrate this effect, Figure 5.14 shows the relative improvement of MELDA over LDA for the nearest neighbor classifier $1 - \text{err}_{\text{MELDA}} / \text{err}_{\text{LDA}}$ (values > 0 indicate that MELDA performs better than LDA) for LETTER², MONK, USPS, DNA, MONK², USPS², DNA² in the order of feature reduction factor $D/(K-1)$. One further result is that in all cases where the squared and the original corpus were used, MELDA performs better in the artificially enlarged features space, while LDA performs worse.

Note that most of the presented results are not competitive with the best error rates obtained on the used corpora. This is due to the fact that none of the classifiers or feature reduction methods were tuned to the specific tasks in these experiments (e.g. as seen above the performance of LDA on the USPS corpus can be considerably improved by first clustering the corpus into 40 clusters and then computing an LDA matrix resulting in a 39-dimensional feature space). However, this does not weaken the obtained results as we are interested in the relative performance of MELDA and LDA, knowing that LDA is a widely used technique.

The main result is that MELDA performs better than LDA when the feature reduction factor $D/(K-1)$ is large. This result is consistent with experience that the maximum entropy framework performs often well for data with a large dimensionality. On some corpora using a very small number of transformed features (even one or two) already produces very good results using MELDA (e.g. MONK², DNA²).

Regarding LDA and log-linear models [Hastie & Tibshirani⁺ 01, p.105] write: “It is generally felt that logistic regression is a safer, more robust bet than the LDA model, relying on fewer assumptions. It is our experience that the models give very similar results, even when LDA is used inappropriately, such as with qualitative predictors.” The experiments reported in this paper suggest that the log-linear model gains in robustness and produces better results when the number of features with respect to the number of classes is large.

This result may be helpful when a feature reduction technique for tasks with comparatively few classes with respect to the number of features is needed. Moreover, classification

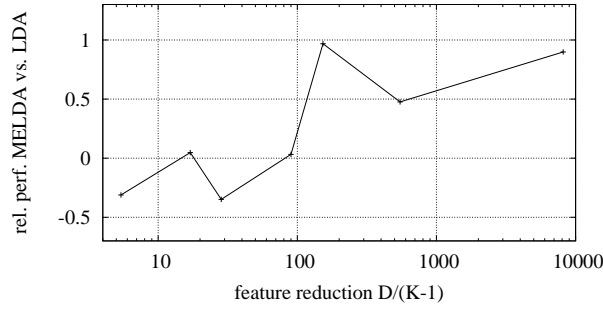


Figure 5.14: Relative improvement of the error rate of MELDA over LDA for the nearest neighbor classifier ($1 - \text{err}_{\text{MELDA}} / \text{err}_{\text{LDA}}$), values > 0 indicate that MELDA performs better than LDA, for LETTER², MONK, USPS, DNA, MONK², USPS², DNA² (in order of feature reduction $D/(K-1)$).

results can be improved by first introducing artificial new features and then applying MELDA to the larger feature space.

5.5 Conclusion

The investigation of relationships to the Gaussian case allowed us to derive various interesting results in this chapter, including

- a probabilistic interpretation of tangent distance, which enabled us to estimate linear representations of the variability of given data;
- the possibility of the use of the GIS algorithm and maximum entropy models for the discriminative estimation of the equivalent of full covariance matrices;
- the derivation of the new maximum entropy linear discriminant analysis.

We could observe reductions in error rate on various data sets, e.g. by the use of the tangent distance in the Gaussian framework or for single prototype classifiers that build on the maximum entropy framework and are computationally very efficient. The interpretation within a Gaussian framework often opens new possibilities for the use of an approach, as for example the tangent distance could be integrated into a hidden Markov model based classifier for continuous handwriting or for sign language recognition as discussed briefly in Section 6.8.

6 Nonlinear deformation models

Writing is nature's way of showing us how sloppy our thinking is.
– Guindon

Math is nature's way of showing us how sloppy our writing is.
– L. Lamport

In this chapter, we discuss different two-dimensional non-linear deformation models for image recognition. The tangent distance as discussed in the previous chapter effectively models global variability of the image, but it is sensitive to local non-linear image transformations, which can be modeled using the approaches presented here. Starting from a true two-dimensional model, we proceed to pseudo-two-dimensional and zero-order deformation models. Experiments show that it is most important to include suitable representations of the local image context of each pixel to increase the recognition performance. With these methods, we achieve very competitive results across five different handwritten character recognition tasks, in particular an error rate of 0.5% on the MNIST task. We also show that the same methods can be used for the categorization of medical images. Using the methods presented here, the previous best error rate of 8.0% on the IRMA-1617 task could be considerably reduced by about one third to 5.3%. In a recent international competition of medical image categorization held within the ImageCLEF 2005 evaluation, the best result among those handed in by twelve groups was achieved using one of the models presented here. Furthermore, improvements in the domain of image sequence recognition (sign language word recognition, gesture recognition) can be achieved by modeling the image variability with these deformation models.

We describe deformation models of different order: Two-dimensional (2D) hidden Markov models (HMMs) take into account the connections between the displacements of pixels in both dimensions of the image plane. They have been introduced in several publications, e.g. [Uchida & Sakoe 98].

Note that the term HMM is not used in its canonical meaning here, as the focus is not on the probability distributions for the emission and transition densities. The term ‘alignment model’ would be more precise, but in the literature it is more common to call such models HMMs, therefore we will also adopt this terminology here. The connection to the conventional HMMs that define a probability distribution on sequences as used for example in speech recognition, are closely related to the distance-based approach that is used here. By using a nearest neighbor approach instead of training, using the maximum approximation or Viterbi alignment, and by taking the negative logarithm of the emission densities as score, we can arrive at the distance-based model used here starting from a conventional HMM.

Pseudo-two-dimensional (P2D) HMMs relax the constraints in one of the dimensions, thus reducing the computational effort considerably. These models are described for example in [Kuo & Agazzi 94]. The P2DHMM aligns image columns (or rows) onto image columns of

the reference image. This restriction can be quite strong for some applications. Therefore we introduce an extension to allow additional distortions perpendicular to the image columns. The resulting model is denoted as pseudo-two-dimensional hidden Markov distortion model (P2DHMDM) with a somewhat lengthy abbreviation.

If we further relax the constraints on the deformation grid such that the pixel displacements are independent of each other, we arrive at a zero-order model that we call image distortion model (IDM). This simple model has been introduced in the literature several times with different names, see for example [Keysers & Dahmen⁺ 00b].

The experiments show that the important aspect for these models is the use of local context information at the pixel level. Using this context information in the form of the image gradient and local image parts, the performance can be improved significantly, leading to state-of-the-art results.

In the experiments, some general results could be observed. For all of the models, the performance increased significantly with the use of local image context. This increase was greater for the simpler models, leading to the conclusion that the context information can compensate for the neglected restrictions. The fact that the more complex models did not outperform the simpler models can be explained as follow: due to the computational complexity of the minimization, approximation methods had to be applied, and for the more complex models usually a smaller number of images was preselected in the hierarchical distance framework using the Euclidean distance to reduce the computational effort.

The software used in the experiments is available for download¹.

6.1 Related work on image matching

There is a large amount of literature dealing with the application of graph matching to computer vision and pattern recognition tasks. For example, graph matching procedures can be used for labeling of segmented scenes. Note that often matching is done after segmentation or contour extraction [Veltkamp & Hagedoorn 01]. In that case the matching is inherently a two-stage procedure while we prefer to use an appearance-based approach in which no intermediate segmentation errors can occur. In the following, we give an overview of related work on matching.

In [Belongie & Malik⁺ 02] a method for image matching is described that is based on representations of the local image context called ‘shape contexts’ which are only extracted at edge points. An assignment between these points is determined using the Hungarian algorithm and the image is matched using thin-plate splines, which is iterated until convergence.

[Wiskott & Fellous⁺ 97] present a higher level approach for face recognition using a single prototype. A graph of facial point positions, as eyes, tip of nose, corners of mouth, etc. is trained manually on a small number of faces. The local surrounding at the facial positions are described by so called jets. Those are a number of values extracted at the wanted facial positions of wavelet transforms of the image with different kernels differing in orientation and frequency. After the points have been found, the graphs are compared using the distances between the node representations.

In [Lowe 99] local features that are invariant to translation, rotation, and scaling and partially invariant to illumination changes and projective transformations are extracted at stable points in scale space. For testing, the features are matched using nearest neighbor.

¹ <http://www-i6.informatik.rwth-aachen.de/~gollan/w2d.html>

The verification of the matches is done by solving a low residual least square problem for the global transformation parameters. According to the author, this enables the system to find partially occluded objects in cluttered environments.

In [Brown & Lowe 02] invariant local features are used to localize matching positions in different images. This has been used to estimate transformations or to create 360° views by joining overlapping photographs. To do so features that are similar to those presented in [Lowe 99] are extracted at interest points in scale space. Then a nearest neighbor matching is performed between the two images. A transformation invariant outlier detection is used to identify false matches. Then a transformation is computed such that the interest points of the first image match those of the second.

[Levin & Pieraccini 92] extend the one-dimensional dynamic time warping to two dimensions and note that the algorithm has exponential complexity. To arrive at a feasible algorithm they propose to restrict the problem by assuming independence between vertical and horizontal displacement and thus arrive at a model that is essentially a pseudo-two dimensional model. The authors also give a statistical interpretation they call planar HMM and present the application of the recognition of handwritten digits, learning a model for each class using a modified Viterbi algorithm.

[Agazzi & Kuo 93, Kuo & Agazzi 94] also describe the use of pseudo two-dimensional hidden Markov models in the domain of document processing. They use such a model to detect the occurrences of keyword in images of documents that are subject to noise.

[Uchida & Sakoe 98] present a dynamic programming algorithm for the true two-dimensional image deformation model which takes into account continuity and monotonicity constraints. The exponential computational is reduced by finding (possibly suboptimal) solutions using beam search.

Some of the literature relevant for this chapter is published in the context of Markov random fields [Abend & Harley⁺ 65], although here usually the task is not recognition, and if recognition plays a role, the random field is not used for matching to given samples or prototypes. We discuss the relationship between pixel labeling and matching in a later paragraph.

[Moore 79] already presented an algorithm to compare two-dimensional patterns allowing for spatial matching between two images. The algorithm is based on an extension of the Levenshtein distance to 2D. This algorithm was recently re-discovered in a slightly modified way and applied to face images, although results were not convincing [Lei & Govindaraju 04]. Experiments with Moore's algorithm for handwritten digit recognition are also presented in [Keysers 00] and a brief summary is given in Section 6.5.5.

[Li & Najmi⁺ 00] use a two-dimensional HMM for image segmentation. The HMM is used to incorporate context of decisions for neighboring blocks in images and is applied to the segmentation of aerial images and document images. The authors note that appropriate restrictions of the 2D model are necessary due to the computational complexity. The method is not used for image matching but to assign labels to parts of images. The authors refer to [Devijver 86] who in their words first discussed 2D models for segmentation and noted that the complexity of the algorithms was exponential in the size of the image.

[Devijver 86] discusses a second order Markov mesh, in which the dependencies between labels assigned to each pixel of an image is limited to immediate neighbors in the two dimensions of the image. Devijver notes that when examining the joint likelihood "the intrinsic complexity of processing an $M \times N$ image is exponential in $M \times N$." (This is not strictly true, because using dynamic programming this complexity can be limited to

being exponential in the minimum of image width and height [Uchida & Sakoe 98], but no polynomial algorithms can be found unless $P = NP$ [Keysers & Unger 03]). The author references other works, in which different simplifying assumptions have been proposed, as e.g. limiting the dependencies to small regions. Then he proceeds to make a special independence assumption that allows him to formulate a non-linear recurrence that can be evaluated in linear time with respect to the image size (and a second term cubic in the number of possible labels). The algorithm is applied to the task of image smoothing and image segmentation.

In [Uchida & Sakoe 03a] the authors investigate the use of class-specific deformations for handwritten character recognition using three different elastic matching techniques. These so-called ‘eigen-deformations’ improve the recognition results considerably, but no pixel context is taken into account.

[DeMenthon & Doermann⁺ 00] present an iterative approximation to 2-D image models that is based on a tiling of the image. No quantitative results are presented but distances between two images and a mixture of these images as well as between an image and a scrambled version of that image are discussed.

The use of iterative matching using coupled one-dimensional HMMs is proposed for two-dimensional matching in [Perronnin & Dugelay⁺ 03], but no pixel context is considered. The approach is applied to artificially distorted face images from the ORL database.

A large variety of methods for classification of medical images is discussed in the literature. A number of these have also been evaluated on the data used in this work, the RWTH Aachen University IRMA (image retrieval in medical applications) database. Best results on these data were achieved using a statistical model incorporating various techniques that cope with the inherent variability of the data [Keysers & Dahmen⁺ 03]. Other techniques like the use of cooccurrence matrices or the Euclidean nearest neighbor yielded higher error rates when applied to this task. The statistical approach with a model of variability (distorted tangent distance) obtained an error rate of 8.0% on the used database of radiographs.

[Würtz 97] describes a recognition system for faces in which local descriptors (using Gabor filters) are matched to a reference in a coarse to fine manner, starting with a global displacement and then aligning local descriptors over two more scales. The coarse to fine matching procedure cannot guarantee optimality but produces adequate mappings for face images. It can be compared to our simulated annealing solution to the 2D-warping problem, although the coupling between neighboring displacements is less strict. The final decision is taken by using the average similarity of locally corresponding feature vectors after subtracting a measure of overall deviation from a rigid matching as a penalty for strong distortions.

[Fergus & Perona⁺ 03] have used a framework that combines information about shape, appearance, relative scale and occlusion using probabilities. Local features of various sizes are extracted at image positions with a high local entropy, scaled to 11×11 sized windows and matched with reference features giving the probability for the appearance. Occlusion is modeled with a background model. From this matching the probability for the shape and the scale are estimated. The training is done using an EM-Algorithm. This method has been tested on different databases for the two-class problem object absence versus object presence.

In the past, matching for handwritten digit recognition has also been approached as a problem of matching a one-dimensional structure in two-dimensional images (the production of handwritten symbols by drawing with a pen implies that there is an underlying one-dimensional structure), e.g. [Burr 81, Steinbiss & Ullrich⁺ 88, Hinton & Williams⁺ 92]. However, in the current literature it seems that for off-line recognition two-dimensional mod-

els perform considerably better, although outlines are still used directly [Cai & Liu 99] and indirectly [Belongie & Malik⁺ 02].

We now take a brief look at the relationship between algorithms that assign labels to each pixel and pixel-to-pixel matching algorithms as they are used in this work. We can reduce the matching problem to one of pixel label assignments if we let the set of labels be the set of pixels of the reference image. At the same time, we let the dependencies of the labels on the pixel values depend on the difference between the pixel values of the assigned reference pixel. In that case, and using an appropriate set of dependencies between the assignments, we obtain a model for image matching from one of pixel labeling. This implies that in principle all labeling algorithms can also be used for matching. Thus, for example the non-contextual maximum likelihood labeling algorithm that assigns the label for each pixel based on the value of that pixel alone corresponds to the IDM matching algorithm, which assigns a matching pixel based on the value of the pixel alone. More experiments to explore the usefulness of labeling algorithms for image matching would therefore be of interest, but are not addressed in this work.

Another type of publications that should be discussed in the context of image matching is literature that deals with image registration. Registration is a concept that is most often applied in medical applications and is connected to the methods discussed here by the inherent optimization or matching process. It is used e.g. to align two X-ray images of the same patient before and after treatment or to align cross-sections of histological images to construct a 3-D representation e.g. in studies that regard rat brains. Examples of registration algorithms are [Viola & Wells 97] and [Fischer & Modersitzki 01], but the literature with respect to this subject is vast. In their review of image warping methods [Glasbey & Mardia 98] give a summary of various methods for image registration or warping and state that the fundamental tradeoff is between the smoothness of the image transform and the quality of the achieved match. The major difference between image registration and matching for classification is that in registration it is known that the two images should match well and the tradeoff is between a good match and a small distortion of the image. In matching for image classification we do not know in advance if the two images should result in a good match and the tradeoff is between a good match for the same class and a generally good match also for different classes. In other words, in image registration, the goal is not to discriminate between different classes of images but to avoid a distortion of the image that is too large. This does not mean that registration algorithms could not be applied to matching for classification, in fact it might well be worthwhile to investigate the performance of various algorithms for registration for the task of classification, but this large field is not addressed in this work.

6.2 Framework for recognition using nonlinear matching

For the discussion of the nonlinear deformation models and the experiments performed with these we use the following framework. We denote the test image (or observation) by $A = \{a_{ij}\}$ where the pixel positions are indexed by $(i, j), i = 1, \dots, I, j = 1, \dots, J$. We choose to deviate from the notation used in other parts of this work in order to be able to use the symbols (x, y) to denote the pixel positions within the reference image (or model) $B = \{b_{xy}\}, x = 1, \dots, X, y = 1, \dots, Y$. At each image position we have a vector of values $a_{ij}, b_{xy} \in \mathbb{R}^U$ that can represent gray values ($U = 1$), color values ($U = 3$), the vertical and horizontal

Table 6.1: Overview of the constraints imposed on the image deformation mappings for the different deformation models.

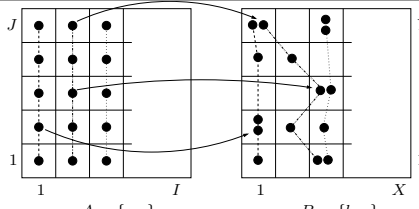
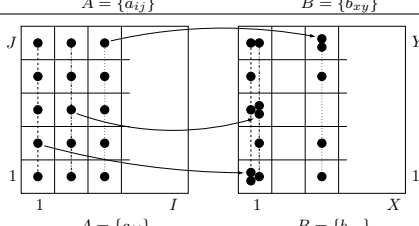
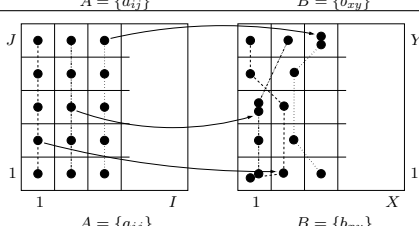
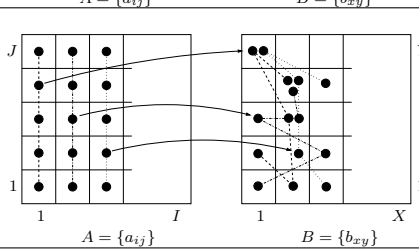
2DW	<p>2-Dimensional Warping (second-order), complete 2D constraints, minimization NP-complete</p> $x_{1j} = 1, x_{Ij} = X, y_{i1} = 1, y_{iJ} = Y,$ $x_{i+1,j} - x_{ij} \in \{0, 1, 2\}, x_{i,j+1} - x_{ij} \in \{-1, 0, 1\},$ $y_{i,j+1} - y_{ij} \in \{0, 1, 2\}, y_{i+1,j} - y_{ij} \in \{-1, 0, 1\}$	
P2DHMM	<p>Pseudo 2-Dimensional Hidden Markov Model (first-order), match columns on columns, columns are independent</p> $x_{1j} = 1, x_{Ij} = X, y_{i1} = 1, y_{iJ} = Y,$ $\exists \{\hat{x}_1, \dots, \hat{x}_I\} : \hat{x}_{i+1} - \hat{x}_i \in \{0, 1, 2\},$ $x_{ij} - \hat{x}_i = 0, y_{i,j+1} - y_{ij} \in \{0, 1, 2\}$	
P2DHMDM	<p>Pseudo 2-Dimensional Hidden Markov Distortion Model (first-order), allow horizontal displacements in P2DHMM</p> $x_{1j} = 1, x_{Ij} = X, y_{i1} = 1, y_{iJ} = Y,$ $\exists \{\hat{x}_1, \dots, \hat{x}_I\} : \hat{x}_{i+1} - \hat{x}_i \in \{0, 1, 2\},$ $x_{ij} - \hat{x}_i \in \{-1, 0, 1\}, y_{i,j+1} - y_{ij} \in \{0, 1, 2\}$	
IDM	<p>Image Distortion Model (zero-order), disregard relative displacements of neighboring pixels, restrict absolute displacement</p> $x_{ij} \in \{1, \dots, X\} \cap \{i' - w, \dots, i' + w\}, i' = \lceil i \frac{X}{I} \rceil,$ $y_{ij} \in \{1, \dots, Y\} \cap \{j' - w, \dots, j' + w\}, j' = \lceil j \frac{Y}{J} \rceil,$ <p>with warp range w, e.g. $w = 3$</p>	

image gradient ($U = 2$), or a larger pixel context (e.g. $U = 18$ for 3×3 pixel contexts of the image gradients).

We now consider image deformation mappings $(x_{11}^{IJ}, y_{11}^{IJ})$ between the reference and the test image that must fulfill certain constraints, denoted by membership in the set \mathcal{M} of all mappings allowed in the considered model, $(x_{11}^{IJ}, y_{11}^{IJ}) \in \mathcal{M}$. Each mapping is a function

$$(x_{11}^{IJ}, y_{11}^{IJ}) : (i, j) \mapsto (x_{ij}, y_{ij}). \quad (6.1)$$

The set \mathcal{M} of possible image deformation mappings depends on and characterizes the model used. Each model will be discussed in more detail in the following sections. A summary of the models is given along with their formal definitions and example mappings in Table 6.1.

The distortion models with their allowed deformation mappings differ in the way they treat the interdependence of local pixel displacements. When a pixel (i, j) is mapped onto a target pixel (x_{ij}, y_{ij}) , we can observe the difference of this mapping to the mapping of its neighbors $(i - 1, j)$ and $(i, j - 1)$ in the original image (cp. Figure 6.1). For example, to ensure a continuous and monotonous mapping, we may want to disallow that the difference in displacement of neighboring pixels is negative (no crossings) and that the difference is larger than, say, two pixels (no holes). Also, we want to restrict the maximum horizontal displacement difference for vertically neighboring pixels to at most, say, one pixel and vice

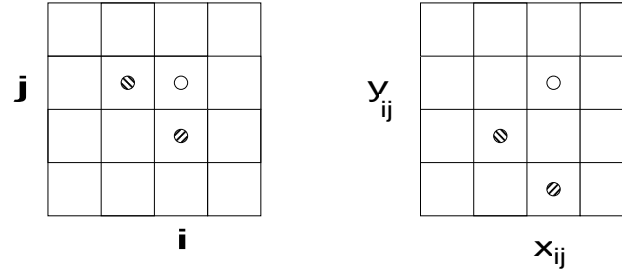


Figure 6.1: 2-dimensional interdependence of local displacements.

versa. Following this approach leads to the true 2-dimensional warping. The other models follow if some of the restrictions are relaxed.

From the constraints presented in Table 6.1 we can observe that there are relative and absolute constraints: relative constraints speak about the relation between the mappings of neighboring pixels, like e.g. $x_{i+1,j} - x_{ij}$, while absolute constraints only look at the original position of the pixel in the image, e.g. possible values for x_{ij} only depend on i and j . The IDM only includes absolute constraints, which allows efficient minimization. Often, an absolute constraint like the one for the IDM is additionally imposed for the other models, e.g. the warp range is limited for the P2DHMM by adding to the constraints of the P2DHMM the constraints of the IDM.

The constraints presented here are all hard constraints, i.e. it is either possible or not to have a certain mapping under a specific model. It is also possible to use cost functions instead of these hard constraints, such that certain mappings are allowed, but only at a higher cost, where this cost is added to the distance value. (The hard constraints are a special case of the cost functions, where the cost takes only the values 0 and ∞ .) Cost functions also allow to additionally penalize some allowed mappings, e.g. $x_{i,j+1} - x_{ij} = 0$ could be assigned cost zero and $x_{i,j+1} - x_{ij} > 0$ could involve higher costs, thus penalizing mappings that deviate much from the identity mapping. Some of such cost functions are easily integrated into the algorithms and do not change the running time significantly, while others may require algorithmic changes. Cost functions of the first kind have been evaluated in detail in [Gollan 03] but in comparison to the additional parameters that need to be tuned no significant improvements could be obtained such that we do not discuss cost functions that can be added to the hard constraints here.

Figure 6.2 shows example images that result from a matching using the different distortion models discussed in this chapter. Each group of four images consists of (A) the test image, (B) the transformed reference image, (C) the displacement grid, and (D) the reference image. The left group shows images of ‘different classes’, while the right group shows images of the ‘same class’. The same class here consists of images of the twin brothers Klaus and Wolfgang Macherey, members of the research staff at the Lehrstuhl für Informatik VI. The ‘different class’ then uses an image of the author. The examples were created using the image gray values alone, i.e. $U = 1$. We can observe that the models with more restrictions do not allow as good matches for the same class as the models with less constraints in the matching. When using the models, we are interested in discriminating between different classes, therefore our goal is to have good matchings (small distances between the test image and the transformed reference image) for images of the same class, but matchings with large distances for images of different classes. When using only the gray values, this is obviously not achieved by the models with less restrictions, e.g. the IDM. This is also reflected in the

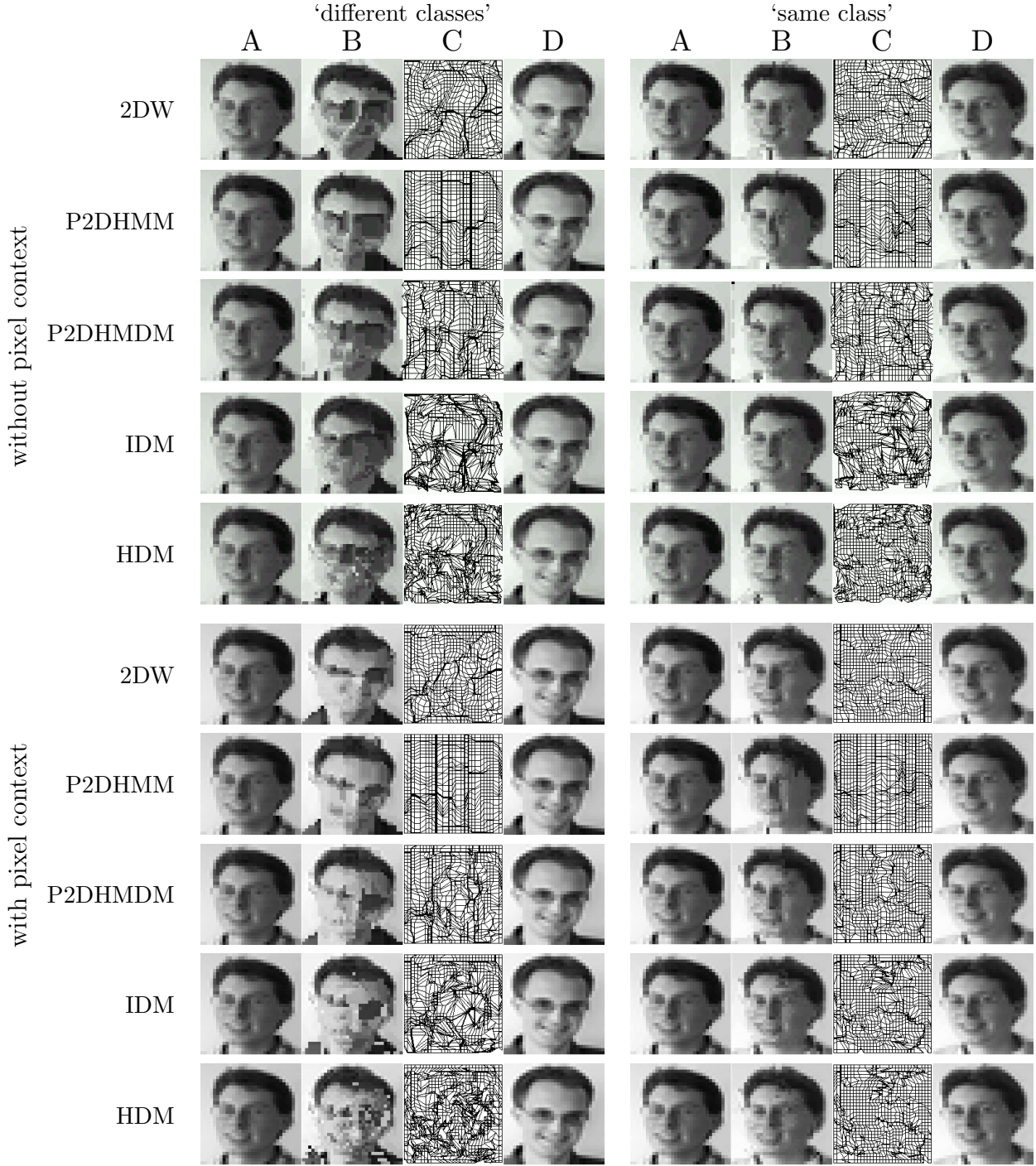


Figure 6.2: Examples of nonlinear matching applied to face images for illustration. Each group of four images consists of (A) test image, (B) transformed reference image, (C) displacement grid, (D) reference image. The left group shows images of 'different classes', while the right group shows images of the 'same class'. The upper examples were determined using only the pixel values, while the lower examples used the pixel context for determining the matching.

error rates for the tasks as discussed later. But we have the hope that the use of local image context will then lead to better discriminative performance also for the models with less constraints, which is confirmed by the experiments presented later on.

The second five examples show the results for the case of using local context for the matching as detailed below. Note that the matchings for the same class remain very accurate, while the matching for the different classes are visually not as good as before, especially for the models with fewer constraints as the IDM. Note also that the displacement grid is more homogeneous for the matchings of the same class.

6.2.1 Decision rule

In this chapter the main emphasis lies on the effect of the different distance functions resulting from the various deformation models. We therefore choose a simple decision rule for the classification process. We use the nearest neighbor decision rule

$$r(A) = \arg \min_k \left\{ \min_{n=1, \dots, N_k} d(A, B_{kn}) \right\}. \quad (6.2)$$

In most experiments, the 3-NN classifier was used, which differs from the decision of the 1-NN classifier if the second and third nearest prototypes are both from the same class and that is not the class of the nearest prototype. In that case the class of the second and third nearest prototypes is chosen. Other groups also report good results for this choice of classifier.

The distance function used within the decision rule has a special structure, i.e. it results from a minimization over all allowed deformation mappings for the model used:

$$d(A, B) = \min_{(x_{11}^{IJ}, y_{11}^{IJ}) \in \mathcal{M}} \left\{ d'(A, B_{(x_{11}^{IJ}, y_{11}^{IJ})}) \right\} \quad (6.3)$$

Here, the minimization process is of varying computational complexity depending on the model used. It is largest for the true two-dimensional model (we will proceed to prove that the minimization in that case is NP-complete in the following sections) and smallest for the zero-order image distortion model. The simpler distance function $d'(\cdot, \cdot)$ used within the minimization usually is chosen to be the Euclidean distance over all vector components of all pixels, where the transformed reference image is compared with the test image:

$$d'(A, B_{(x_{11}^{IJ}, y_{11}^{IJ})}) = \sum_{i,j} \sum_u \|a_{ij}^u - b_{x_{ij}y_{ij}}^u\|^2 \quad (6.4)$$

6.2.2 Feature extraction and pixel level models

This section describes the features and distance measures used at the pixel level.

Extraction of pixel features

In first experiments that are described in more detail in [Gollan 03], we observed that the deformation models did not lead to good classification results when they were applied to the pixel values alone. The best result on the USPS corpus that was achieved using only gray values for the deformation models was 3.9% using the P2DHMM and an absolute Euclidean cost function with appropriate weight. In this experiment, virtual training data by shifts in

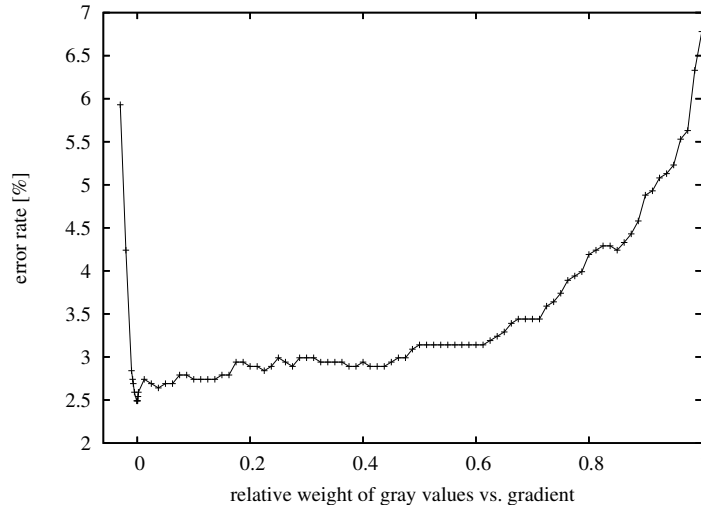


Figure 6.3: USPS P2DHMM error rate with respect to the weight of the gradient and gray value.

the 8-neighborhood was used. The improvement with respect to the Euclidean distance (no matching), which achieves an error rate of 4.8% with the virtual training data, is relatively small. The reason for this limited improvement is that too many unwanted deformations can also be modeled if we include only the gray values in the matching. We therefore investigated different possibilities to include more information at the pixel level, i.e. used $U > 1$:

One straightforward way to include the local image context is to use derivatives of the image values with respect to the image coordinates as computed by the horizontal and vertical Sobel-filter. These values have the additional advantage of invariance with respect to the absolute image brightness.

Now the question arises of how to weight the importance of the context information with respect to the image gray values. Figure 6.3 shows the error rate on the USPS corpus (using the P2DHMM) with respect to the relative weight of the gradient image (a relative weight of one means that only the gradient information is used). From the graph it is clear that best results are obtained when using only the gradient information.

To use gradient information is a fairly standard approach in many image processing and recognition contexts. It is for example also used by [Lowe 99] (histograms of gradient information around interest points) and by [Belongie & Malik⁺ 02] (histograms of contour point occurrence around selected contour points).

Of course, when using the first derivative, it is natural to ask if the second derivative could lead to similar improvements. Unfortunately, the additional use of the second derivative lead to only small improvements in the experiments performed and to a drop in performance in some cases. Therefore, we chose not to use higher order derivatives.

A second way to include the local image context is to use local sub images that are extracted around the regarded pixel, e.g. of size 3×3 pixels. Naturally, the size of the optimal context depends on the resolution of the objects in the images. For handwritten character recognition we observed that a local context of 3×3 pixels lead to the best results across a number of different tasks and the performance also generalized well for medical images.

The contexts can be extracted from the gray values in the image, thus leading to a vector

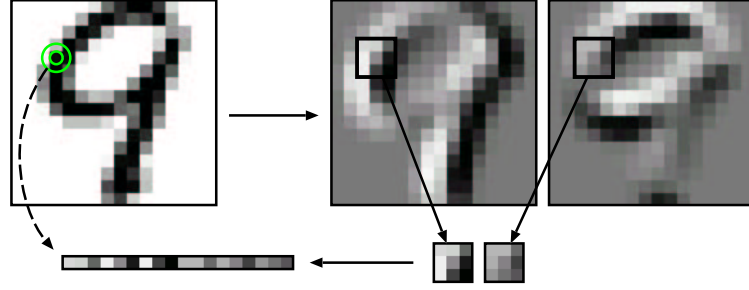


Figure 6.4: Illustration of the context extraction at the pixel level for the use within the deformation models. The original image (left) is processed and the horizontal and vertical gradient images are computed using the Sobel operator (right). For each pixel (marked with a circle), a local sub-window of size 3×3 pixels is extracted from both gradient images (bottom right) and interpreted as a vector of dimension $U = 18$ (bottom left).

of dimension $U = 9$. This already leads to improvements in the classification. However, we can combine the two methods and extract the contexts from the gradient images, which changes the value of an image pixel to be a vector of dimension $U = 2 \cdot 3 \cdot 3 = 18$. This combination lead to consistently better results and was therefore adopted in all experiments presented in the domain of character recognition in the following. Figure 6.4 illustrates the process of the feature extraction: the horizontal and vertical gradient images are calculated using the Sobel-filter, then the local 3×3 context is extracted in the gradient images and the values are stacked onto each other to form the pixel-level feature vector.

To give an example of the dependency on the original image (although the Sobel operator is a standard method) we present the dependency of the center pixel of the horizontal Sobel gradient, which is the fifth of 18 values. Using the standard Sobel operator, the new pixel values are computed directly from the image $\{a_{ij}\} \mapsto \{\tilde{A}_{ij}^5\}$:

$$\begin{aligned} \tilde{A}_{ij}^5 = & \quad a_{i+1,j-1} - a_{i+1,j+1} \\ & + 2 a_{i,j-1} - 2 a_{i,j+1} \\ & + a_{i-1,j-1} - a_{i-1,j+1} \end{aligned}$$

All results presented in the following were obtained using 3×3 sub images of the horizontal and vertical gradient images only, i.e. 18-dimensional vectors as pixel values, as these settings showed the overall best performance among those investigated.

It is interesting to compare the local context extraction used in the shape context approach [Belongie & Malik⁺ 02] to the one used here. The shape context approach uses histograms of the occurrence of contour points as descriptors for the context of selected contour points. To determine the contours in the image, the gradients are usually computed in a step before. For example, one suitable measure of the edge/contour strength is the sum of the squared horizontal and vertical gradient. This means that shape contexts capture the local context as the distribution of points sampled from areas of high absolute image gradient. Furthermore, the shape contexts are usually much larger than the local context extracted in the work presented here; it can encompass the whole image. On the other hand, the local context used here also includes the sign of the gradient, describing its direction, and can therefore distinguish between contours that occur at changes from foreground to

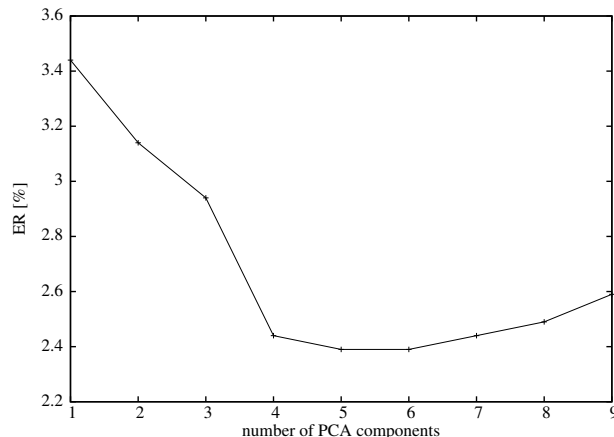


Figure 6.5: USPS IDM results using PCA components calculated on 3×3 contexts. The graph shows the error rate with respect to the number of used PCA components.

background or vice versa. Furthermore, we do not rely on preprocessing (contour detection) to determine contour points first. While contour detection does not pose a problem for handwritten characters or images of icons, it seems questionable if it could be feasible to perform this preprocessing steps on images without a sufficiently clear contour, as e.g. the case in the domain of radiograph categorization. Note that we became aware of the partial similarity with the shape context methodology only after most of the experiments described here were performed.

The choice to use the image gradient as a feature for the matching process was originally a heuristic. Therefore, we compared the performance of the Sobel operator to the use of the PCA on local sub images. The PCA transformation was estimated on all local 3×3 sub images of the USPS corpus and then the error rate of the image distortion model with respect to the number of used PCA components was observed. The dimensionality of the pixel feature vector was therefore $U = 9, 18, 27, \dots, 81$ in the experiments. Figure 6.5 shows the achieved error rates. The lowest error rate of 2.4% can be observed for the use of five or six PCA components. The same error rate is also achieved when using the two components that are obtained from the horizontal and vertical Sobel operator. We can conclude that it is possible to use a more general framework of pixel level feature extraction as given by the PCA, but it does not seem to provide advantages over the heuristic use of the image gradients. On the contrary, the PCA requires the additional computation of the PCA vectors, which is not necessary for the Sobel operators, and it requires to use a larger number of components.

One open question remains in this context: it would be interesting to investigate which methods could be suitable to extract *discriminative* local information from the image. To do so, we would need to use a representation as a basis to discriminate local regions in the image for recognition, for example clusters of local regions. This could be achieved by using models that contain explicit states as in hidden Markov models. As this would require a complete change of the setup used so far (use models with explicit states instead of reference images that define the models and state emission characteristics), we did not pursue this direction of research in this work.

One possibility to advance in this direction could be to use convolutional neural networks [LeCun & Boser⁺ 89, Simard & Steinkraus⁺ 03]. In convolutional neural networks the weights of the first layers of the neural network are tied such that at each position the

Table 6.2: USPS error rates [%] for the IDM using the tangent distance for the distance calculations between local 3×3 sub-images in comparison to the Euclidean distance. The error rates are obtained using the 100 closest images with respect to the Euclidean distance without image matching. The tangent distance uses vertical and horizontal translation vectors.

local context features	local distance	
	Euclidean distance	tangent distance
3×3 image context	3.5	3.2
3×3 gradient context	2.6	2.5

same representation of the local context of a pixel is extracted. However, these weights are trained during the training phase that is governed by a discriminative criterion. Therefore, the resulting local feature extraction is trained discriminatively via backpropagation of the criterion gradient to the input layer and results in discriminatively trained feature extraction units. However, this approach is sufficiently different from the matching-of-prototypes approach considered in this work to make it difficult to transfer the results to our approach.

Tangent distance for local context

If we are matching images by aligning local sub-images, it is natural to ask whether we can gain performance by again matching these sub-images to each other. For the patch-based approaches we can answer this question positively as discussed in Chapter 7, although work on the effects of this approach still has to be done. Here, we will briefly summarize the effect of using the tangent distance for matching of the 3×3 pixel contexts within the IDM for handwritten digit recognition. This has the effect that we additionally allow for sub-pixel displacements, while the pixel-to-pixel matching approach followed here only allows for displacements of a discrete number of pixels.

Table 6.2 shows the resulting error rates for the USPS corpus in comparison to the error rates obtained using the Euclidean distance. The experiments use a pre-selection of the 100 closest prototypes with respect to the Euclidean distance without image matching. We can observe small improvements in the performance for the use of contexts of the original gray values and for the use of the gradients. Note that it is necessary to use the same deformation for both horizontal and vertical gradients in order to obtain a valid tangent distance for the gradients, it is not sufficient to compute the tangent distance on both layers independently. Instead, the correct result can be achieved by concatenating the two layers for both the images and the tangent vectors. Unfortunately, we were not able to improve our best results on the USPS or the MNIST corpus by applying the tangent distance to the local contexts.

In the experiments we first used only the tangent vectors corresponding to vertical and horizontal translation. We then discovered that no improvements could be obtained by using more than these two vectors, which is probably due to the small dimensionality of the local sub-images.

The use of the tangent distance as a local distance measure improved the results for the recognition of sign language words considerably as described in Section 6.8.2 below.

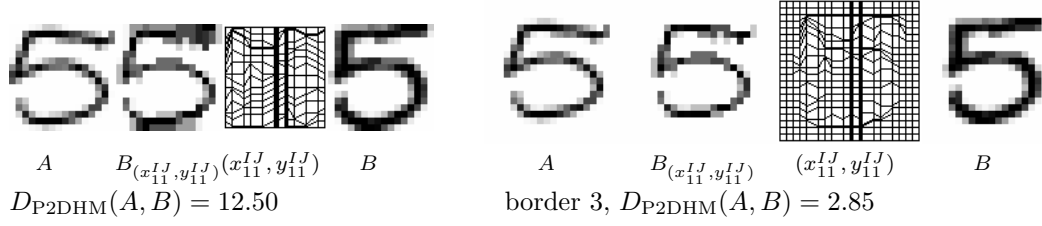


Figure 6.6: Example of the use of image border padding. A border of three pixels width is added in the second case which relaxes the constraints of the matching using the P2DHMM and leads to a smaller distance. In the left matching many border pixels of the reference image are foreground pixels and the model cannot match these correctly to the background pixels on the border of the test image. This is avoided by padding the image with background pixels (right).

Preprocessing

Some methods of preprocessing remain to be discussed. The first is the use of padding, i.e. the enlargement of the images. This preprocessing has several aims: we want to allow for the extraction of pixel contexts at the border of the images and we want to relax the border constraints of the deformation models that are based on the HMM concept. Here, the basic model demands that the borders of the images must be matched onto each other, which is relaxed most easily by enlarging the images by a few pixels. This is especially effective in the area of handwritten character recognition, because here we can extend the image with pixels of the background color, without introducing any additional unwanted image gradients, because the background is known and homogeneous. This is not the case for example for radiograph classification. In that domain the simplest padding is to enlarge the image with repetitions of the border values or to use the mean gray value for the padding. Padding is also used for patch-based classification of images as discussed in Chapter 7. Figure 6.6 shows an example of a better matching that is possible after padding the image with a border of three pixels width.

The second preprocessing method relevant in the context of the matching discussed here is the scaling of the images. During the experiments for handwritten character recognition we observed that a size smaller than about 16×16 pixels lead to a poorer performance. Therefore, the images of the corpora that contain images of a smaller size were scaled up to this size using spline interpolation. Figure 6.7 shows a comparison of spline interpolation to bilinear interpolation for a handwritten digit ‘9’. Spline interpolation lead to the best results in tests as reported by [Gollan 03].

For other corpora, like the IRMA data, it is helpful to down-scale the images from very large sizes, because sizes larger than a certain number of pixels do not lead to better results but only increase the computational load. For example, for the IRMA data heights of more than 32 pixels did not improve the results such that a preprocessing to this height was applied.

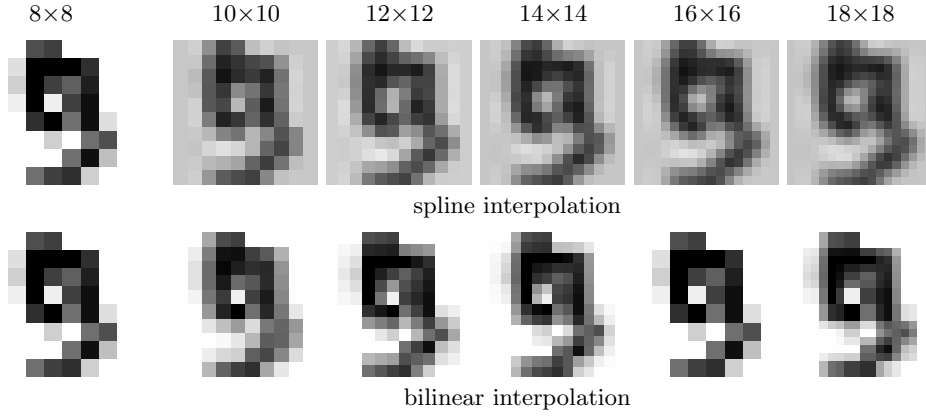


Figure 6.7: This example shows the scaling up from 8×8 pixels to larger sizes using spline (top row) and linear (bottom row) interpolation. The background color changes for the spline interpolation because negative values can occur and the images are re-normalized to the full gray scale range.

Thresholding of local distances

The use of thresholding of local distances is a fairly common procedure that was also used in this work, cp. e.g. [Würtz 97]. [Vasconcelos & Lippman 98] write with respect to the subject of thresholding in image classification that “It is well-known, that a few (maybe even one) outliers of high leverage are sufficient to throw mean squared error estimators completely off-track.” and propose — similar to the approach taken here — to substitute the square function by a functional, that weighs large errors less heavily, then propose to use a thresholding function for that functional. The rationale behind thresholding is that a few image regions with large differences as possibly caused by noise or partial occlusion should not enlarge the distance too much.

A threshold is introduced by replacing the local distance component in (6.4) with a thresholded version

$$d'(A, B_{(x_{11}^{IJ}, y_{11}^{IJ})}) = \sum_{i,j} \min \left\{ d_0, \sum_u \|a_{ij}^u - b_{x_{ij}y_{ij}}^u\|^2 \right\}, \quad (6.5)$$

where the maximum local distance d_0 is usually chosen to be about five percent of the maximum possible distance. This procedure alleviates the problem that otherwise each local distance contributes to the overall match equally, which should not be the case in the presence of occlusions or locally strong distortions, for example. Especially on the IRMA database the thresholding proved to be very effective.

A thresholded Euclidean distance also arises from a theoretical point of view if we assume a mixture distribution of a Gaussian distribution of features and an underlying uniform noise distribution. It is also observed in various applications that long-tailed distributions comparable to the one discussed here have better properties in the presence of noise and outliers.

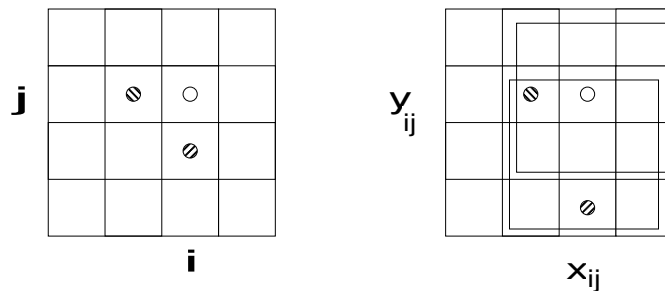


Figure 6.8: Example of the constraints imposed by neighboring pixels by the 2DW model. Given the displacement of the left neighbor, the center pixel (hollow circle) can be mapped to one of the nine pixels within the upper square; given the lower neighbor, the mapping is restricted to the nine pixels within the lower square. Combining both, the center pixel can be mapped onto the six pixels within the intersection of the two squares without violating any constraints.

6.3 Second-order: two-dimensional model

We begin the discussion of the two-dimensional matching algorithms with the true two-dimensional model. The two-dimensional HMM or two-dimensional warping (2DW) is an extension to two dimensions of the (0,1,2)- or (loop, jump, skip)-HMM that is frequently used e.g. in speech recognition or the recognition of continuous handwriting. The model ensures monotonicity (no backward steps) and continuity (no large jumps) of the displacement grid. In contrast to the one-dimensional case, which allows polynomial solutions, the minimization of this true 2D model is NP-complete as discussed in Section 6.4. Therefore, approximation algorithms are used as e.g. dynamic programming with beam search [Uchida & Sakoe 98], or simulated annealing. The following discussion of the model and the dynamic programming algorithm is largely based on the works of S. Uchida and colleagues [Uchida & Sakoe 98, Uchida & Sakoe 99b, Uchida & Sakoe 99a, Uchida & Sakoe 00a].

The 2DW model results from imposing the following constraints on the image deformation mappings $(x_{11}^{IJ}, y_{11}^{IJ})$ allowed within the matching for recognition (6.3) as summarized before in Table 6.1. We first impose the border constraints that image borders should be matched on image borders: $x_{1j} = 1, x_{Ij} = X, y_{i1} = 1, y_{iJ} = Y$. This constraint can be relaxed most easily by padding the images with background pixels. Secondly, we require that horizontally adjacent pixels should not be matched onto pixels that deviate from the relative position in the original image by more than one pixel. To do so we need two constraints, one in the horizontal direction, $x_{i+1,j} - x_{ij} \in \{0, 1, 2\}$, and one in the vertical direction, $x_{i,j+1} - x_{ij} \in \{-1, 0, 1\}$. The same constraints are imposed for vertically adjacent pixels: $y_{i,j+1} - y_{ij} \in \{0, 1, 2\}$, $y_{i+1,j} - y_{ij} \in \{-1, 0, 1\}$. Figure 6.8 show an example of the constraints imposed on the mapping of a pixel, if the mappings of its left and lower neighbor are already determined. This point of view is used in the dynamic programming algorithm described below.

Uchida and colleagues introduced a dynamic programming algorithm to determine optimal matchings under the discussed constraints. We will briefly discuss and illustrate this algorithm here, where the illustration is based on [Gollan 03].

The dynamic programming algorithm proceeds along the test image pixels in the order $(1, 1), (1, 2), (1, 3), \dots, (1, J), (2, 1), (2, 2), (2, 3), \dots, (I, J)$. In each of these stages, the algo-

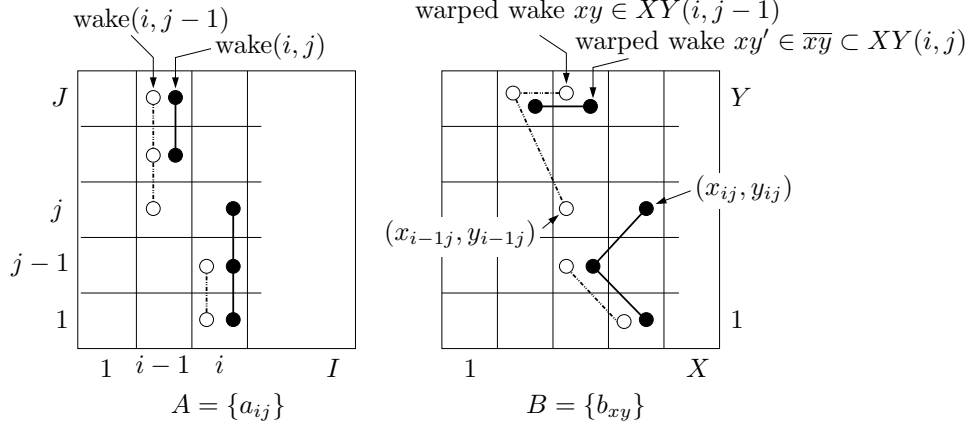


Figure 6.9: Illustration of two wakes and corresponding possible warped wakes of two subsequent stages of the 2DW dynamic programming algorithm.

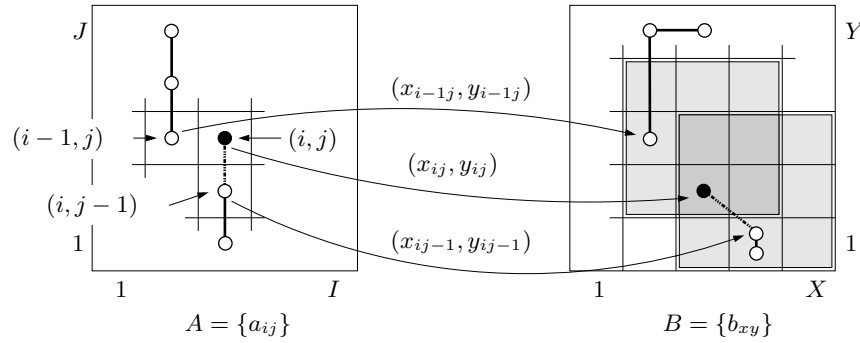


Figure 6.10: Illustration of the mapping restrictions for a pixel at stage (i, j) , given a warped wake. The warped wake determines the mappings of the left and lower neighbor of the pixel at (i, j) and therefore the possible mappings of the pixel itself, which are given by the overlapping 3×3 areas.

algorithm can recombine hypotheses that share the same assignment of pixels for one column of pixels. That is, in each stage (i, j) each state of the dynamic programming algorithm is described by the assignment of pixels

$$\begin{aligned}
 xy_{ij}^J : (i-1, j+1) &\mapsto (x_{i-1, j+1}, y_{i-1, j+1}) \\
 (i-1, j+2) &\mapsto (x_{i-1, j+2}, y_{i-1, j+2}) \\
 &\vdots \\
 (i-1, J) &\mapsto (x_{i-1, J}, y_{i-1, J}) \\
 (i, 1) &\mapsto (x_{i, 1}, y_{i, 1}) \\
 &\vdots \\
 (i, j-1) &\mapsto (x_{i, j-1}, y_{i, j-1}) \\
 (i, j) &\mapsto (x_{i, j}, y_{i, j})
 \end{aligned}$$

This assignment is called a ‘warped wake’ and $XY(i, j)$ denotes the set of all warped wakes xy_{ij}^J at the position (i, j) . The name ‘warped wake’ is used, because it denotes the warping

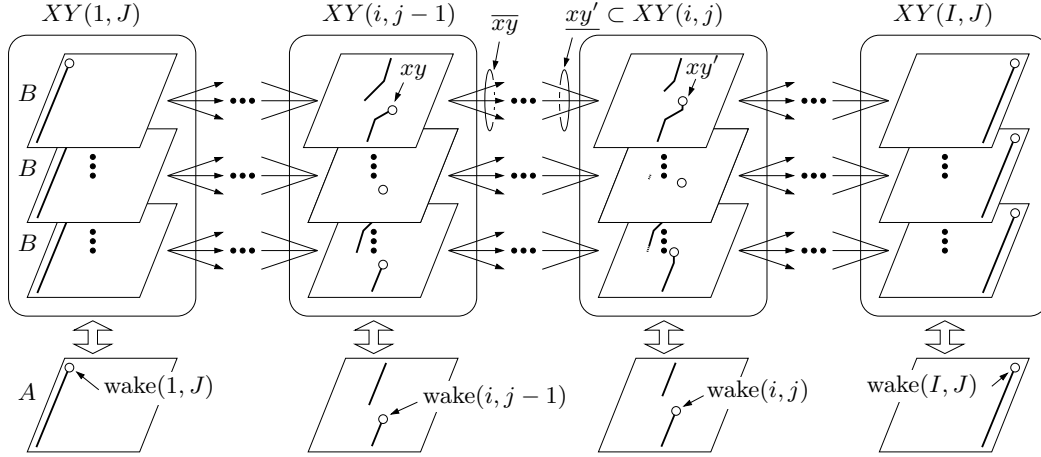


Figure 6.11: Illustration of the 2DW dynamic programming algorithm. Each column represents one stage of the algorithm which is associated with one pixel position (i, j) and the corresponding wake (below) and possible warped wakes (above). The algorithm proceeds from left to right and uses recombination of all incoming paths for each state, i.e. for each warped wake.

of a wake of J pixels following the moving position (i, j) . More formally,

$$\begin{aligned} XY(i, j) &= \{xy_{ij}^J\} \\ &= \left\{ \{x_{i'j'}, y_{i'j'}\} : (i' = i \quad \wedge \quad j' \in \{1, \dots, j\}) \vee \right. \\ &\quad \left. (i' = i - 1 \quad \wedge \quad j' \in \{j + 1, \dots, J\}) \right\} \end{aligned}$$

By $Pred(xy_{ij}^J)$ we denote the set of possible predecessors of xy_{ij}^J , i.e. all warped wakes in $XY(i, j - 1)$ that can be continued with an assignment of (i, j) without violating any of the constraints defined above. The auxiliary quantity $D(i, j, xy_{ij}^J)$ used in the dynamic programming algorithm is the minimal cost for an image matching up to (i, j) with the last J pixels aligned according to xy_{ij}^J .

$$D(i, j, xy_{ij}^J) = (a_{ij} - b_{xy_{ij}^J(i, j)})^2 + \min_{xy_{ij-1}^J \in Pred(xy_{ij}^J)} \{D(i, j - 1, xy_{ij-1}^J) + \mathcal{T}(\dots)\}$$

(The special case of $j = 1$ is handled analogously.)

The necessity to use such a wake as a representation of an isolating contour between the part of the image already processed and the part still to be processed is described in a related context in [Li & Najmi⁺ 00]: “The fact that in the 2-D case, only a sequence of states on a diagonal, rather than a single block, can serve as an ‘isolating’ element in the expansion of [...] causes computational infeasibility” Here, the authors use a diagonal structure instead of a line to minimize the interaction. This has the advantage that the interaction between diagonal neighbors may be considered as less strong than between direct neighbors, but the overall length of the diagonal is never less than that of the shorter image width.

Unfortunately the number of possible warped wakes is exponential in the image height J . It is necessary to store the complete information of the boundary of the warping performed

Table 6.3: Effect of the beam search threshold on the search space and the error rate for the USPS data. The experiments were performed using a maximum beam size of 500 and the closest 50 nearest neighbors with respect to the Euclidean distance.

threshold	average number of active warped wakes (states)	error rate[%]
0.2	166	4.1
0.5	272	3.3
0.8	337	3.1
∞	469	2.9

so far, because the possibility of future assignments depends on exactly these warpings at the boundary between already matched pixels and those pixels that still need to be matched. Figure 6.9 illustrates two wakes and warped wakes of two subsequent stages of the 2DW dynamic programming algorithm. Figure 6.11 illustrates the mapping restrictions for a pixel at stage (i, j) , given a warped wake. The warped wake determines the mappings of the left and lower neighbor of the pixel at (i, j) and therefore the possible mappings of the pixel itself. Figure 6.11 illustrates the dynamic programming algorithm.

Recognition results using the second-order model

We performed only few experiments using the second-order true two-dimensional deformation model, due to its very high computational complexity. All of these experiments used the USPS data, because it is comparatively small both in the number of samples and in the size of the images. The best result we obtained was an error rate of 2.7% using a constant beam size of 500 (larger beam sizes were not used due to the large computational requirements). It is likely that better results could be obtained using the true two-dimensional model with more search effort. But observing the very good error rates at much lower computational complexity of the simple models described in the following sections, we did not pursue this direction further.

Due to the large number of possible states (or warped wakes) it is impractical to perform an exhaustive search within the algorithm already for comparatively small image sizes. Especially for classification, when a large number of matches need to be determined, only a partial search can be performed. (For example, for a complete nearest neighbor run on the USPS database, $7,291 \cdot 2,007 = 14,633,037$ distances need to be calculated, each defined by a matching. Even if a hierarchical search is done and we compare each test image to only the 100 closest references, the number of matches is too large to make it practical to determine the exact best match.) To determine an approximate best match, the beam search technique is used, that is, in each stage of the algorithm, only the best hypothesis and those hypotheses with a score within a threshold of the best score (auxiliary quantity) are kept and expanded in the following stage. Table 6.3 shows the dependency of the error rate on the beam search threshold using a maximum beam size of 500 and the closest 50 nearest neighbors with respect to the Euclidean distance.

Figure 6.12 (top) shows examples in which the use of the distance resulting from 2DW and using only pixel grayvalues in a nearest neighbor classifier leads to correct classification, while a nearest neighbor without any matching leads to incorrect classification. Figure 6.12

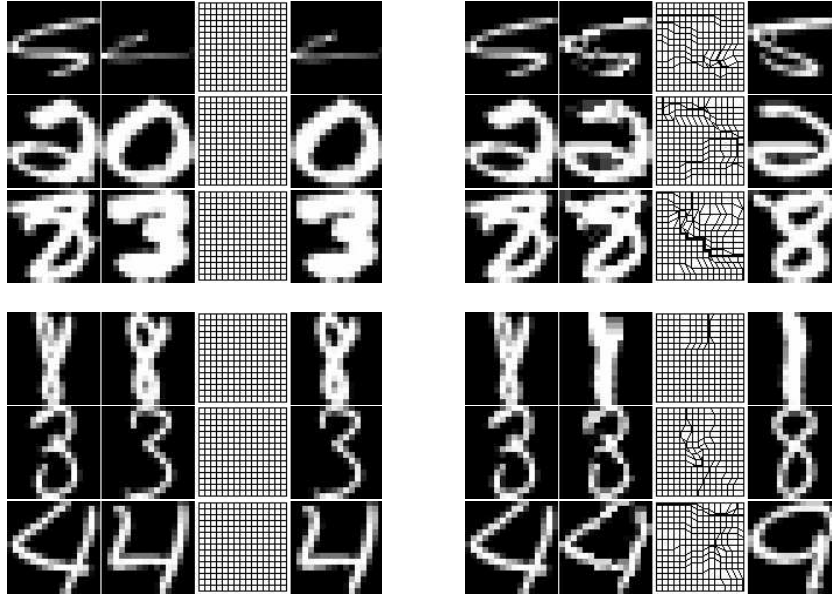


Figure 6.12: Examples of nearest neighbor classifications in which 2DW performs better (top) or worse (bottom) than the Euclidean distance without matching on the USPS corpus, respectively. Left: Euclidean nearest neighbors; right: nearest neighbors according to 2DW using only pixel grayvalues.

(bottom) shows examples of the opposite case, in which 2DW performs worse than no matching. Note that in these examples no pixel context has been used.

Note that there are other possible strategies to determine an approximation to the best matching deformation than dynamic programming with beam search. These include the following: simulated annealing is a common strategy to find approximative solutions to hard optimization problems. A more detailed description of the application of this method to the problem of character recognition is given in [Gollan 03]. Two other techniques that may be used are turbo-decoding [Perronnin & Dugelay⁺ 03] and piece-wise linear matching [Uchida & Sakoe 00b, Ronee & Uchida⁺ 01].

6.4 Two-dimensional matching is NP-complete

The general problem of two-dimensional image matching as described in the previous section is NP-complete. We will deviate from the discussion of the different deformation model in this section and prove this claim. For the readers that would like to skip this section, the discussion of the deformation models continues with Section 6.5 on page 125. The following proof has been presented first in [Keysers & Unger 03].

6.4.1 Introduction

In image recognition, a common problem is to match two given images, e.g. when comparing an observed image to given references. In that process, elastic image matching, 2D-warping [Uchida & Sakoe 98] or similar types of invariant methods [Keysers & Dahmen⁺ 00b] can be used. For this purpose, we can define cost functions depending on the distortion introduced

in the matching and search for the best matching with respect to a given cost function. Here we show that it is an algorithmically hard problem to decide whether a matching between two images exists with costs below a given threshold. We show that the Problem IMAGE MATCHING is NP-complete by means of a reduction from 3-SAT, which is a common method of demonstrating a problem to be intrinsically hard [Garey & Johnson 79]. This result shows the inherent computational difficulties in this type of image comparison, while interestingly the same problem is solvable for one-dimensional sequences in polynomial time, e.g. the dynamic time warping problem in speech recognition [Ney & Mergel⁺ 92].

This result has the following implications: researchers who are interested in an exact solution to this problem cannot hope to find a polynomial time algorithm, unless $P = NP$. Furthermore, one can conclude that exponential time algorithms as presented and extended by [Uchida & Sakoe 98, Uchida & Sakoe 99a, Uchida & Sakoe 99b, Uchida & Sakoe 00a, Uchida & Sakoe 00b] may be justified for some image matching applications. On the other hand this shows that those interested in faster algorithms – e.g. for pattern recognition purposes – are right in searching for sub-optimal solutions. One method to do this is the restriction to local optimizations or linear approximations of global transformations as presented in [Keysers & Dahmen⁺ 00b]. We can also restrict the constraints such that the minimization under these fewer constraints to the warping can be found in polynomial time, e.g. [Kuo & Agazzi 94]. Another possibility is to use heuristic approaches like simulated annealing or genetic algorithms to find an approximate solution. Furthermore, methods like beam search are promising candidates, as these are used successfully in speech recognition.

6.4.2 Definition of the image matching problem

Among the varieties of matching algorithms, we choose the one presented by [Uchida & Sakoe 98] as a starting point to formalize the Problem IMAGE MATCHING. Without loss of generality, let the images be given as square grids of size $M \times M$ with grayvalues (resp. node labels) from a finite alphabet $\mathcal{G} = \{1, \dots, G\}$. To define the problem, two distance functions are needed, one acting on grayvalues $d_g : \mathcal{G} \times \mathcal{G} \rightarrow \mathbb{N}$, measuring the match in grayvalues, and one acting on displacement differences $d_d : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{N}$, measuring the distortion introduced by the matching. For these distance functions we assume that they are monotonous functions (computable in polynomial time) of the commonly used squared Euclidean distance, i.e. $d_g(g_1, g_2) = f_1(\|g_1 - g_2\|^2)$ and $d_d(z) = f_2(\|z\|^2)$ with f_1, f_2 monotonously increasing. Now we call the following optimization problem the IMAGE MATCHING problem. Let $\mathcal{M} = \{1, \dots, M\}$.

Instance: The pair (A, B) of two images A and B of size $M \times M$.

Solution: A mapping function $f : \mathcal{M} \times \mathcal{M} \rightarrow \mathcal{M} \times \mathcal{M}$.

Measure:

$$\begin{aligned}
 c(A, B, f) = & \sum_{(i,j) \in \mathcal{M} \times \mathcal{M}} d_g(A_{ij}, B_{f(i,j)}) \\
 & + \sum_{(i,j) \in \{1, \dots, M-1\} \times \mathcal{M}} d_d\left(f\left((i, j) + (1, 0)\right) - \left(f(i, j) + (1, 0)\right)\right) \\
 & + \sum_{(i,j) \in \mathcal{M} \times \{1, \dots, M-1\}} d_d\left(f\left((i, j) + (0, 1)\right) - \left(f(i, j) + (0, 1)\right)\right)
 \end{aligned}$$

Goal: $\min_f c(A, B, f)$

In other words, the problem is to find the mapping from A onto B that minimizes the distance between the mapped grayvalues together with a measure for the distortion introduced by the mapping. Here, the distortion is measured by the deviation from the identity mapping in the two dimensions. The identity mapping fulfills $f(i, j) = (i, j)$ and therefore $f((i, j) + (x, y)) = f(i, j) + (x, y)$.

The corresponding *decision problem* is fixed by the following

Question: Given an instance of IMAGE MATCHING and a cost c' , does there exist a mapping f such that $c(A, B, f) \leq c'$?

In the definition of the problem some care must be taken concerning the distance functions. For example, if either one of the distance functions is a constant function, the problem is clearly in P (for d_g constant, the minimum is given by the identity mapping and for d_d constant, the minimum can be determined by sorting all possible matchings for each pixel by grayvalue cost and mapping to one of the pixels with minimum cost). But these special cases are not those we are concerned with in image matching in general.

We choose the matching problem as described by [Uchida & Sakoe 98] to complete the definition of the problem. Here, the mapping functions are restricted by continuity and monotonicity constraints: the deviations from the identity mapping may locally be at most one pixel (i.e. limited to the eight-neighborhood with squared Euclidean distance less than or equal to 2). This can be formalized in this approach by choosing the functions f_1, f_2 as e.g.

$$f_1 = id, \quad f_2(x) = \text{step}(x) := \begin{cases} 0 & , \quad x \leq 2 \\ (10 \cdot G)^{M \cdot M} & , \quad x > 2 \end{cases}$$

6.4.3 Reduction from 3-SAT

3-SAT is a very well-known NP -complete problem [Garey & Johnson 79, p.259], where 3-SAT is defined as follows:

Instance: Collection of clauses $C = \{c_1, \dots, c_K\}$ on a set of variables $X = \{x_1, \dots, x_L\}$ such that each c_k consists of 3 literals for $k = 1, \dots, K$. Each literal is a variable or the negation of a variable.

Question: Is there a truth assignment for X which satisfies each clause $c_k, k = 1, \dots, K$?

The dependency graph $D(\Phi)$ corresponding to an instance Φ of 3-SAT is defined to be the bipartite graph whose independent sets are formed by the set of clauses C and the set of variables X . Two vertices c_k and x_l are adjacent iff c_k involves x_l or \bar{x}_l .

Given any 3-SAT formula Φ , we show how to construct in polynomial time an equivalent IMAGE MATCHING problem $\mathbf{1}(\Phi) = (A(\Phi), B(\Phi))$. The two images of $\mathbf{1}(\Phi)$ are similar according to the cost function (i.e. $\exists f : c(A(\Phi), B(\Phi), f) \leq 0$) iff the formula Φ is satisfiable. We perform the reduction from 3-SAT using the following steps:

- From the formula Φ we construct the dependency graph $D(\Phi)$.
- The dependency graph $D(\Phi)$ is drawn in the plane.
- The drawing of $D(\Phi)$ is refined to depict the logical behavior of Φ , yielding two images $(A(\Phi), B(\Phi))$.

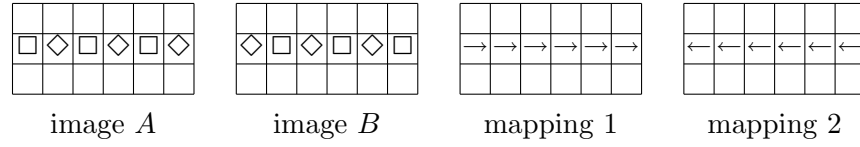


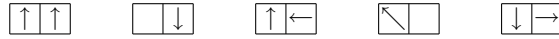
Figure 6.13: The straight connector component with two possible zero cost mappings

For this, we use three types of components: one component to represent variables of Φ , one component to represent clauses of Φ , and components which act as interfaces between the former two types. Before we give the formal reduction, we introduce these components.

Basic components

For the reduction from 3-SAT we need five components from which we will construct the instances for IMAGE MATCHING, given a Boolean formula in 3-DNF, resp. its graph. The five components are the building blocks needed for the graph drawing and will be introduced in the following, namely the representations of connectors, crossings, variables and clauses. The connectors represent the edges and have two varieties, straight connectors and corner connectors. Each of the components consists of two parts, one for image A and one for image B, where blank pixels are considered to be of the ‘background’ color.

We will depict possible mappings in the following using arrows indicating the direction of displacement (where displacements within the eight-neighborhood of a pixel are the only cases considered). Blank squares represent mapping to the respective counterpart in the second image. For example, the following displacements of neighboring pixels can be used with zero cost:



On the other hand, the following displacements result in costs greater than zero:



Figure 6.13 shows the first component, the straight connector component, which consists of a line of two different interchanging colors, here denoted by the two symbols \diamond and \square . Given that the outside pixels are mapped to their respective counterparts and the connector is continued infinitely, there are two possible ways in which the colored pixels can be mapped, namely to the left (i.e. $f(2, j) = (2, j - 1)$) or to the right (i.e. $f(2, j) = (2, j + 1)$), where the background pixels have different possibilities for the mapping, not influencing the main property of the connector. This property, which justifies the name ‘connector’, is the following: It is not possible to find a mapping, which yields zero cost where the relative displacements of the connector pixels are not equal, i.e. one always has $f(2, j) - (2, j) = f(2, j') - (2, j')$, which can easily be observed by induction over j' . That is, given an initial displacement of one pixel (which will be ± 1 in this context), the remaining end of the connector has the same displacement if overall costs of the mapping are zero. Given this property and the direction of a connector, which we define to be directed from variable to clause, we can define the state of the connector as carrying the ‘true’ truth value, if the displacement is 1 pixel in the direction of the connector and as carrying the ‘false’ truth value, if the displacement is -1 pixel in the direction of the connector. This property then

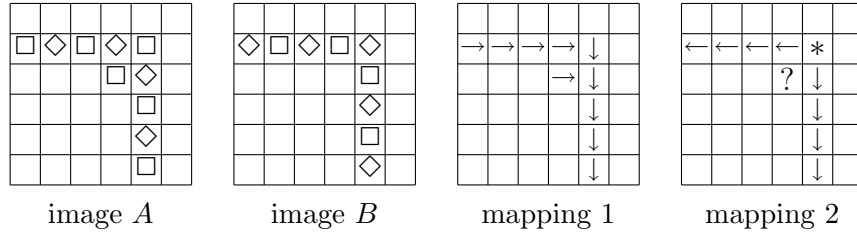


Figure 6.14: The corner connector component and two example mappings

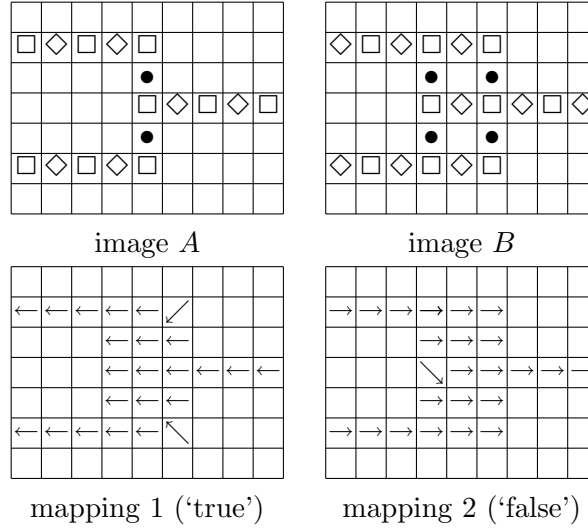


Figure 6.15: The variable component with two positive and one negated output and two possible mappings (for true and false truth value)

ensures that the truth value transmitted by the connector cannot change at mappings of zero cost.

For drawing of arbitrary graphs, clearly one also needs corners, which are represented in Figure 6.14. By considering all possible displacements which guarantee overall cost zero, one can observe that the corner component also ensures the basic connector property. For example, consider the first depicted mapping, which has zero cost. On the other hand, the second mapping shows, that it is not possible to construct a zero cost mapping with both connectors ‘leaving’ the component. In that case, the pixel at the position marked ‘?’ either has a conflict (that is, introduces a cost greater than zero in the criterion function because of a mapping mismatch) with the pixel above or to the right of it, if the same color is to be met and otherwise, a cost in the grayvalue mismatch term is introduced.

Figure 6.15 shows the variable component, in this case with two positive (to the left) and one negated output (to the right) leaving the component as connectors. Here, a fourth color is used, denoted by \bullet . This component has two possible mappings for the colored pixels with zero cost, which map the vertical component of the source image to the left or the right vertical component in the target image, respectively. (In both cases the second vertical element in the target image is not a target of the mapping.) This ensures ± 1 pixel relative displacements at the entry to the connectors. This property again can be deduced

by regarding all possible mappings of the two images. The property that follows (which is necessary for the use as variable) is that all zero cost mappings ensure that all positive connectors carry the same truth value, which is the opposite of the truth value for all the negated connectors. It is easy to see from this example how variable components for arbitrary numbers of positive and negated outputs can be constructed.

Figure 6.16 shows the most complex of the components, the clause component. The component consists of two parts. The first part is the horizontal connector with a ‘bend’ in it to the right. This part has the property that cost zero mappings are possible for all truth values of x and y with the exception of two ‘false’ values. This ‘two input disjunction’ can be extended to a three input disjunction using the part in the lower left. If the z connector carries a ‘false’ truth value, this part can only be mapped one pixel downwards at zero cost. In that case the junction pixel (the fourth pixel in the third row) cannot be mapped upwards at zero cost and the ‘two input clause’ behaves as described above. On the other hand, if the z connector carries a ‘true’ truth value, this part can only be mapped one pixel upwards at zero cost, and the junction pixel can be mapped upwards, thus allowing both x and y to carry a ‘false’ truth value in a zero cost mapping. Thus there exists a zero cost mapping of the clause component iff at least one of the input connectors carries a ‘true’ truth value.

The described components are already sufficient to prove NP-completeness by reduction from PLANAR 3-SAT (which is an NP-complete sub-problem of 3-SAT where the additional constraints on the instances is that the dependency graph is planar), but in order to derive a reduction from 3-SAT, we also include the possibility of crossing connectors.

Figure 6.17 shows the connector crossing, whose basic property is to allow zero cost mappings iff the truth values are consistently propagated. This is assured by a color change of the vertical connector and a ‘flexible’ middle part, which can be mapped to four different positions depending on the truth value distribution.

Reduction

Using the previously introduced components, we can now perform the reduction from 3-SAT to IMAGE MATCHING.

Proof of the claim that IMAGE MATCHING problem is NP-complete:

Clearly, the IMAGE MATCHING problem is in NP since, given a mapping f and two images A and B , the computation of $c(A, B, f)$ can be done in polynomial time. To prove NP-hardness, we construct a reduction from the 3-SAT problem. Given an instance of 3-SAT we construct two images A and B , for which a mapping of cost zero exists iff all the clauses can be satisfied.

Given the dependency graph D , we construct an embedding of the graph into a two-dimensional pixel grid, placing the vertices on a large enough distance from each other (say $100(K + L)^2$). This can be done using well-known methods from graph drawing, see e.g. [di Battista & Eades⁺ 99]. From this image of the graph D we construct the two images A and B , using the components described above. Each vertex belonging to a variable is replaced with the respective parts of the variable component, having a number of leaving connectors equal to the number of incident edges under consideration of the positive or negative use in the respective clause. Each vertex belonging to a clause is replaced by the respective clause component, and each crossing of edges is replaced by the respective crossing component. Finally, all the edges are replaced with connectors and corner connectors, and the remaining pixels inside the rectangular hull of the construction are set to the background

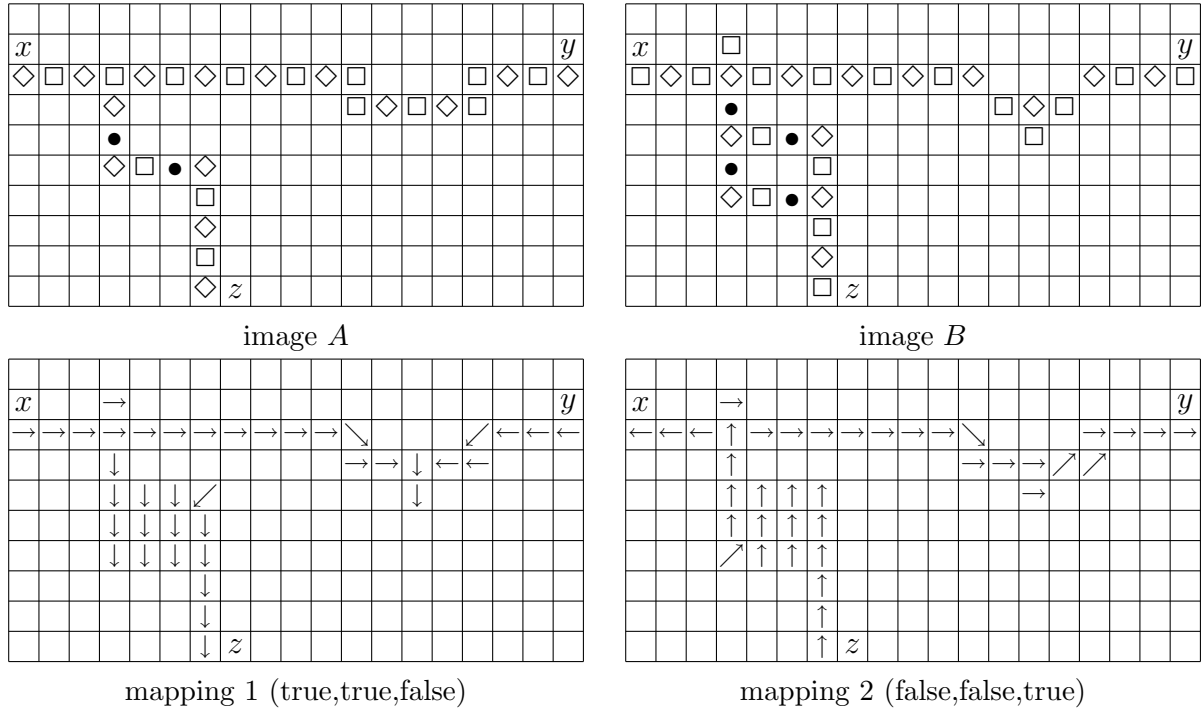


Figure 6.16: The clause component with three incoming connectors x, y, z and zero cost mappings for the two cases (true, true, false) and (false, false, true)

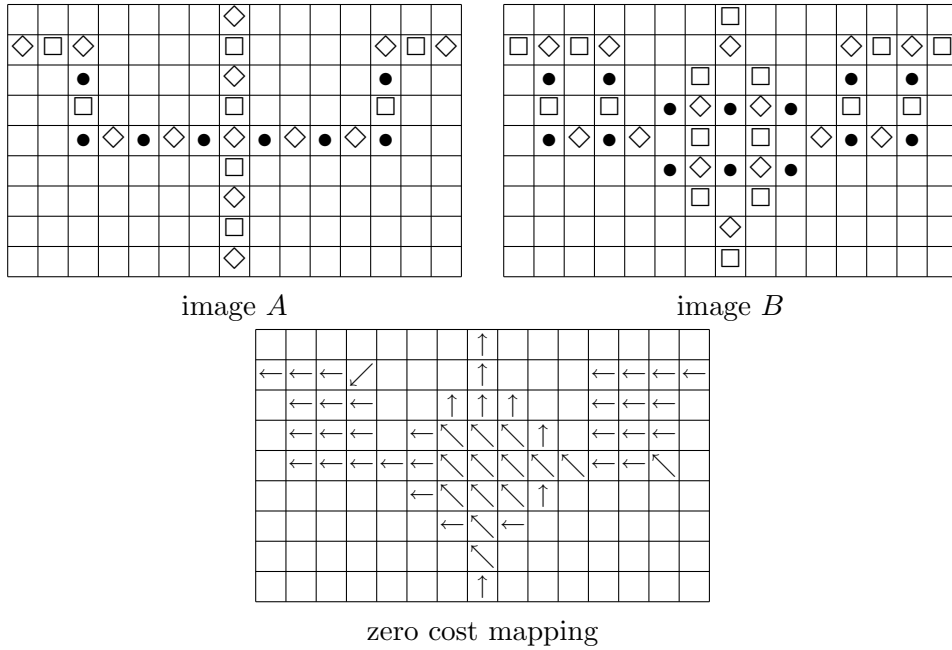


Figure 6.17: The connector crossing component and one zero cost mapping

grayvalue. Clearly, the placement of the components can be done in such a way that all the components are at a large enough distance from each other, where the background pixels act as an ‘insulation’ against mapping of pixels, which do not belong to the same component. It can be easily seen, that the size of the constructed images is polynomial with respect to the number of vertices and edges of D and thus polynomial in the size of the instance of 3-SAT, at most in the order of $(K + L)^2$. Furthermore, it can obviously be constructed in polynomial time, as the corresponding graph drawing algorithms are polynomial.

Let there exist a truth assignment to the variables x_1, \dots, x_L , which satisfies all the clauses c_1, \dots, c_K . We construct a mapping f , that satisfies $c(f, A, B) = 0$ as follows.

For all pixels (i, j) belonging to variable component l with $A(i, j)$ not of the background color, set $f(i, j) = (i, j - 1)$ if x_l is assigned the truth value ‘true’, set $f(i, j) = (i, j + 1)$, otherwise. For the remaining pixels of the variable component set $f(i, j) = (i, j)$ if $A(i, j) = B(i, j)$, otherwise choose $f(i, j)$ from $\{(i, j + 1), (i + 1, j + 1), (i - 1, j + 1)\}$ for x_l ‘false’ respectively from $\{(i, j - 1), (i + 1, j - 1), (i - 1, j - 1)\}$ for x_l ‘true’, such that $A(i, j) = B(f(i, j))$. This assignment is always possible and has zero cost, as can be easily verified.

For the pixels (i, j) belonging to (corner) connector components, the mapping function can only be extended in one way without the introduction of nonzero cost, starting from the connection with the variable component. This is ensured by the basic connector property. By choosing $f(i, j) = (i, j)$ for all pixels of background color, we obtain a valid extension for the connectors. For the connector crossing components the extension is straightforward, although here – as in the variable mapping – some care must be taken with the assignment of the background value pixels, but a zero cost assignment is always possible using the same scheme as presented for the variable mapping.

It remains to be shown that the clause components can be mapped at zero cost, if at least one of the input connectors x, y, z carries a ‘true’ truth value. For a proof we regard all seven possibilities and construct a mapping for each case. In the description of the clause component it was already argued that this is possible, and we omit the formalization of the argument here.

Finally, for all the pixels (i, j) not belonging to any of the components, we set $f(i, j) = (i, j)$, thus arriving at a mapping function which has $c(f, A, B) = 0$, as all colors are preserved in the mapping and no distortion of squared Euclidean distance greater than 2 is introduced.

On the other hand, let there exist a mapping f with $c(f, A, B) \leq 0$. This means that $c(f, A, B) = 0$ since there are only positive cost terms involved in the term for c . Furthermore, we can conclude, that

$$\begin{aligned} \forall (i, j) \in \mathcal{M} \times \mathcal{M} \quad & d_g(A_{ij}, B_{f(i,j)}) = 0 \\ \forall (i, j) \in \{1, \dots, M-1\} \times \mathcal{M} \quad & d_d\left(f\left((i, j) + (1, 0)\right) - \left(f(i, j) + (1, 0)\right)\right) = 0 \\ \forall (i, j) \in \mathcal{M} \times \{1, \dots, M-1\} \quad & d_d\left(f\left((i, j) + (0, 1)\right) - \left(f(i, j) + (0, 1)\right)\right) = 0 \end{aligned}$$

This means that f maps all pixels onto pixels of the same color and that the difference in mapping between neighboring pixels has at most squared Euclidean distance 2.

These facts now ensure a number of basic properties:

1. The basic elements of the components and the overall represented graph are mapped

Table 6.4: Open questions about the complexity with respect to the distortion measure.

distortion measure	$f_2(x) = \text{const.}$...	$f_2(x) = x$...	$f_2(x) = \text{step}(x)$
algorithmic complexity	$\in P$?	?	?	NP -complete

to their corresponding parts, because of the monotonicity and continuity assured by the maximum local mapping difference.

2. All the variable components are mapped such that the leaving connectors all carry consistent truth values (i.e. all the positive outputs carry the same truth value, and all the negative outputs carry the negated truth value).
3. All (corner, crossing, straight) connectors are mapped such that the basic connector property of propagated truth values is fulfilled, since no neighboring pixels are mapped into opposite directions.
4. Each clause component has at least one entering connector carrying a ‘true’ truth value, as otherwise a mapping with zero cost is not possible.

Now we construct an assignment of truth values to the variables x_1, \dots, x_L , which satisfies all the clauses c_1, \dots, c_K . We set x_l to ‘true’, if $f(i, j) = (i, j - 1)$ for the pixels of the corresponding variable component, which are not background pixels. We set x_l to ‘false’, otherwise. By means of the properties stated above we can conclude that this assignment satisfies all the clauses c_1, \dots, c_K , which concludes the proof.

6.4.4 Discussion

With the completion of the proof, some open questions remain. From the theoretical point of view, it is interesting to understand for which distortion measure the problem becomes NP -complete, as indicated by the question marks in Table 6.4 and in which way the result is influenced if we allow interpolation of the discrete pixel array. Another interesting theoretical question is if the problem remains NP -complete if we allow only less than the four colors used in the proof here.

In the presented proof, the displacements of the pixels in the mapping are at most one pixel for each zero-cost mapping between two constructed images. This may seem to be a restriction with respect to the general matching problem which allows larger displacements. But obviously the general problem is NP -hard, if already the class of problems which contains only these restricted versions is NP -hard.

With the proof that the problem IMAGE MATCHING is NP -complete, we have arrived at a fundamental result for a problem that has been studied thoroughly in the literature before. The works of Uchida and Sakoe deal with exactly the problem formalized here and they present a variety of exponential algorithms and restrictions to apply to the problem in order to make it tractable. [Ronee & Uchida⁺ 01] state that the computation “is in the exponential order of the image size”, but they do not elaborate on this statement. In the same manner, [Levin & Pieraccini 92] discuss a very similar problem and state that their algorithm is exponential in the general case, but do not go into more detail on this question. They propose to regard the lines of the images independently to keep the problem tractable. This is a similar approach to the use of pseudo 2-dimensional hidden Markov models as presented

in [Kuo & Agazzi 94], but the drawback is that any correspondence information between the deformation of the lines is disregarded, which may not be advisable in all domains. Also, [Li & Najmi⁺ 00] uses two-dimensional constraints for image segmentation and the authors discuss necessary approximations to their model because of its exponential complexity, then presenting a sub-optimal algorithm with polynomial time complexity. The same statement is also true for the algorithms discussed in [Devijver 86]. [Samaria 94] writes for the case of face recognition that “In the general case, the HMM would need to be 2D”, but also restricts the problem to a 1-dimensional one. [Moore 79] already discusses a similar task which is dealt with using a generalization of the 1-dimensional edit- or Levenshtein-distance, which also disregards local dependencies between certain lines. There are also works that view the image-warping problem based on a physical model of the grayvalue-surface that is represented by an image [Moghaddam & Nastar⁺ 96] and others that use variational approaches that are similar to the computation of optical flow [Schnörr & Weickert 00], which is inherently linked to the problem of image registration [Fischer & Modersitzki 01].

We may conclude that despite of the fundamental result that the IMAGE MATCHING problem is *NP*-complete there are a variety of methods to modify the basic problem such that it becomes tractable or to approximate the best solution. It is an interesting question to find out which of the possibilities is best suited for which practical application.

6.5 First-order: one- and pseudo two-dimensional models

The complexity of the true two-dimensional matching algorithms — even when using approximations like beam search or simulated annealing — is very large. Therefore, we consider models of a lower order, which is an approach that is fairly common in the literature, as discussed in Section 6.1.

6.5.1 Pseudo two-dimensional hidden Markov model

To proceed from two-dimensional models to models of lower order, we relax some of the constraints of the true two-dimensional case. Usually, this is done by allowing the columns of an image to be matched onto the reference image independently, which leads to the so-called pseudo two-dimensional hidden Markov model (P2DHMM) [Kuo & Agazzi 94]. The P2DHMM is obtained from the 2DW model by neglecting the dependencies between pixels of neighboring image columns. Here, the relative displacement in the vertical image direction between neighboring columns is neglected, and all pixels from one column are mapped onto the same target column.

Figure 6.18 shows a schematic overview of a pseudo two-dimensional hidden Markov model. The model consists of a one-dimensional hidden Markov model with so-called superstates that are themselves one-dimensional hidden Markov models. Table 6.1 includes a visualization of a possible beginning of a mapping that is allowed under this model. Each column of the test image is mapped onto a column of the reference image and the mapping within the columns is determined as for a one-dimensional hidden Markov model. This assignment of complete columns onto other columns implies two things: First, only complete columns of the reference image can be disregarded. Second, dependencies between the vertical displacements of the pixels in neighboring columns are ignored. The first of this limitation is somewhat alleviated by allowing additional deviations from the column assignment

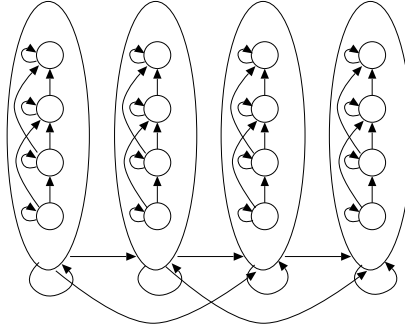


Figure 6.18: Schematic overview of a pseudo two-dimensional hidden Markov model. The model consists of a one-dimensional hidden Markov model with superstates that are themselves one-dimensional hidden Markov models.

as described below. The second limitation cannot be overcome easily without turning the minimization into a problem for which only exponential minimization algorithms are known.

The main direction of the model is usually chosen left-to-right. This seems to be the canonical form for western text, but is an arbitrary decision for general objects or for isolated characters. The algorithm can be applied just the same with the main direction of the model being top-to-bottom.

The interpretation using the concept of superstates shows that the P2DHMM can be seen as a straightforward extension of a one-dimensional HMM. The only difference to the alignment that is used in a classical one-dimensional HMM with left-to-right warping and regarding each column of the image as a feature vector is that the distance between the feature vectors is again determined by allowing one-dimensional nonlinear alignment. Such classical one-dimensional HMMs have found widespread use in the off-line recognition of continuous (handwritten) text, but are used mostly without the nonlinear alignment. Instead, elaborate preprocessing methods are often used and the matching of column vectors can also use other methods for invariance as e.g. the tangent distance [Toselli & Juan⁺ 04].

From the superstate representation of the P2DHMM we can also infer a strictly one-dimensional representation, in which the arcs between the superstates are replaced by arcs with long distance jumps between the first states of the column HMMs and which does not need the concept of the superstate. The input to such a model then is the one-dimensional sequence of pixel representations in the order $(1, 1), (1, 2), \dots, (1, J), (2, 1), \dots, (I, J)$. The P2DHMM is thus computationally equivalent to a one-dimensional HMM and therefore the minimization process has the same complexity as for an HMM. Nevertheless, the computational effort can be substantial and methods like beam search can be used to speed up the minimization.

6.5.2 Pseudo two-dimensional hidden Markov distortion model

To relax the constraint that always complete columns of the images must be matched we can allow additional distortions from the columns that are matched within the P2DHMM model. These distortions, e.g. distortions of one pixel displacement, are modeled to be independent of each other and we call the resulting model pseudo two-dimensional hidden Markov distortion model (P2DHMDM) [Gollan 03, Keysers & Gollan⁺ 04b, Keysers & Gollan⁺ 04a]. This name and abbreviation is rather lengthy but is supposed to show the relationship to

the P2DHMM of the model and also to the image distortion model (IDM), which will be discussed in Section 6.6.

Table 6.1 includes a visualization showing the first three columns of an example mapping between two images, where columns of the test image are mapped onto possibly distorted columns of the reference image. The mapping of one column of the test image is determined by dynamic programming similar to the case of the P2DHMM, only now a deviation from the target column is allowed, which is at most one pixel for this example and also for the experiments reported here.

This additional flexibility within the mapping of the columns leads to a slightly higher computational complexity, namely a factor three if at most one pixel distortion is allowed.

6.5.3 Recognition results using the pseudo-2D models

Using the P2DHMDM, we were able to improve the results for different recognition tasks with respect to the P2DHMM. Because the experiments with the simpler IDM preceded the experiments with the pseudo-2D models, we also refer to their results here, although the IDM will be discussed in Section 6.6 below.

- On the USPS task, the error rate of the P2DHMM is 2.7% and the error rate of the IDM is 2.4%, which can be reduced to 1.9% using the additional flexibility of the P2DHMDM, again using local pixel contexts [Keysers & Gollan⁺ 04b]. Thus, on the USPS database, the P2DHMDM performed better than the other models and gave the best results published so far.
- On the MNIST data, only a very small improvement from 0.54% for the IDM to 0.52% could be achieved. This statistically not significant improvement probably does not justify the increased complexity of the P2DHMDM over the IDM. The P2DHMDM result was achieved with a 3-NN, a preselection of the 100 closest references with respect to the Euclidean distance, an absolute warp range of three pixels, and 3×3 local pixel contexts of the derivatives. The warp range is included by adding to the constraints of the P2DHMDM the constraints of the IDM with warp range three, as shown in Table 6.1.
- On the UCI optical digits corpus, a scaling to 16×16 pixels using spline interpolation was performed. Here, P2DHMDM achieved an error rate of 0.8%, being the best published error rate.
- On the MCEDAR task, also a scaling to 16×16 pixels using spline interpolation was used, and again the P2DHMDM achieved the best published error rate of 3.3% using a 3-NN classifier.
- On the IRMA task of medical radiographs using local thresholding of distances and local contexts, the error rate could be reduced from 6.6% for the IDM and 5.7% for the P2DHMM to 5.3% for the P2DHMDM [Keysers & Gollan⁺ 04a]. This result is the best error rate reported so far and a remarkable relative improvement of about one third with respect to the best result reported before of 8.0% that included tangent distance [Keysers & Dahmen⁺ 03].

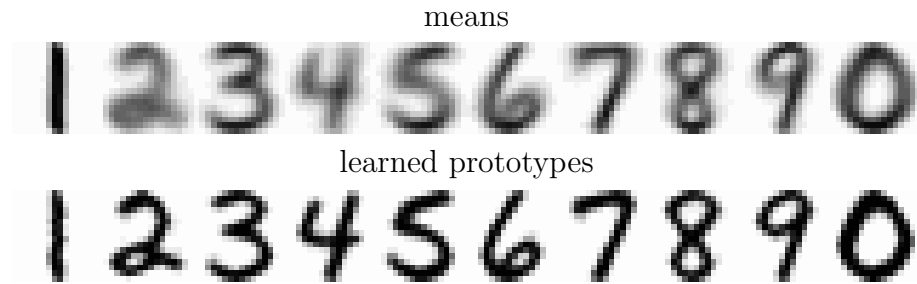


Figure 6.19: Prototypes for the ten classes of the USPS training set. The upper row shows the means using no matching, which are equivalent to the maximum likelihood estimates of a Gaussian density. The bottom row shows the prototypes obtained using the deformation model.

Table 6.5: USPS error rates [%] using the prototypes as shown in Figure 6.19 and no deformation or the P2DHMDM, respectively.

distance	means	learned prototypes
no deformation	18.6	26.1
P2DHMDM	25.3	4.9

6.5.4 Training of prototypes

Some experiments were performed for training of prototypes with the presented matching models on the USPS corpus. Using the training data, a reference model for each of the ten classes was estimated as described in the following and then used as a single reference for that class for testing. We describe the results at this position, because best results were obtained using the P2DHMDM for the matching to the test data.

The prototypes are initialized using the mean images of each class (see Figure 6.19). Then, each training image is matched to the prototype. Averaging the pixel values that the prototype pixels are matched to, a new set of prototypes is obtained, that better represents the training data and also takes into account the variability of the training images as modeled by the used distortion model. This procedure is repeated until the prototype images do not change any more. Figure 6.19 shows the mean images for all ten classes and the prototypes learned using the deformation model. Note that [Matsumoto & Uchida⁺ 04] proposed a similar method for prototype learning based on matching that was developed independently.

The error rates presented in Table 6.5 show that the P2DHMDM (the best model for this task) performs significantly better using the learned prototypes [Keysers & Gollan⁺ 04b]. Interestingly using only one prototype per class, the error rate is as low as 4.9%. The learned prototypes appear much less blurred than the mean images, as the image variability is compensated by the non-linear deformation model. On the other hand the corresponding mean images perform better if no deformation is used in the classifier, which means that the maximum likelihood estimate performs well in that case.

Other results that these error rates can be compared to (i.e. that also use a single prototype per class) are briefly described in the following: The discriminative training based on the maximum entropy framework achieves an error rate of 5.7% using second-order feature functions and no deformation model. The combination of the discriminative training and

the use of deformation models was attempted, but did not lead to good results. Using the empirical means as above and an estimated, 14-dimensional tangent vector subspace, the error rate obtained was 5.0%. [Hastie & Simard 98] report an error rate of 4.1% using a 12-dimensional trained subspace approach that also changes the means, but these results could not be confirmed by our experiments when re-implementing the approach [Keysers 00].

We can thus conclude that the use of the described models of image variability also gives state-of-the-art-results for the use of single prototypes per class.

6.5.5 Moore's algorithm

[Moore 79] presented an algorithm to compare two two-dimensional patterns allowing for spatial matching between two images. The algorithm is based on an extension of the well-known Levenshtein distance (or edit distance) to 2D. It also disregards the dependency between neighboring image lines and is therefore also a model of first-order. Experiments with Moore's algorithm for handwritten digit recognition are also presented in [Keysers 00]. We give a brief description here, following a short description of the Levenshtein distance.

The one-dimensional Levenshtein distance between two sequences a and b is defined as the minimum number of operations needed to transform one sequence into the other by using the operations substitution, deletion, and insertion of symbols in either sequence. The distance is best computed using dynamic programming. Informally, we have three possibilities for each matching step, which are to skip the last symbol of a or, skip the last symbol of b or, match the last symbol of a with the last symbol of b . For images, we replace symbols by pixel grayvalues and use the local squared Euclidean distance as a cost function for the match of two grayvalues.

Now, a straightforward generalization to two dimensions is difficult, because skipping of an element from a one-dimensional signal leaves a one-dimensional signal in a canonical way, but skipping of a pixel from a two-dimensional images does not leave an image. For an extension to 2D of the three possibilities we now have 15 possibilities (in general we have $2^{2D} - 1$, where D is the dimensionality of the structures to be matched), which involve the skipping of rows or columns. The resulting dynamic programming recurrence is very lengthy and can be found in [Moore 79]. Informally, we have the following possibilities: Given two images A and B , we can

1. skip the right column of A
2. skip the right column of B
3. skip the lower row of A
4. skip the lower row of B
5. skip the right column of A and the lower row of B
6. skip the lower row of A and the right column of B
7. match the right columns of A and B
8. match the lower rows of A and B
9. match the right columns of A and B , skip the lower row of A
10. match the right columns of A and B , skip the lower row of B
11. match the lower rows of A and B , skip the right column of A
12. match the lower rows of A and B , skip the right column of B
13. skip the lower row and the right column of A
14. skip the lower row and the right column of B
15. match the lower rows and the right columns of A and B

The one-dimensional sub-patterns are matches using the one-dimensional Levenshtein distance. Although this matching method is easily implemented and run, it seems somewhat difficult to interpret, because it is not clear, which deformations are allowed at which cost. Furthermore, no interdependence between the matched or skipped columns or rows is taken into account. As the original Levenshtein distance, the matching is symmetric. It allows skipping of pixels (and complete columns and rows) in both images equally.

For the USPS task the recognition performance with this matching method was not promising (only very small improvements with respect to the Euclidean distance could be obtained) and at the same time the computational complexity of the method is very large. Results obtained are presented in [Keysers 00]. Therefore, no additional results for other tasks were produced.

The method was recently rediscovered in [Lei & Govindaraju 04] but the authors do not mention that the model does not consider dependencies between neighboring columns. In their variant, a matching between rows and columns is always required, which could lead to different results. They present as result a warping of a few example images, which do not allow a meaningful judgment of the usefulness for recognition purposes and the authors also mention the high computational complexity of the method.

6.6 Zero-order: the image distortion model

If we further relax the constraints on the image mapping functions by eliminating all relative constraints and keeping only the absolute constraints, we arrive at zero-order models of image variability. Their advantage is that the minimization process is computationally much simpler and at the same time high recognition performance can be achieved when using the appropriate local image context representation. In this section we discuss the simple zero-order image distortion model. In Section 6.7 we will have a look at an extension of this model that uses the Hungarian algorithm to match pixel representations and thus achieves a more homogeneous pixel displacement grid.

One successful and conceptually simple method for determining an image matching is to use a zero-order model that completely disregards dependencies between the pixel mappings. An informal description of the model is the following: for each pixel in the test image, determine the best matching pixel within a region of size $w \times w$ at the corresponding position in the reference image and use this match. The formal constraints are given in Table 6.1. Due to its simplicity and efficiency this model has been described in the literature several times independently with differing names and is called image distortion model (IDM) here. When used with the appropriate pixel-level context description it produces very good classification results for object recognition tasks like handwritten digit recognition [Keysers & Gollan⁺ 04b] and radiograph classification [Keysers & Gollan⁺ 04a].

In first experiments, we did not include local pixel context for the matching in the IDM [Keysers & Dahmen⁺ 00b, Keysers 00]. In that case we could already improve results for the medical images of the IRMA task, but we were not able to improve the results for the handwritten digit recognition task USPS. The inclusion of local context then gave a noticeable improvement of the performance in both applications and we could generalize the improvements across different data sets [Keysers & Gollan⁺ 04a, Keysers & Gollan⁺ 04b, Gollan 03, Dreuw 05].

The IDM is a zero-order model, i.e. dependencies in the displacement grid are neglected. Consequentially, it results from using no relative constraints but only uses a global warp range as an absolute constraint (cp. Table 6.1). Since the dependencies are neglected, the minimization process for the IDM is computationally inexpensive. In comparison with the Euclidean distance it needs approximately a factor of $(2w + 1)^2$ more in computation time, where w denotes the warp range, i.e. the maximum allowed absolute displacement of one pixel.

If we briefly interpret the IDM from a probabilistic point of view, we can do so by using a zero-order model for the pixel alignment, denoted by the probability $p(x, y|i, j)$. We use a Gaussian density for the pixel values which results in the use of the Euclidean distance for the distance between the test image and the best matching of the prototype.

$$\begin{aligned}
p(A|B) &= \\
&= \prod_{i,j} p(a_{ij}|B) \\
&= \prod_{i,j} \sum_{x,y} p(x, y|i, j) p(a_{ij}|b_{xy}) \\
&= \prod_{i,j} \sum_{x,y} p(x, y|i, j) \mathcal{N}(a_{ij}|b_{xy}, \sigma^2) \\
&= \prod_{i,j} \frac{1}{\sqrt{2\pi}\sigma} \sum_{x,y} \exp \left[\log p(x, y|i, j) - \frac{1}{2\sigma^2} (a_{ij} - b_{xy})^2 \right] \\
&\cong \prod_{i,j} \frac{1}{\sqrt{2\pi}\sigma} \exp \left[\max_{x,y} \left\{ \log p(x, y|i, j) - \frac{1}{2\sigma^2} (a_{ij} - b_{xy})^2 \right\} \right] \\
&= \frac{1}{(\sqrt{2\pi}\sigma^2)^{I \cdot J}} \exp \left[- \sum_{i,j} \min_{x,y} \left\{ -\log p(x, y|i, j) + \frac{1}{2\sigma^2} (a_{ij} - b_{xy})^2 \right\} \right]
\end{aligned}$$

We used the maximum approximation of the sum of exponentials in the fifth step. The result is that we have a local cost function $-\log p(x, y|i, j)$ for the pixel displacement. If we assume a uniform probability within the region allowed by the absolute warp range constraint $p(x, y|i, j) = \frac{1}{2w+1}$ for x, y in the corresponding admissible area (see Table 6.1) and 0 otherwise, we arrive at the matching procedure without cost functions.

If a suitable pixel context representation is used with the IDM, a separate cost function is not necessary. In previous experiments, that used only the pixel value itself, a suitable cost function was necessary, though. In that case the cost function represents the cost for explaining a pixel a_{ij} in the input image with a pixel b_{xy} of the reference image and is introduced to compensate for the fact that in an unrestricted distortion model (i.e. with uniform cost in the admissible region) wanted as well as unwanted transformations can be modeled. With growing admissible region the possible transformations may violate the assumption that they respect the class-membership of the image. In fact, the distortion distance between almost any two images can be reduced to a value near zero by increasing the region, which leads to a significant decrease in classification performance. In the first series of experiments with the IRMA data, an appropriate choice of warp range (or admissible region size) led to a strong improvement of radiograph classification, even when the cost function was disregarded, but with the use of a cost function even better could be achieved. To determine the cost function, two methods can be proposed [Dahmen & Theiner⁺ 00]:

- Choose the cost function empirically, e.g. by using a weighted Euclidean distance between pixels. This way, small local transformations are preferred to (most probably unwanted) long-range pixel transformations.
- Learn the cost function by using training samples and a maximum likelihood approach. That is, apply meaningful transformations in training and use (smoothed) relative frequencies of possible transformations; the more often a transformation was observed in training, the lower its cost.

The IDM is a very natural approach and the idea can be found in various settings in the literature e.g. [Huttenlocher & Lilien⁺ 99, Smith & Bourgoïn⁺ 94]. Nevertheless it is an effective means to compensate for small local image variations and the intuitiveness of the model may be seen as an advantage. For example, [Smith & Bourgoïn⁺ 94] denote a similar model by ‘pixel distance metric’: “For mismatched pixels between the test and training images, it takes into account the distance to the nearest pixel of the same color” (and the authors give six references with descriptions of similar techniques) This can be compared to the cost function used in the IDM, but here only exact matches respectively binary images are considered.

For the tangent distance, we can nicely visualize a schematic view of the space of all images that have a distance of zero to a given prototype in the vector space of images. The visualization is not as straightforward for the image distortion model. If we consider a very low-dimensional example, a feature vector of length two, we can observe the following: for the (reference) feature vector (a_1, a_2) , there are three other (test) feature vectors that have a distance of zero under the basic IDM, i.e. (a_1, a_1) , (a_2, a_1) , and (a_2, a_2) . These four vectors form a rectangle with two corners on the main diagonal of the feature space. In a higher dimensional space and with more restrictions on the possible switchings due to a limited warp range, sets of possible images will form low-dimensional hyper-rectangles in the high-dimensional feature space. Therefore, we can view the IDM as being a ‘hyper-rectangle distance function’.

Recognition results using the IDM

The IDM led to very good results at a low computational complexity:

- On the USPS task, the IDM achieves a very good error rate of only 2.4% which is surprising for such a simple model. This result used a simple 1-NN classifier and the discussed representation of pixel context. For example, to achieve the same result, the kernel density classifier using tangent distance has to include two-sided tangent distance and nine-fold virtual training and test data to achieve the same result.
- On the MNIST task, due to the good results of the IDM on the other databases and the lower complexity, at first only the IDM was tested. It resulted in an error rate of 0.54% as compared to the rate of 0.63% reported in [Belongie & Malik⁺ 01] and 0.56% reported in [DeCoste & Schölkopf 02]. After publication of these results in [Keysers & Gollan⁺ 04b] we became aware of other results on the MNIST task that provided even better results than these, most notably the 0.42% error rate of [Simard & Steinkraus⁺ 03]. For a more detailed discussion of results please refer to Section 3.1.2 (pages 12ff.). We also tested the P2DHMDM on the MNIST data as

described above and obtained a slightly but not significantly improved error rate of 0.52%.

- On the UCI optical digits corpus, a scaling to 16×16 pixels using spline interpolation was performed. Here, the IDM performed as good as the more complex P2DHMDM and both achieved an error rate of 0.8%, which is the best error rate known.
- On the MCEDAR task, also a scaling to 16×16 pixels using spline interpolation was used. The IDM achieves an error rate of 3.5% using a 3-NN classifier, which is only slightly worse than the best published error rate of 3.3% obtained by the P2DHMDM.
- On the ETL6A corpus, only the IDM was tested, to further investigate its generalization capabilities. The IDM obtains an error rate of 0.5% here. This is equal to the best published error rate reported in [Uchida & Sakoe 03a], which is obtained using a model of variability based on eigen-deformations.
- On the IRMA dataset of medical radiographs, the IDM also produces very good results. As reported in [Keysers & Gollan⁺ 04a], the error rate that is obtained by using the IDM with thresholding in a 1-NN classifier on the medical data is 6.6% and thus already better than all previously reported error rates. Previous experiments already incorporated the IDM, but did not use the local context in the matching process [Keysers & Dahmen⁺ 03]. Using the proposed techniques for the inclusion of local context information, the performance using the image distortion model and thresholding could be significantly improved from 9.0% to 6.6% error rate. Other results on these data are summarized in Table 3.9 on page 23. Although the error rates of the P2DHMM (5.7%) and the P2DHMDM (5.3%) are still better, in large-scale applications the IDM is still preferable because of the much lower runtime. For example, the IDM was used in the 2004 ImageCLEF evaluation of content-based medical image retrieval and obtained the best result among those approaches using visual information only and not using relevance feedback [Deselaers & Keysers⁺ 04c]. In that application, a (possibly) somewhat better but much slower distance measure is not feasible. This statement will probably remain true, because even though computers continue to provide more and more computing power, also the databases of images that are used grow substantially in size. Figure 6.20 shows the result of an example query on the IRMA database using appearance-based distance measures as discussed in this chapter.

In the automatic annotation task of the 2005 ImageCLEF evaluation of content-based medical image retrieval using the IRMA data, the error rate of 12.6% obtained by the IDM was the best among 42 results submitted. The second best result uses the image distortion model along with the normalized cross covariance of gray values, and Tamura texture features [Lehmann & Güld⁺ 05]. In comparison, the baseline error rate obtained by a 1-nearest neighbor classifier using 32×32 thumbnail images is 36.8%. The average error rate of all submissions was 32.7% and the median was 22.3%.

6.7 Matching using the Hungarian algorithm

The deformation models that are discussed in this chapter all have in common that the task is to find a matching between pixels in an observed image and those in a reference image.

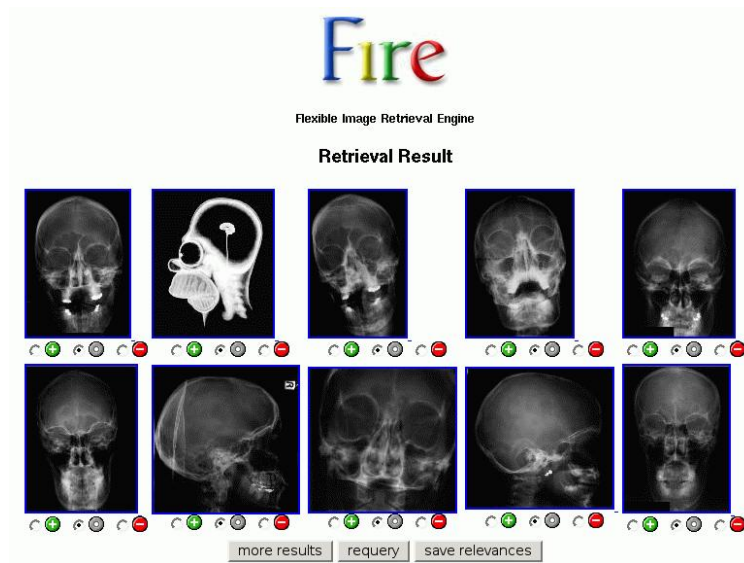


Figure 6.20: Image retrieval for medical images using the flexible image retrieval engine developed in [Deselaers 03]. The figure shows a screen-shot of a web-based interface. A retrieval on the IRMA database enriched with one cartoon image has been performed with the query image shown in the upper left.

The term ‘matching’ is a well-known expression in graph theory, where it refers to a selection of edges in a (bipartite) graph. This edge selection implies an assignment between vertices of the graph. We therefore may want to take a look at algorithms known from graph theory that may be useful to determine such a matching also for image pixels, if we choose an appropriate construction of graphs from two images that are to be compared. In this section we will explore this possible application and use the Hungarian algorithm to solve different assignment problems for image pixels. The Hungarian algorithm solves the ‘minimum weight bipartite matching’ problem. It has been used before in the context of image matching to assign image region descriptors known as ‘shape contexts’ of two images onto each other [Belongie & Malik⁺ 01, Belongie & Malik⁺ 02].

The use we make of the Hungarian algorithm amounts to the inclusion of an additional constraint in the IDM: we include the constraint that in the matching process each pixel of both compared images must be matched at least once. This constraint gives each pixel in both images the same importance. The optimal matching under this constraint can be determined using the Hungarian algorithm. We call the resulting model the Hungarian Distortion Model (HDM). The HDM respects larger parts of the reference image than using the IDM alone and thus leads to more homogeneous displacement fields in the matching. The algorithm as used here does not take into account dependencies between the displacements of neighboring pixels and is therefore also a zero-order model, as is the IDM, on which it is based. Note that it is not obvious if such dependencies could be included within a matching that is based on the Hungarian algorithm.

6.7.1 Construction of the bipartite graph

The construction of the bipartite graph in the case discussed here is straightforward: each pixel position of one of the two images to be compared is mapped to a node in the graph.

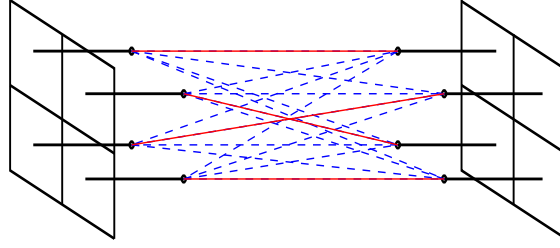


Figure 6.21: Schematic view of the construction of the bipartite graph for the Hungarian matching. Each image pixel is turned into a vertex, the two components originating from the two images. All vertices of the two components are connected by edges, where the edge weight corresponds to the distance between the pixel representations. The Hungarian algorithm determines the matching (solid lines) with minimum total edge weight.

Two nodes are connected by an edge if and only if they represent pixel positions that are not from the same image. This means that the two parts of the bipartite graph represent the two images. The weight of an edge is the distance between the respective pixel values, possibly enlarged by penalties for too large absolute distortions. Figure 6.21 shows a simple illustration for images with four pixels.

More formally, the constructed graph $G = (V, E)$ consists of the vertices $V = \{(i, j)^A\} \cup \{(x, y)^B\}$ with edges $E = \{((i, j)^A, (x, y)^B)\}$ and edge weights $w(((i, j)^A, (x, y)^B)) = \sum_{u=1}^U \|a_{ij}^u - b_{xy}^u\|^2$. In this case we include appropriately weighted position features in the U -dimensional local pixel context (e.g. $\frac{i-1}{I-1}, \frac{j-1}{J-1}, \dots$) that describe the relative pixel position in order to assign higher costs to mappings that deviate much from a linear matching. In that case the value of U is two larger than if the context alone is used, e.g. for the use of 3×3 contexts of the gradients, we have $U = 20$.

6.7.2 Outline of the Hungarian algorithm

The outline of the Hungarian algorithm given here follows [Knuth 94, pp. 74–89], which was also the basis for our implementation. The name ‘Hungarian’ algorithm is due to a constructive result published by two Hungarian mathematicians in 1931 that is used in the algorithm [Knuth 94, p. 78].

To explain the basic idea, we assume the weights of the edges are given by the entries of a matrix and we assume that both components of the graph have the same number of vertices, meaning that the weight matrix $A \in \mathbb{R}^{N \times N}$ is square. Then the goal of the algorithm is to find a permutation $\pi : \{1, \dots, N\} \mapsto \{1, \dots, N\}$ such that $\sum_{n=1}^N A_{n\pi(n)}$ is minimized.

Now, we can make the following three observations [Knuth 94, p. 77]:

- (a) Adding a constant to any row of the matrix does not change the solution, because exactly one term in the sum is changed by that amount independent of the permutation.
- (b) Adding a constant to any column of the matrix does not change the solution for the same reason.
- (c) If $A_{nn'} \geq 0$ for all $1 \leq n, n' \leq N$, i.e. the matrix is nonnegative, and $A_{n\pi(n)} = 0$ for all $1 \leq n \leq N$ then π is a solution.

Let two zeroes in A be called independent, if they appear in different rows and columns. The algorithm now uses the following ‘Hungarian’ theorem: The maximum number of mutually independent zeroes in A is equal to the minimum number of lines (rows or columns) that are needed to cover all zeroes in A . Assume an algorithm that finds such a minimum number of lines and a corresponding maximum set of mutually independent zeroes (as outlined below). Then the complete algorithm using the observations above can be formulated as follows.

1. subtract the minimum element of each row from each row
2. subtract the minimum element of each column from each column
3. find a maximum set of N' mutually independent zeroes and the corresponding minimum number of lines
4. if the solution has $N = N'$ mutually independent zeroes, output their indices and stop
5. otherwise cover all zeroes in A with N' lines and find the minimum uncovered value (which must be greater than zero); subtract it from all uncovered elements and add it to all doubly covered elements (this can be done by subtracting it from each row that is not in the set of lines and adding it to all columns that are in the set)
6. go to 3

To fully understand the algorithm, we need to know how step 3 works and why we can be sure that the algorithm always terminates. The detailed discussion of the first point is beyond the scope of this overview. We try to give a short idea and otherwise refer to [Knuth 94]: Start with an initial set of independent zeroes (empty or greedily constructed), and call these ‘special’. Choose a column, if there is a path of alternating rows and columns starting from a row that does not contain one of the special zeroes, where the elements of the path are linked by alternating special and non-special zeroes in A (note that this can be determined efficiently). Choose a row, if it contains a special zero that is not covered by a column. If there is an uncovered zero, we can increase the number of special zeroes by either making this zero special or by using an alternating path constructed before and exchanging special and non-special zeroes in this path. Repeat this until all zeroes are covered.

1. Choose an initial set of independent zeroes (e.g. greedily constructed) and call these ‘special’.
2. Cover rows containing one of the special zeroes and mark all other rows.
3. While there are marked rows, choose the next marked row: for each zero in the row that is not in a covered column, two cases are possible: a) the column already contains a special zero in another row ‘ ρ ’: cover the column and uncover and mark ρ . b) a new special zero is found and processed. When the row is processed completely, unmark it.

Termination of the algorithm is guaranteed, because in step 5 either the number of mutually independent zeroes or the number of covered columns is increased by the newly introduced zero. Since this can happen at most N times, the algorithm must terminate.

The total running time of this algorithm is $O(N^3)$, where the average case can be much lower if good initial assignments can be chosen. This complexity implies that the application of the HDM to large images is only possible at a high computational cost.

Note that there are other algorithms to solve the assignment problem, but most of these algorithms are developed for special cases of the structure of the graph (which is always a complete bipartite graph here).

6.7.3 Application of the Hungarian algorithm

The Hungarian algorithm gives us a tool to solve an assignment problem. In the case of image matching, this means that we can determine the best matching of pixels onto each other, where each pixel is matched exactly once. It is possible to use the Hungarian algorithm for this task directly, but in many cases it is more appropriate to match the pixels onto each other such that each pixel is matched at least once or such that each pixel of the test image is matched exactly once. This last case corresponds to the setting used in dynamic time warping and is used most frequently throughout the experiments. We then require that the reference image ‘explains’ all the pixels in the test image. These possibilities lead to three applications of the Hungarian algorithm for image matching, which are explained in the following.

Each pixel matched exactly once This case is solved in a straightforward manner. Construct the weight matrix as discussed in Section 6.7.1, then apply the Hungarian algorithm to solve for a minimum weight matching.

Each pixel matched at least once For this case, a reduction to the exact match case is done, where we follow an overview given by Richard Zens based on [Keijsper & Pendavingh 98]. This solves the ‘minimum weight edge cover’ problem.

1. construct the weight matrix as discussed in Section 6.7.1
2. for each vertex compute the minimum weight of all incident edges and remember one edge that has this minimum weight
3. for each edge subtract from its weight the minimum weight of both vertices as computed in the previous step
4. use the Hungarian algorithm to compute a minimum weight matching (for the Hungarian algorithm, subtract the overall minimum edge weight from all edge weights to make sure the weights are nonnegative)
5. delete all edges from the matching that have a weight larger than zero (their vertices can be covered better by using the minimum weight incident edges)
6. for each uncovered vertex, add an edge with minimum weight to the cover

This algorithm determines a minimum weight edge cover and therefore matches each pixel at least once.

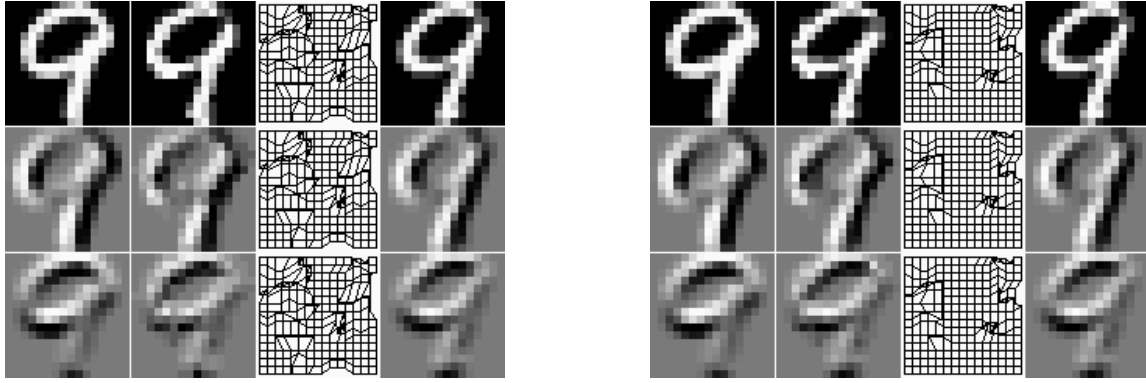


Figure 6.22: Examples of pixel displacements; left: image distortion model; right: Hungarian distortion model. Top to bottom: gray values, horizontal, and vertical gradient; left to right: test image, distorted reference image, displacement field, and original reference image. The matching is based on the gradient values alone, using 3×3 local sub images and an absolute warp range of 2 pixels.

Each pixel of the test image matched exactly once This task is solved by the image distortion model, we only need to choose the best matching pixel for each pixel in the test image. If we would like to infer such a matching from a matching as computed by the Hungarian algorithm, we let the previous algorithm be followed by the step:

7. for each pixel of the test image delete all edges in the cover except one with minimum weight

The resulting matching then does not have the overall minimum weight (which would be determined by the IDM) but due to the construction of the matching, it respects larger parts of the reference image. Therefore, the resulting matching is more homogeneous. In informal experiments this last choice showed the best performance and was used for the experiments presented in the following.

6.7.4 Complexity of the HDM

As mentioned above, the complexity of the Hungarian algorithm is cubic in the number of vertices. For the case of images this results in a problematic runtime in the order $O(I^3 J^3)$ (assuming images of the same size). For example, for the USPS task this results in the duration of 0.1 seconds per image comparison on a 1.8GHz machine with $U = 20$. We therefore cannot present any results using the HDM for the MNIST data set that contains larger and more images.

6.7.5 Experimental results for the HDM

The HDM was evaluated on the USPS database and achieved an error rate of 2.2% which is – though not being the best known result – state-of-the-art and an improvement over the 2.4% error rate achieved using the IDM alone. A complete overview of results for the USPS task is presented in Table 3.2 on page 10.

Figure 6.22 shows the comparison of two typical examples of pixel displacements resulting from the IDM and the HDM for the matching of two images containing the handwritten

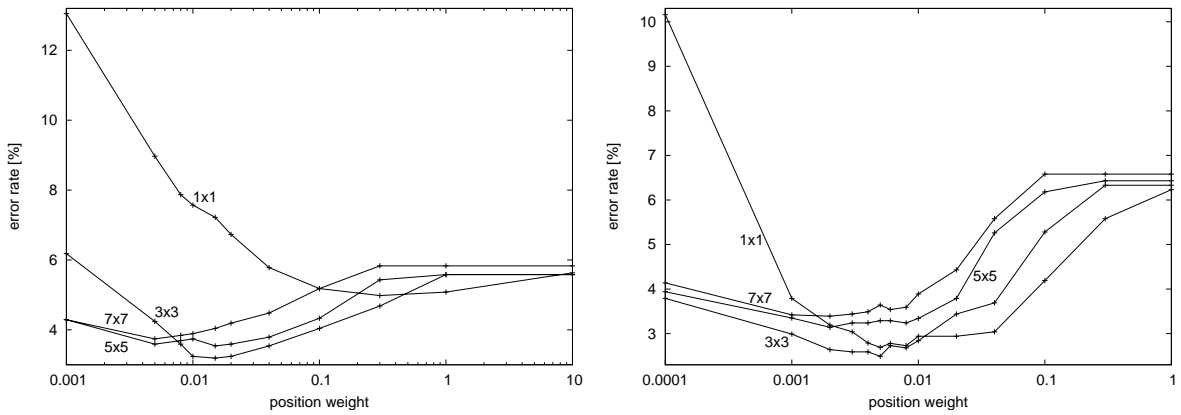


Figure 6.23: Error rates on USPS vs. position weight and sub image size using HDM with grayvalues (left) and with gradients (right) (preselection: 100 nearest neighbors, Euclidean distance).

digit ‘9’. It can be observed that the HDM leads to a more homogeneous displacement field due to the additional restriction imposed in the calculation of the mapping.

Figure 6.23 (left) shows the error rate of the HDM with respect to the weight of the position feature in the matching process. The pixel features used are the grayvalue contexts of sizes 1×1 , 3×3 , 5×5 , and 7×7 , respectively. Interestingly, already using only pixel grayvalues (1×1), the error rate can be somewhat improved from 5.6% to 5.0% with the appropriate position weight. Best results are obtained using sub images of size 3×3 leading to 3.2% error rate.

Figure 6.23 (right) shows the error rate of the HDM with respect to the weight of the position feature using the vertical and horizontal gradient as the image features with different local contexts. Interestingly, the 1×1 error rate is very competitive when using the image gradient as features and reaches an error rate of 2.7%. Again, best results are obtained using sub images of size 3×3 and position weights around 0.005 relative to the other features, with an error rate of 2.4%.

All previously described experiments used a preselection of the 100 nearest neighbors with the Euclidean distance to speed up the classification process. (One image comparison takes about 0.1s on a 1.8GHz processor for 3×3 gradient contexts.) Using the full reference set in the classifier finally reduces the error rate from 2.4% to 2.2% for this setting.

Note that the overall improvement using the HDM instead of the IDM corresponds to an improvement from 2.4% to 2.2%. This improvement is statistically not significant on a test corpus with a size of 2,007 images but is still remarkable in combination with the resulting more homogeneous displacement fields.

6.8 Gesture and sign language recognition using models of variability

The models of image variability discussed in this work were also applied to the task of appearance-based gesture and sign-language recognition with very positive results [Dreuw 05, Dreuw & Keyers⁺ 05, Zahedi & Keyers⁺ 05a, Zahedi & Keyers⁺ 05b,

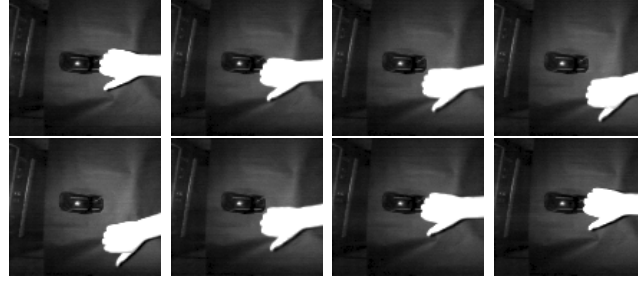


Figure 6.24: Example images from the LTI gesture recognition database.

Table 6.6: Results for gesture recognition on the LTI database using image comparison methods based on models of variability

method	reference	ER[%]
Euclidean distance	[Dreuw & Keysers ⁺ 05]	5.7
segmentation, shape features	[Pelkmann 99]	4.3
tangent distance	[Dreuw & Keysers ⁺ 05]	1.4
IDM	[Dreuw & Keysers ⁺ 05]	1.4

Zahedi & Keysers⁺ 05c], which underlines their broad applicability. We briefly summarize the main results for these tasks of image sequence classification here.

The decision rule to classify image sequences is based on the methods used in automatic speech recognition [Ney & Ortmanns 00]. We employ standard 0-1-2 hidden Markov models and Gaussian mixture densities with variance pooling for the emission densities. Using appearance-based features, i.e. down-scaled images and their spatial and temporal derivatives enables us to use invariant distance measures based on the models of variability as discussed above. The decision rule for an image sequence x_1^T is based on Bayes' decision rule. The class conditional distribution is factorized assuming the Markov property and hidden states s_1^T . We approximate the sum over all state sequence probabilities by the maximum probability. The state-specific emission densities in turn are modeled using invariant distance measures within a Gaussian density:

$$\begin{aligned}
 r(x_1^T) &= \arg \max_k \{p(k) \cdot p(x_1^T|k)\} \\
 p(x_1^T|k) &\cong \max_{s_1^T} \left\{ \prod_{t=1}^T p(s_t|s_{t-1}, k) \cdot p(x_t|s_t, k) \right\} \\
 p(x|s, k) &= \frac{1}{Z} \exp(-d(x, \mu_{ks}))
 \end{aligned}$$

6.8.1 Gesture recognition

For the task of gesture recognition we used sequences of infrared images of size 106×96 showing a top view of the user's hand. The data was recorded inside a car and originates from the Lehrstuhl für Technische Informatik (LTI) of the RWTH Aachen University [Pelkmann 99, Akyol & Canzler⁺ 00]. The database contains 14 different dynamic gestures with a total of 359 image sequences, which were split into a training and test set. The



Figure 6.25: Example images from the BOSTON50 database for sign language word recognition.

Table 6.7: Results for sign language word recognition on the BOSTON50 database using image comparison methods based on models of variability.

method	reference	ER[%]
Euclidean distance	[Zahedi & Keysers ⁺ 05a]	23.6
tangent distance	[Zahedi & Keysers ⁺ 05a]	22.2
IDM	[Zahedi & Keysers ⁺ 05a]	21.9
IDM + tangent distance for local contexts	[Zahedi & Keysers ⁺ 05a]	17.2

only reference result available is due to [Pelkmann 99] and achieved an error rate of 4.3% using segmentation of the hand and various shape-based features within a classifier based on hidden Markov models. Figure 6.24 shows example images from the database.

A summary of the results obtained in the experiments is given in Table 6.6. (More details can be found in [Dreuw 05].) We can observe that both the tangent distance and the IDM, when used within the Gaussian emission densities, improve the recognition result considerably.

6.8.2 Sign language recognition

In sign language recognition a considerable number of errors are due to the variability of the input signal. Each signer may utter a word differently, depending on his individual signing style or the predecessor and successor of the uttered word. Therefore, a large visual variability of utterances for each word exists. To model the variability of utterances, the tangent distance and the image distortion model can be used to account for global and local variations, respectively [Zahedi & Keysers⁺ 05a].

For the task of sign language recognition we used a database of 50 different words of American sign language, the BOSTON50 database. The BOSTON50 database was created from the database of ASL sentences published by the National Center for Sign Language and Gesture Resources at Boston University². The gray images of the sequences are of size 195×165 pixels and two simultaneous camera views are available, one frontal and one lateral.

²<http://www.bu.edu/asllrp/ncslgr.html>

Table 6.8: Summary of results for handwritten digit recognition and the IRMA-1,617 data.

database	matching model					
	none	2DW	P2DHMM	P2DHMDM	IDM	HDM
USPS	5.6	2.7	2.5	1.9	2.4	2.2
MNIST	3.5			0.52	0.54	
UCI	2.0		1.1	0.8	0.8	
MCEDAR	5.7			3.3	3.5	
IRMA	15.8		5.7	5.3	6.6	

Algorithm **IDM-distance**; input: test image A , reference image B ; output: distance

use the horizontal and vertical Sobel-filter on the images A, B

yielding the filtered images A^v, A^h, B^v, B^h

for each pixel position (i, j) of A

for each pixel position $(x, y), |x - [i \frac{X}{T}]| \leq 2, |y - [j \frac{Y}{J}]| \leq 2$ of B

determine the local Euclidean distance

$$\sum_{m=-1}^1 \sum_{n=-1}^1 [A_{i+n, j+m}^v - B_{x+n, y+m}^v]^2 + [A_{i+n, j+m}^h - B_{x+n, y+m}^h]^2$$

add the minimum local distance to the distance

Figure 6.26: Summary of the IDM distance computation to underline its algorithmic simplicity. This algorithm used in a simple 3-NN classifier to compute the distance measure can lead to excellent results for different tasks. Additionally it can be easily implemented in a few lines of code, thus making it an ideal baseline distance measure for use in appearance-based image recognition.

These two views are merged into one feature vector after down-sampling to 13×11 pixels. According to the experiments reported in [Zahedi & Keysers⁺ 05b], the features of the front and side cameras are weighted with 0.38 and 0.62, respectively. There are 483 utterances of the 50 words available, each word occurs between 2 and 37 times. A leaving-one-out approach is used for this database, i.e. each utterance is classified using the remaining set as training utterances. The words were signed by three different signers, one man and two women. The signers were dressed differently. Here, very good results were achieved by using one HMM for each training utterance. Figure 6.25 shows example images from the database.

A summary of the results obtained in the experiments is given in Table 6.7. As for the gesture recognition data, we can observe that both the tangent distance and the IDM, when used within the Gaussian emission densities, improve the recognition result considerably. Unfortunately, so far no results of other groups for comparison on this data set are available.

6.9 Conclusion

We discussed several nonlinear models of image variability that can be used as distance measures in appearance-based classifiers. From experiments on several real-world image recognition tasks as summarized in Table 6.8 we can conclude the following: the simplest

model — the image distortion model — when used with pixel level features that describe the local image context represents the best compromise between computational complexity and recognition accuracy, while the more complex P2DHMDM even leads to slightly better results on the average and outperforms the conventional P2DHMM. The most complex model, the 2DW, performed worse or could not be evaluated due to the high computational demand.

We could show that the IDM with local gradient contexts leads to excellent results for the recognition of handwritten digits and of medical images and also improves the results for the appearance-based recognition of image sequences. On all datasets considered, state-of-the-art results were obtained using the efficient IDM, which can be described in a few statements and implemented in a few lines of code. A concrete summary of the IDM approach for image matching as used for digit recognition (0.54% error rate on MNIST) is as simple as summarized in Figure 6.26.

7 Recognition based on local patches

Modeling an airplane is much harder than telling airplanes from bicycles.

– C. Bishop, 2004

In this chapter, we discuss the approach of modeling the variability of images by regarding each image as a collection of smaller sub-images, usually called patches. This approach can be interpreted as a further step along the direction taken in the context of deformable models described in Chapter 6: Now the matching of local contexts (patches) is not restricted to one image any more. Instead, each patch of a test image can be matched to any patch from the set of patches that were extracted from the training images of the hypothesized class.

A patch-based approach to recognition has several advantages in the presence of variability. For instance, the classification is inherently invariant to translations of the object in the image if the position of a patch is disregarded in the classification process, which is often the case. It is also evident that this approach can handle occlusions well. If parts of an object are occluded in an image, the remaining visible parts may still be used to recognize the object correctly or to learn about the appearance of an object from this instance. Further advantages include that changes in the geometrical relation between image parts can be modeled to be flexible or can even be ignored, and the algorithm can focus on those image parts that are most important to recognize the object.

Another advantage of patch-based methods as considered here is that patches containing different local variations can be simultaneously chosen from various training images from the same class to explain a test image patch set. Thus, it is possible to approximate image transformations that were not present in the training data.

7.1 Introduction and related work

We first introduce our baseline method for patch-based classification and some related work.

7.1.1 Introduction

The basic impulse for our interest in the use of local patches for image classification came from the publications of R. Paredes and colleagues, e.g. [Paredes & Perez-Cortes⁺ 01]. In the course of experiments carried out in his group at the ITI in Valencia and at RWTH-i6, we observed that the method of ‘local features and direct voting’ lead to very good results on several classification tasks:

- Olivetti Research Laboratory - face recognition [Paredes & Perez-Cortes⁺ 01]
- US Postal Service - digit recognition [Keysers & Paredes⁺ 02]
- Image Retrieval in Medical Applications - medical images [Paredes & Keysers⁺ 02]

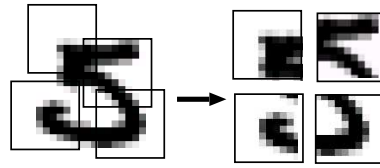


Figure 7.1: Example of patches extracted from an image of a handwritten digit.

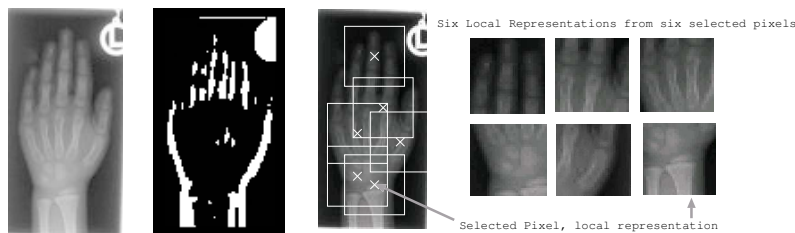


Figure 7.2: Example of patches extracted from an image of a medical radiograph. The second image shows all positions with a local variance above the selected threshold. From all the marked pixels, a local patch will be extracted. Six of these patches are shown on the right.

- University of Surrey competition - face verification [Messer & Kittler⁺ 03]

Here we try to investigate the method of [Paredes & Perez-Cortes⁺ 01] by systematic variations and extensions. Especially the simple but very effective probability model presented by Paredes and colleagues seems to leave some space for further investigation, especially the use of discriminative training as described in Section 5.3. The main points of interest are:

- Examine the effect of extracting patches at multiple scales.
- Review alternate feature reduction techniques. Here especially the linear discriminant analysis and the discrete cosine transform are of interest.
- Inspect invariant distance measures, especially the tangent distance.
- Experiment with various alternatives for the direct voting approach.
- Examine the use of discriminative training.

These single inspections are driven by the goal of better understanding why the patch approach leads to such good results despite its apparent simplicity and identifying and documenting the important aspects of the method. We show that improvements are possible for each of the investigated enhancements. This shows that the important aspect of the framework is the decomposition of the training images into sets of patches for each class.

In the discussed approach, each image is represented by several (possibly overlapping) square windows that correspond to a set of local appearances. Figure 7.1 shows an example

from a character recognition task and Figure 7.2 shows an example of possible local representations for the medical IRMA task. Note that the Figures use far fewer patches than the classifiers for purposes of illustration.

When applying the paradigm of classification by image parts, we must take decisions about the following two points:

- At which points in the image do we extract image patches that should capture the object parts?
- Given the image patches, how do we decide which class of object is present in the image?

We look at different answers to both questions. For the question regarding patch positions we can use image patches that are extracted at points of interest and also at regularly spaced intervals. We can use an available interest point detector [Loupas & Sebe⁺ 00] or a threshold to the local variance [Paredes & Perez-Cortes⁺ 01] (other authors use the local entropy [Fergus & Perona⁺ 03]) or other interest point detectors. In general, a selection of patches with highly discriminative content would be preferable and a variety of methods exist that are expected to detect such patches [Deriche & Giraudon 93].

For the decision rule, we start with extensions of the decision rule based on direct voting [Paredes & Perez-Cortes⁺ 01]. We also compare a discriminative log-linear model to other models that operate on the same patches, including naive Bayes, maximum likelihood of the class conditional probability, and a nearest neighbor model.

We discuss this method for automatically learning *discriminative* image patches for the recognition of given object classes [Deselaers & Keysers⁺ 05a]. The approach applies discriminative training of log-linear models to image patch frequencies that are represented by histograms. The use of histograms works by first clustering all training patches and then using only the information about the cluster each patch is closest to. Thus, this amounts to performing a class-independent vector quantization, which allows us to abstract from the patch itself to a cluster of similar patches in the discriminative training. We show that the method works well on three tasks and performs significantly better than other methods using the same patches. For example, the method automatically learns that patches containing an eye are most important for distinguishing face from background images. The recognition performance is very competitive with error rates presented in other publications.

For the basic classification method [Paredes & Perez-Cortes⁺ 01], we proceed as follows: For each test image, the same criterion to determine the positions of the patches is applied and the patches are extracted. The number of patches extracted from each image may vary. Each patch is associated with the same class label as the image it was obtained from. All these feature vectors are then joined to form a new training set.

The dimensionality of the patch vectors is then reduced using a principal component analysis on the set of all local patches extracted from the training set, by keeping only the first components, usually 40.

In testing, the same criterion for patch extraction is used and the PCA transformation is applied. For each patch of the test image, the patch from the training images that is most similar with respect to the Euclidean distance is searched among the training patches. The nearest neighbors of the test patches are grouped according to their class labels and the class which contains most of nearest neighbors is chosen as the classification result. This decision

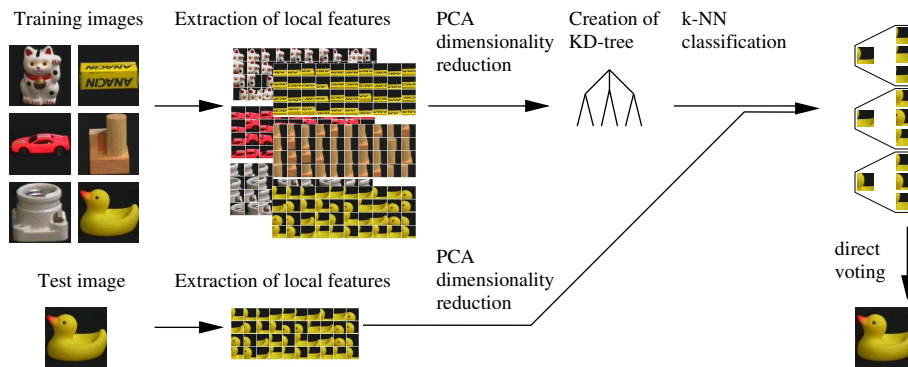


Figure 7.3: Schematic view of the basic classification approach [Paredes & Perez-Cortes⁺ 01] with local patches.

rule is called ‘direct voting’ because every patch gives an equally weighted vote to the class it belongs to. A probabilistic interpretation of this approach is discussed in Section 7.3.3.

Representing objects by several patches involves a computational problem if the number of patches to represent one object is very large. The k -NN algorithm needs to compare every patch of a test object with every patch of every training object. This high computational cost is considerably reduced by using a fast approximate k -nearest neighbor search technique [Arya & Mount⁺ 98].

The complete method is illustrated in Figure 7.3.

7.1.2 Related work

Related work includes [Mohan & Papageorgiou⁺ 01] who use predetermined parts of human bodies to detect humans in cluttered scenes. [Dorko & Schmid 03] use image patches to classify cars, but the extracted patches from the training set are hand-labeled whether they are part of a car or not. [Leibe & Schiele 04] use scale-invariant interest points and manually segmented training data for classification. In contrast to these approaches, we only need weak supervision in training, i.e. only information about the presence of an object in the image. [Fergus & Perona⁺ 03] and [Weber & Welling⁺ 00] statistically model relative position, scale, occurrence, and appearance of object parts with a mixture density. [Schmid & Mohr 97] use circular regions at corners characterized by expressions that depend on the image derivatives and are invariant with respect to image rotation. Matches are found by comparing the local regions and counting the number of matching regions. The spatial configuration is also regarded. [Gao & Vasconcelos 05] present a method to determine salient local features based on the concept of discriminative power in a classification framework. [Lazebnik & Schmid⁺ 05] recently presented an approach that uses the maximum entropy framework in combination with image patches for recognition and also takes into account dependencies between the occurrence of different patches. However, one of the results presented is that the use of the relation between parts did not improve the recognition performance.

Many of the proposed patch approaches compare the local patches on a per image basis and model the global relation between the patch positions. In the approaches discussed here, the relation between patches is ignored and the comparison is not done on a per image basis.

Instead, every patch of the test image is compared to all patches of all training images; therefore only the local image context is relevant to the matching and the final decision.

Recently, patch-based classification is a topic of research activities at several groups world-wide. To illustrate this fact, we briefly mention some related approaches that were presented at the 2005 International Conference on Computer Vision and Pattern Recognition: [Marée & Geurts⁺ 05] extract patches of size 16×16 pixels from randomly chosen positions and use boosted trees for classification. They present good results on several datasets. [Eppshtein & Ullman 05] extract patches and identify semantically equivalent object fragments by taking into account spatial contexts (neighbor patches) and relative positions. [Fergus & Perona⁺ 05] extend their previous approach towards using computationally less expensive spatial relationships, e.g. by using a star instead of a completely connected network. [Holub & Perona 05] compare discriminative and generative models and come to the conclusion that generative models work well for distinct classes (e.g. cows vs. motorbikes), but discriminative models outperform generative ones in the case of very similar classes (e.g. different persons).

7.2 Feature extraction

In a conventional classifier, each object for training and test is represented by a feature vector and a discrimination rule is applied to classify a test vector. In the image classification problem, this feature vector is often obtained from the whole image, using the appearance-based approach (each pixel corresponds to one feature) or some type of feature extraction. However, in the patch-based approach, each image is represented by several (possibly overlapping) square windows that correspond to a set of ‘local appearances’.

7.2.1 Patch extraction

To obtain the patch vectors from an image, a selection of windows with relevant and discriminative content is needed. In the experiments, we use two methods for the selection of windows:

- In the methods based on [Paredes & Perez-Cortes⁺ 01] the local variance of the gray values in a small window around each pixel is used as measure of information. Those pixels having local variance above a certain global threshold are selected and the surrounding windows are used as a representation of the whole image. This approach has been shown to improve recognition results compared to taking all local patches [Kölsch 03]. To reduce the number of local representations (also called patches) of an object a subsampling procedure may be applied to the selected pixels.
- In the extended methods investigating the use of discriminative training, we use about 500 square image patches as patches extracted around interest points obtained using the method proposed by Louprias and colleagues [Louprias & Sebe⁺ 00]. Additionally, we use 300 patches from a uniform grid of 15×20 cells that is projected onto the image. In contrast to the interest points from the detector, these points can also fall onto very homogeneous areas of the image. This property is important for capturing homogeneity in objects in addition to points that are detected by interest point detectors, which are usually of high variance. In informal experiments, this combination of points of

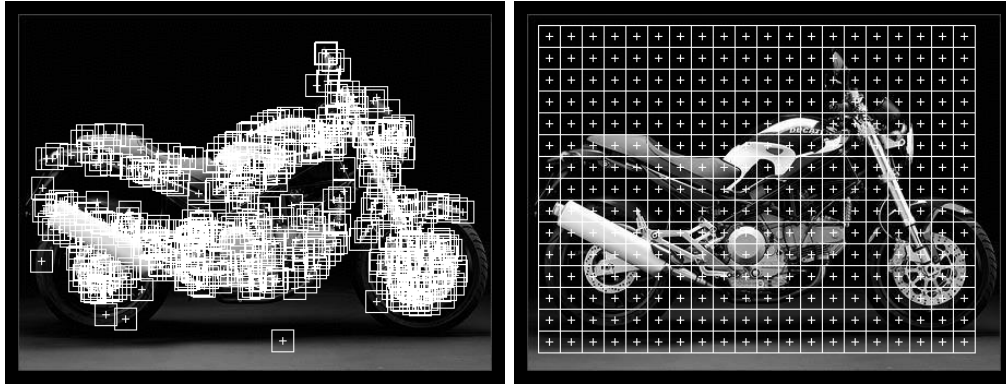


Figure 7.4: Patch extraction: salient points and uniform grid.

interest and regular grid performed better than both methods alone. Figure 7.4 shows the points of interest detected in a typical image. The patches are allowed to extend beyond the image border, in which case the part of the patch falling outside the image is padded with zeroes.

There is a large variety of interest point detectors (or salient point detectors) available that could be used instead of the choices described here. For example, instead of the local variance, we could also use the local entropy. Furthermore, there exist interest point detectors that directly return a scale parameter of the region, which could then be used for scale invariance. However, the investigation of various interest point detectors was not the goal of the experiments performed here.

Note that [Kölsch 03] describes a method for the estimation of the discriminative power of local patches by classifying the patches into two groups using Gaussian mixture densities based on the information how the patches contribute to a classification of the training images. In the experiments performed, although no large gain in classification accuracy could be obtained, it was possible to reduce the amount of patches extracted considerably.

7.2.2 Multi-scale patch extraction

In the baseline method [Paredes & Perez-Cortes⁺ 01], the extracted patches all have the same size. In many settings it is possible to experimentally evaluate which image patch size performs best on the given task. We may want to choose the scale at which a patch contains sufficient structure for recognition and still provides invariance towards global non-linear transformations. For face recognition it turns out to best select patches about the size of an eye, whereas for the IRMA radiography classification task the best results were obtained with patches that were about two thirds the size of the original image.

However, this approach can lead to problems if the objects to be recognized are of different scales, because by restricting the comparison to sub images of only one size we implicitly assume that the objects are represented at the same scale in all images to be classified.

It is also probable that there is more than one relevant patch size for one task. To classify the radiography of a chest for example, a patch of the shoulder might be a good indicator but the overall shape of the chest probably will lead to a good vote as well.

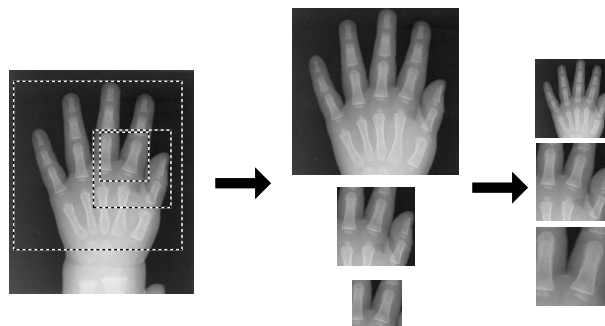


Figure 7.5: Extraction of multi-scale patches.

Table 7.1: Results for multi-scale patch extraction on the IRMA data.

	direct voting	kernel densities
fixed size	10.3	9.7
multiple scales	10.0	9.4

Motivated by these considerations and also by the results presented in [Fergus & Perona⁺ 03], we relax the constraint of using patches of a fixed size by extracting patches at different scales. To do so, a minimal patch size, a step size, and a maximal patch size are chosen [Kölsch & Keyers⁺ 04]. Then the extraction is performed at all pixel positions for windows of the chosen sizes and the patches with low local variance are discarded. In the next step, the patches are scaled to a fixed size. This extraction is done for the training and the test images, resulting in a potentially larger number of patches. However, this number of sub images can be adjusted by increasing the variance threshold. The procedure of multi-scale patch extraction is visualized in Figure 7.5

For example, in the experiments presented in [Deselaers & Keyers⁺ 05b] we extract patches of the following sizes: at each extraction point, we extract square patches of 7, 11, 21, and 31 pixels width. To be able to use these patches in the proposed training and classification framework, all extracted patches are scaled to a common size of 15×15 pixels.

Table 7.1 shows the improvements gained using multi-scale patches on the IRMA data for the baseline method and for kernel densities, which are discussed below [Kölsch & Keyers⁺ 04]. The results improve from 10.3% to 10.0% error with direct voting and from 9.7% to 9.4% for the kernel densities approach. This shows that the multi-scale extraction leads to improvements independent of the probability model.

7.2.3 Dimensionality reduction

To reduce the computational complexity of the subsequent classification steps, usually a dimensionality reduction of the extracted image patches is performed.

PCA

After the patches are extracted, a PCA dimensionality reduction is applied to reduce the large dimensionality of the data, keeping 40 coefficients (cp. Section 4.5.1).

The use of the PCA allows us to further process the patches in the following way: it can be observed that different images are often taken under different lighting conditions and thus
















	1st PCA comp.	w/ 1st PCA component		w/o 1st PCA component	
		bright	dark	bright	dark
airplanes					
faces					
motorbikes					

Figure 7.6: For each of the three Caltech tasks: first PCA component, a bright image patch and a dark image patch reconstructed from the PCA vectors using all 40 PCA components, and the same patches reconstructed from the PCA vectors after discarding the first PCA component.

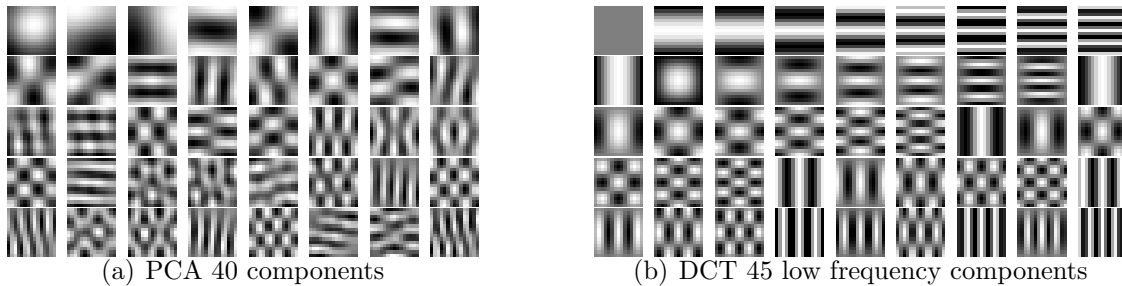


Figure 7.7: (a) The 40 first principal components, computed on the ORL face recognition corpus and (b) the 45 lowest frequencies using the DCT.

the brightness of otherwise very similar patches can vary significantly. Evidently, the brightness of a patch should usually not change its class membership, but the Euclidean distance between two patches that are identical except for brightness can be very high. A practical approach for brightness normalization in this context can be found in the PCA transformation: the first PCA vector for a collection of image patches usually captures the change in brightness and thus contributes most to the overall brightness of the image patches. Thus we may discard the first component of the PCA transformed vectors in order to reduce the effect of the global brightness of the image patches on the feature vector [Deselaers & Keysers⁺ 05b]. This approach can be found in various other settings, e.g. [Martinez & Kak 01]. Figure 7.6 illustrates the effect: for each of the three Caltech tasks (airplanes, faces, motorbikes) it shows the first component of the PCA matrix (clearly capturing global patch brightness) and an example of a bright and a dark patch reconstructed from the PCA-transformed and dimensionality-reduced representations, one with and one without the first PCA component. It can be observed that the differences in brightness are reduced for the patches that have been reconstructed without the first PCA component.

DCT

If we visualize the vectors belonging to the PCA transformation for the patches, we can observe an interesting similarity to the vectors that correspond to the discrete cosine transformation (DCT). Figure 7.7 shows a comparison between these vectors. This observation can be viewed in the context of the DCT as being used in the JPEG image compression scheme. Also in JPEG, the high frequency components of the image blocks are quantized more, because they contain information that is less relevant for the visual interpretation. It is also known that the DCT leads to a whitening of the covariance matrix in the transformed space if the covariance structure in the original space has a band structure, which is approximately true for image patches, in which the covariance of pixel brightnesses mostly depends on their relative position.

An alternative to the use of the PCA is therefore the use of the DCT. The advantage of the DCT is that the transformation matrix is known beforehand and does not depend on the training data.

To reduce the dimensionality using the DCT, the low frequency components are extracted. There are several ways to do so as discussed in [Kölsch 03]. The best method among those tried in the experiments was to keep those coefficients corresponding to small sums of the two indices for the two-dimensional transformation, where the pair $(0, 0)$ corresponds to the direct current component (image average).

Doing so, we can obtain an error rate of 10.9% on the IRMA data set compared to 10.3% when using the PCA. We can conclude that the computational complexity of the method in training can be reduced by using the discrete cosine transform with a small loss in recognition performance only.

However, we continue to use the PCA in the following experiments because our main goal is classification accuracy. In a setting where training time is important, for example to be able to quickly integrate new object classes into a system, it may be preferable to use the DCT instead.

LDA

Another question arising in the context of dimensionality reduction of the patches is whether we can use the class information to obtain more discriminative representations. To do so, the LDA as described in Section 4.5.2 can be used. In experiments on the IRMA corpus, to keep it comparable with the results corresponding to the use of the PCA, every class was subdivided into seven virtual classes [Kölsch 03]. This results in 42 pseudo-classes and thus a 41-dimensional feature space, similar to the 40 dimensions typically used with the PCA.

The resulting error rate on the IRMA corpus is 13.3% using the LDA versus 10.3% using the PCA. This demonstrates that the LDA is not well-suited for the local patch approach, which is a similar result to the one reported in [Fergus & Perona⁺ 03].

7.2.4 Histogramization

In the approach using discriminative training [Deselaers & Keysers⁺ 05a, Deselaers & Keysers⁺ 05b] we use an additional preprocessing step. In order to estimate discriminative weights that are pooled for visually similar patches the data are clustered with a Linde-Buzo-Gray algorithm using the Euclidean distance. Note that it is also possible to directly estimate the cluster densities using e.g. Gaussian mixture densities.

First experiments performed at the time of writing this document show promising results, although not for all regarded tasks [A. Hegerath, personal communication, July 2005].

After the clustering, we discard all information for each patch except its corresponding closest cluster center identifier. For the test data, this identifier is determined by evaluating the Euclidean distance to all cluster centers for each patch. Thus, the clustering assigns a cluster to each image patch and allows us to create histograms of cluster frequencies by counting how many of the extracted patches belong to each of the clusters. The histogram representation with as many bins as there are cluster centers is then determined by counting and re-normalizing to a sum of one.

In [Deselaers & Keysers⁺ 05b] we evaluate an extension of this approach, in which the histogram bin assignment is not crisp. Instead, the unit weight of each local patch is distributed among the bins by using a weight proportional to the negative exponential of the Euclidean distance to the cluster center.

7.3 Different models for patch-based classification

In this section we describe different models that were applied for patch-based image classification. These models are of two kinds: 1. models that use the patches directly and 2. models that use a histogram representation of the patches.

Having obtained the representation by (histograms of) image patches, we need to define a decision rule for the classification of images. In the following we briefly present different methods that use these representations. Note that most of the decision rules as they are presented here are simplified by the fact that in the experiments we assume a uniform prior distribution $p(k) = 1/K$.

7.3.1 Direct voting

Given a test image x , we obtain L_x patch vectors, denoted by $\{x_1, \dots, x_{L_x}\}$. Then, to solve the problem of classification of a test object represented by patches, the sum rule is used to obtain the posterior probability of the object from the posterior probabilities of its local representations [Paredes & Perez-Cortes⁺ 01]:

$$r(x) = \operatorname{argmax}_k P(k|x) \approx \operatorname{argmax}_k \sum_{l=1}^{L_x} P(k|x_l)$$

To model the posterior probability of each patch, a κ -nearest-neighbor approach is used:

$$P(k|x) \approx \frac{v_k(x)}{\kappa},$$

where $v_k(x)$ denotes the number of votes from class k found for the patch x among the κ nearest neighbors of the new training set. We adopt the sum rule as an approximation for the object posterior probabilities and the κ -NN estimate is used to approximate each patch posterior probability, yielding

$$r(x) = \operatorname{argmax}_k \sum_{l=1}^{L_x} \frac{v_k(x_l)}{\kappa} = \operatorname{argmax}_k \sum_{l=1}^{L_x} v_k(x_l). \quad (7.1)$$

Global patch search and direct voting was proposed in [Paredes & Perez-Cortes⁺ 01] and uses the PCA-transformed image patches directly without computing a histogram representation. A KD-tree is created from the training image patches to admit efficient nearest neighbor searches. Using this KD-tree, each test image patch is assigned the class of its nearest neighbor using approximate search. The search is global, because all patches originating from one class are treated equally independent of the image they were extracted from. The individual classifications of all patches are then combined by direct voting. The classification output is the class that most of the image patches have been assigned to. This method is known to obtain very competitive results in various tasks like face recognition, radiograph recognition, and character recognition [Paredes & Keysers⁺ 02] and therefore serves as a very good baseline. [Kölsch & Keysers⁺ 04] give a formalized description of the classification process and describe improvements that can be obtained by e.g. multi-scale patch extraction, a modified voting scheme, or invariant distance measures in the nearest neighbor search. In this work, we use the basic classification method as described above for comparison.

Approximative nearest patch search

Representing objects by several patches involves a computational problem if the number of patches is very large. The k -nearest neighbor algorithm needs to compare every patch of a test object with every patch of every training object. (Note that there are simple methods to speed up this search, for example to stop distance calculations if it becomes clear that the regarded sample cannot be one of the nearest neighbors.) Although the number of local representations can be reduced using subsampling (e.g. only every second patch is used) this procedure may lead to a degradation in the classification accuracy.

If we are dealing with a search among, for example, a million vectors with a dimensionality of 40, techniques for speeding up this search may be considered. One possible idea for improvement could be to use fast search structures. However, at feature dimensionalities greater than eight most structures do not perform significantly better than the brute force approach [Arya & Mount⁺ 98]. In this situation, approximative search strategies can lead to a significantly increased speed while limiting the maximum error by a given boundary.

The high computational cost is considerably reduced by using a fast approximate k -nearest neighbor search technique. This technique uses a KD-tree structure to store the set of local patches from the training images. In a KD-tree, the search of the nearest neighbor of a test point is performed starting from the root, which represents the whole feature space, and choosing at each node the sub-tree that represents the region of the space containing the test point. When a leaf is reached, an exhaustive search of the prototypes residing in the associated region is performed. Unfortunately, the process is not complete at this point. It is possible that among the regions defined by the initial partition, the one containing the test point is not the one containing the nearest prototype. It is possible to determine if this can happen in a given configuration, in which case the algorithm backtracks as many times as necessary until all the regions that can hold a prototype closer to the test point than the nearest one in the original region are checked.

If a guaranteed exact solution is not needed, the backtracking process can be aborted as soon as a certain criterion is met by the current best solution. In [Arya & Mount⁺ 98], the concept of $(1 + \epsilon)$ -approximate nearest neighbor query is introduced. A point p is a $(1 + \epsilon)$ -approximate nearest neighbor of q if the distance from p to q is less than $(1 + \epsilon)$ times the distance from p to its nearest neighbor. This concept is used to obtain an efficient

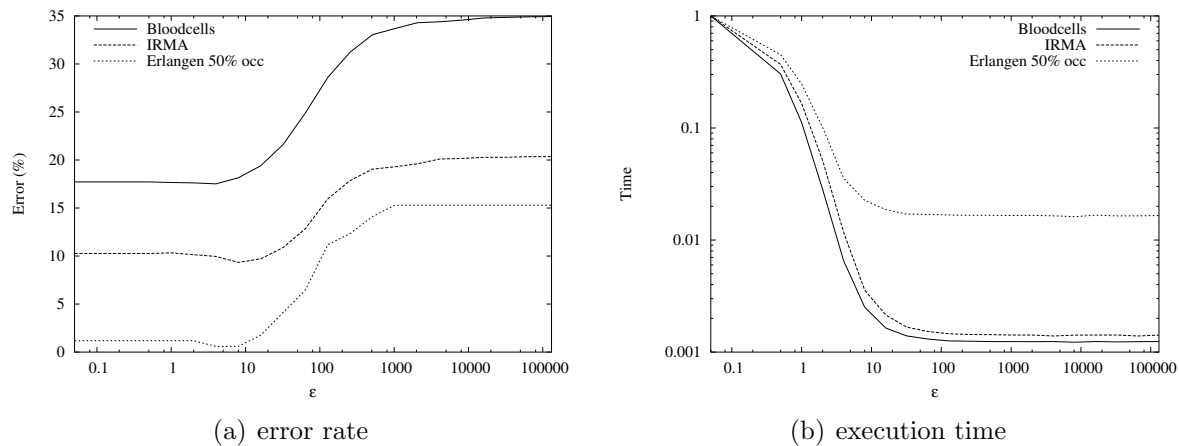


Figure 7.8: Error rates and relative execution times for the Bloodcells, IRMA, and Erlangen corpus with respect to the approximative search parameter ε .

approximate search that can easily cope with very large sets of reference vectors. The experimental results showed that the real error is usually significantly smaller than the theoretical bound in practical situations.

This strategy fits the concept of local patches with direct voting well. As one single result has little importance, it is the majority of votes that leads to the good overall recognition results. Thus, we do not expect the overall result to change significantly by using this technique.

In experiments, this expectation was not only confirmed, but the approximative search even turned out to reduce the recognition error on the data sets IRMA, Erlangen, and Bloodcells as shown in Figure 7.8(a). At the same time, execution time decreases rapidly as the value of ε increases. This is depicted in Figure 7.8(b).

An explanation of the good performance in the presence of approximation may be the following: we expect the occurrence of training patches from class k in the neighborhood of a test patch x to be proportional to $p(k|x)$, as the number of patches grows and the neighborhood shrinks. Now, if we choose not the nearest neighbor, but another sample that is ‘very close’ to the test patch, the distribution of samples in the area the approximate nearest neighbor is chosen from will be sufficiently close to the distribution around x to yield the same result on average. Because we are choosing a large number of approximate nearest neighbors and we are dealing with a large number of training samples, it is likely that for small ε the decision will probably remain unchanged.

7.3.2 Tangent distance

The similarity between two patches is measured using the Euclidean distance in the baseline method. However this measure is inherently sensitive to all transformations, as e.g. rotation, scaling, brightness changes, etc. The tangent distance as discussed in Section 5.2 incorporates invariance with respect to small global transformations that are known a priori. The transformations modeled are usually the six projective transformations eventually complemented by some problem specific transformations. To our knowledge the tangent distance

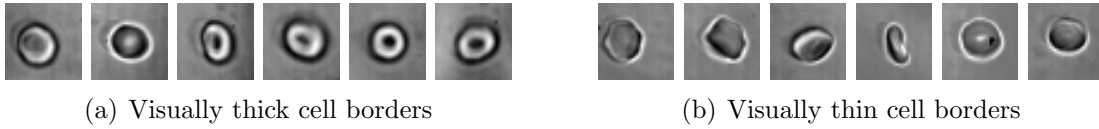


Figure 7.9: Example images of red blood cells with varying thickness of the cell borders as a motivation to add the line thickness tangents for the distance measurement (examples from the class ‘stomatocyte’).

has only been used to compare entire images, so far. Here, it is used to measure the similarity of the image patches to account for transformations of the patches.

We use the tangent distance in combination with the patches for the nearest neighbor search. However, the patches are PCA-transformed before the search. The tangent distance cannot be computed directly on the PCA-transformed vectors, as it is originally designed for images. However, the linearity of the PCA transformation allows us to transform the tangent vectors using the same PCA transformation matrix as is used for the images and then calculate the TD in the reduced space.

On the IRMA data the tangent distance is used with the tangent vectors for affine transformations and additive image brightness. With the baseline method, the result is improved from 10.3% to 7.7% by using the tangent distance and if the kernel densities are used as a probability model the result is improved from 9.7% to 7.4% error. Thus, we observe a significant decrease of recognition error in both conditions. The error rate of 7.4% is the second best published on this data set. Only the use of nonlinear deformation models as presented in Chapter 6 led to better results.

On the red blood cells data the tangent distance is tested with the same tangents as for IRMA and additionally with a tangent for line thickness. This tangent was included here, because the width of the cell borders of the cells varies strongly within each class as illustrated in Figure 7.9. The result obtained is 17.2% error for the Euclidean distance and 13.5% error for the tangent distance using the baseline local patch method. This result represents the best known outcome on this database.

On the USPS data, the baseline approach leads to an error rate of 3.0% [Keysers & Paredes⁺ 02]. Using the tangent distance instead of the Euclidean distance improves the error rate to 2.6%.

We can conclude that the use of the tangent distance leads to consistent improvements for the patch-based classification approach on several data sets.

7.3.3 Kernel densities

The probability model for each patch $p(k|x)$ used in the baseline method is binary, meaning that only the class that the nearest neighbor of the patch belongs to is assigned the probability one and all other classes obtain the probability zero. This might not be the best choice, especially if two classes have similar patches. A solution can be to use more than one nearest neighbor and to weight the neighbors according to their distance from the test sample. A standard method to accomplish this is to use kernel densities.

From the distance function (the Euclidean distance in the baseline method), we compute

Table 7.2: Improvements in error rates using kernel densities.

Corpus	Error (%)	
	baseline	kernel densities
Bloodcells	17.7	17.2
Erlangen	1.2	0.6
IRMA	10.3	9.7

the patch posterior probability as

$$p(k|x) = \frac{\exp[-\frac{1}{2(\lambda\sigma)^2}d(x, \hat{x}_k)]}{\sum_{k'=1}^K \exp[-\frac{1}{2(\lambda\sigma)^2}d(x, \hat{x}_{k'})]}, \quad (7.2)$$

where σ^2 denotes the empirical variance over the training patches and λ is an empirical parameter. We denote by x the test patch vector and \hat{x}_k its nearest neighbor from the class k according to the distance measure d . This approach is derived from the maximum approximation to the kernel density or Parzen window estimator. In the baseline method, a binary probability model is used. $p(k|x)$ is one if the nearest neighbor of x is from class k and zero otherwise, which amounts to direct voting. This is a limiting case of the model (7.2) above for $\lambda \rightarrow 0$. The posterior probability that the image X with the patches x_1, \dots, x_{L_X} is from class k is then computed as in the baseline approach using the sum rule, which is known to be well suited for noisy data [Kittler 98]:

$$p(k|X) = \frac{1}{L_X} \sum_{n=1}^{L_X} p(k|x_n)$$

This approximation led to improvements of the recognition, results are presented in Table 7.2. Here the only task from the Erlangen data that is considered here is object recognition with partial occlusion and changing illumination, because it is the most difficult one. The best error rate on this corpus of 4.8% is reported in [Reinhold & Paulus⁺ 01]. The experimental results show that the use of kernel densities also improves the results when combined with multi-scale patch extraction and with the tangent distance. This suggests that the use of the distance to estimate the probability is generally better than using discrete decisions as in the baseline approach.

Results for the Erlangen data

We briefly discuss the experiments on the Erlangen data with the patch-based approach here, for which a short summary has already been given above. The Erlangen corpus features occlusion, brightness changes, and changing background. Results for this corpus have been presented in [Reinhold & Paulus⁺ 01]. The authors use an approach based on local, multi-resolution Gabor features and a probabilistic object model. We will compare their results to ours, gained with the unconstrained local patch approach using the baseline direct voting and kernel densities. For these tests we used an additional background class for the local patches, which was trained on the background image that is supplied with the training data. The voting (or weighted voting for the kernel density model) then takes

Table 7.3: Patch-based results on Erlangen data in the presence of partial occlusion and changing backgrounds.

approach	25% occlusion	50% occlusion	heterogeneous backgr.
[Reinhold & Paulus ⁺ 01]	0.0	4.8	0.0
patch-based, baseline	0.0	1.2	0.6
patch-based, kernel densities	0.0	0.6	0.0

only those patches (weights) into account that do not belong to the background class. The results obtained for the three conditions with two illuminations are compared to those of [Reinhold & Paulus⁺ 01] in Table 7.3. It can be seen that the results of the patch-based approach have the tendency to perform better than the model based on Gabor features used in [Reinhold & Paulus⁺ 01], although the improvements are unlikely to be significant, because one test image sample corresponds to about 0.6% change in error rate.

7.3.4 Nearest neighbor

In the following sections we briefly discuss several classification methods that are based on the representation of image patch sets by histograms [Deselaers & Keysers⁺ 05a]. These histograms, as described above, contain the counts of closest patch cluster centers.

Using the histograms of image patches as a representation for the images, we can employ a simple nearest neighbor classifier. Usually, the nearest neighbor is a useful benchmark because it is a simple classifier with good performance in many applications. Here, we choose the Jensen-Shannon divergence to compare two histograms. This choice is based on findings in previous experiments [Deselaers & Keysers⁺ 04b], where this measure provided good performance across different tasks. The resulting decision rule for the nearest neighbor is then

$$X \mapsto r(X) = \arg \min_k \left\{ \min_{n=1 \dots N_k} d(h(X), h(X_n)) \right\},$$

where $d(h, h') = \sum_{c=1}^C h_c \log \frac{2h_c}{h_c + h'_c} + h'_c \log \frac{2h'_c}{h'_c + h_c}$.

The nearest neighbor approach differs most from the baseline approach of [Paredes & Perez-Cortes⁺ 01] because here each image defines its own reference based on the contained local patches. For the following methods, the estimation procedures are also based on the individual images (which is not the case for the baseline method) but the resulting model does not reflect this, i.e. no image-to-image comparison is performed, but each class is described by one model.

7.3.5 Naive Bayes

In the following approaches we use Bayes' decision rule

$$\begin{aligned}
 r(X) &= \arg \max_k \{p(k|X)\} \\
 &= \arg \max_k \{p(k) p(X|k)\} \\
 &= \arg \max_k \{p(X|k)\},
 \end{aligned}$$

where the last equality holds due to $p(k) = 1/K$, which can be assumed in the experiments on the Caltech data that were performed using these approaches. Because we use the histogram representation of the images we let $p(k|X) := p(k|h(X))$ and $p(X|k) := p(h(X)|k)$.

In the naive Bayes approach, the assumption is made that the distributions of the feature vector components are conditionally independent. Thus, for the patch representation we assume that $p(X|k) = \prod_{l=1}^{L_X} p(x_l|k)$. As we assume uniform priors, the decision is not changed when we use the product of posterior probabilities. Furthermore, we apply the logarithm to convert the product into a sum:

$$\begin{aligned} r(X) &= \arg \max_k \left\{ \prod_{l=1}^{L_X} p(x_l|k) \right\} = \arg \max_k \left\{ \prod_{l=1}^{L_X} p(k|x_l) \right\} \\ &= \arg \max_k \left\{ \sum_{l=1}^{L_X} \log p(k|x_l) \right\} \\ &= \arg \max_k \left\{ \sum_{c=1}^C h_c(X) \log p(k|c) \right\}. \end{aligned}$$

Here we assume that these patch posterior probabilities are equal for patches within the same cluster: $p(k|x) = p(k|c(x))$. Finally, the cluster posterior probabilities are estimated from the relative frequencies on the training data:

$$p(k|c) = \frac{\sum_{n=1}^{N_k} h_c(X_{kn})}{\sum_{n=1}^N h_c(X_n)}.$$

7.3.6 Generative single Gaussian densities

Another often used baseline classification method is to use a single Gaussian density for the class-conditional probability $p(h|k) = \mathcal{N}(h|\mu_k, \Sigma)$ for each object class with pooled diagonal covariance matrices Σ . The parameters of the model are estimated using the maximum likelihood estimator, maximizing $\prod_{k=1}^K \prod_{n=1}^{N_k} p(h_{kn}|k)$. In classification, Bayes' decision rule is used.

7.3.7 Discriminative training

The approach based on maximum likelihood of the class-conditional distributions does not take into account the information of competing classes during training. We can use this information by maximizing the class posterior probability $\prod_{k=1}^K \prod_{n=1}^{N_k} p(k|X_{kn})$ instead. Assuming a Gaussian density with pooled covariances for the class-conditional distribution, this maximization is equivalent to maximizing the parameters of a log-linear or maximum entropy model as discussed in Section 5.3

$$p(k|h) = \frac{1}{Z(h)} \exp \left(\alpha_k + \sum_{c=1}^C \lambda_{kc} h_c \right),$$

where $Z(h) = \sum_{k=1}^K \exp(\alpha_k + \sum_{c=1}^C \lambda_{kc} h_c)$ is the renormalization factor. Recall that also the generative Gaussian model can be rewritten in this form and we can furthermore always find a generative model that results in the same posterior distribution. The maximizing distribution

is unique and the resulting model is also the model of highest entropy with fixed marginal distributions of the patches. Efficient algorithms to determine the parameters $\{\alpha_k, \lambda_{kc}\}$ exist. We use a modified version of generalized iterative scaling [Darroch & Ratcliff 72]. Bayes' decision rule is used for classification.

Note that [Kölsch 03] describes the use of discriminative training for the baseline patch-based approach without histogramization. Some improvements to the baseline approach could be obtained, but a special treatment for the patches that have not been seen in training had to be introduced for the method to work. By using a clustering of the patches here, we can avoid that problem.

7.3.8 Relation between the models

There exists a strong relation between the structure of the decision rule resulting from the naive Bayes, the Gaussian, and the log-linear model. In all three cases the decision rule can be rewritten as an arg max operation of a linear function of the histogram representation:

$$r(X) = \arg \max_k \left\{ \alpha_k + \sum_{c=1}^C \lambda_{kc} h_c(X) \right\}$$

For the naive Bayes model we have $\alpha_k = 0$ and $\lambda_{kc} = \log p(k|c)$, for the Gaussian model the parameters $\{\alpha_k, \lambda_{kc}\}$ are a function of the parameters $\{\mu_k, \Sigma\}$, and for the log-linear model the parameters are trained directly.

In this formulation of the decision rule, evidently, patches assigned to those clusters c that have the highest absolute difference of coefficients $|\lambda_{kc} - \lambda_{k'c}|$ contribute the most to the discrimination between the classes k and k' according to the model. This correspondence is used to visualize the most discriminative patches, where the sign of the difference $\lambda_{kc} - \lambda_{k'c}$ determines if the patch cluster contains indicators for class k or k' .

The relation to the baseline method of [Paredes & Perez-Cortes⁺ 01] is also worth to be discussed here. In the baseline method, for each test image patch we determine the closest training patch and count a vote of one for the class that patch belongs to. In the extension using kernel densities, the weight of this vote is adjusted according to the distances observed.

The histogram-based methods differ from this procedure in the following way: for each test image patch we determine not the closest training patch but the closest cluster center. Associated with this cluster center is then a weighted vote for each of the classes. The determination of the weights is based on different procedures that take into account the distribution of the patches among the clusters in the training images.

The clustering of image patches implies a certain generalization of the weights and makes the training procedures more reliable because it serves as smoothing. If we try to view Paredes' method as a histogram approach this is also possible: if we let each training patch form a single cluster and assign the weights $\lambda_{kc} = 1$ if patch number c is from class k and 0 otherwise, we arrive at the baseline method. This implies that the two main differences between Paredes' method and the histogram-based approach using discriminative training are

- smoothing of the influence of each training patch by clustering and
- discriminative training of the voting weights for each patch.

7.3.9 Extensions to the histogram-based approach

We investigated several extensions to the patch-based approach using histograms [Deselaers & Keysers⁺ 05b]. Most importantly, the method is substantially improved by adding multi-scale patches so that it better accounts for objects of different sizes as already discussed above. Other extensions tested include using Sobel-filtered patches, the generalization of histograms using smoothing, and the method to account for varying image brightness in the PCA domain as discussed above. These extensions improve results significantly:

- we use image patches in various scales enabling us to account for objects at different scales (see above);
- to account for different lighting conditions in the images, we incorporate a method for brightness normalization (see above);
- we use Sobel-filtered images in addition to the gray values to account for edge structures in the images (see below);
- as the histograms created can be very sparse (e.g. there are approximately 1,000 data points in a 4,096-bin histogram), we generalize the histograms to use non-binary bin assignments (see below).

Sobel-filters

In many applications of pattern recognition, derivatives can improve classification performance significantly, e.g. in automatic speech recognition, derivatives are normally used. Also in the recognition of handwritten characters, image gradients can strongly improve the results, as the local derivatives allow mapping of edges to edges (cp. Section 6.2.2). To take advantage of these effects, we enrich the patches by their horizontally and vertically Sobel-filtered versions. That is, the data is tripled by adding the pixel values of the horizontally and vertically Sobel-filtered patches. Then, the PCA transformation is applied to all three versions (gray values, horizontal Sobel, vertical Sobel) at once and the dimensionality is reduced to 40 as for the other approaches.

Smoothed histograms

A weakness of the original histogram-based approach might be that the histograms are high-dimensional and may be very sparse, e.g. the histograms have 4096 bins but only 800 patches (2400 for multi-scale patches) are extracted per image. Thus, most of the bins are empty and do not contribute to the result. To counteract this and have smoother histograms, we generalize the histograms to use non-binary bin assignments, i.e. patches do not only contribute to their closest cluster center, but to all cluster centers that are sufficiently close. Given an image patch and the Euclidean distance $d_c := d(x, c)$ to cluster center c the corresponding histogram count h_c is updated as

$$h_c \leftarrow h_c + \frac{\exp(-\frac{d_c}{\alpha})}{\sum_{c'} \exp(-\frac{d_{c'}}{\alpha})}.$$

Figure 7.10 shows a discrete and the corresponding smoothed histogram. It can be clearly seen that the discrete histogram has many bins with low counts, which is not the case for the smoothed version.

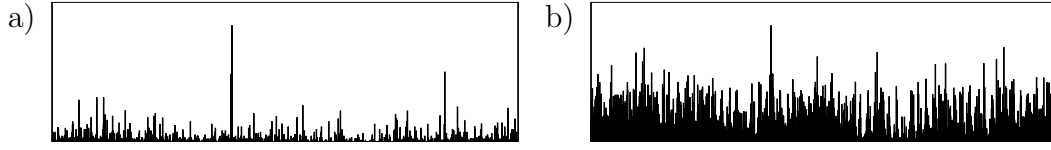


Figure 7.10: Discrete patch histogram (a) in comparison to smoothed histogram (b).

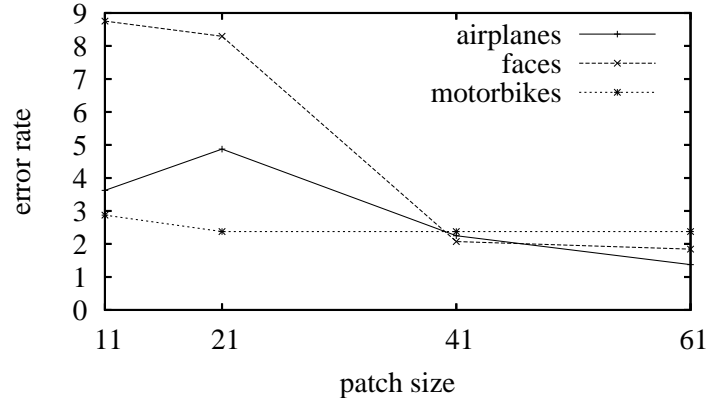


Figure 7.11: Discriminative model on the unscaled data: effect of patch size on the error rate.

7.4 Experiments and results using histograms

We already presented some results along with the discussion of extensions to the baseline method of [Paredes & Perez-Cortes⁺ 01]. In this Section we discuss the results obtained with the patch-based approach using histograms [Deselaers & Keysers⁺ 05a, Deselaers & Keysers⁺ 05b].

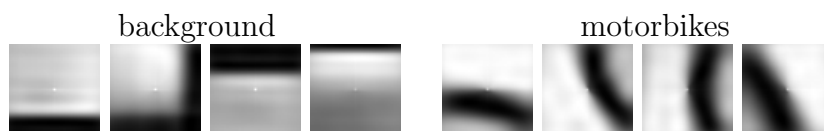
First, we evaluate two series of experiments for each of the Caltech tasks [Deselaers & Keysers⁺ 05a]: in the first series, each image retains its original size and in the second series each image is scaled to the height of 225 pixels, the mean height of the input images. For each of the tasks, first the image patches are extracted. The PCA is determined using the training data patches only and all patches are processed using the PCA coefficients. Then, the image patches from the training data are clustered as a prerequisite for the creation of the histograms. Finally, we create the patch histograms for training and test data.

One parameter that must be determined for the experiments is the image patch size. We first use the images of original size, extract the patches as described above, and apply the classification methods to these data. Figure 7.11 shows the error rate for these experiments using the discriminative model. The other classifiers behave similarly but yield larger error rates. It can be observed that the largest patch size (61×61) performs best. The resulting error rates for this patch size are shown in Table 7.4. They show that the discriminative model outperforms the other methods and are also very competitive with error rates presented in the literature for the same tasks as shown in Table 7.5. The second best approach is the naive Bayes model.

Visualizing the patches that are most discriminative according to the difference in coefficients from the discriminative approach shows an interesting effect. This effect results from

Table 7.4: Error rates, size 61×61 , original data, 512 clusters.

method	airplanes	faces	motorbikes
global patch search	7.8	18.4	15.8
nearest neighbor	6.1	6.2	9.6
naive Bayes	4.6	5.8	6.9
generative Gaussian	15.4	30.0	19.0
discriminative model	1.4	1.8	2.4

Figure 7.12: Most discriminative patches for size 61×61 for the classes background (left) vs. motorbikes (right).

the property that the images of the background class are generally smaller than the images from the other classes. The four most discriminative patches for the background and the motorbikes class are shown in Figure 7.12. It can be clearly observed that the patches for the background class show image borders, while the patches for the motorbike class show parts of the wheels. On the one hand this shows that visually meaningful patches are learned to be discriminative for motorbikes. On the other hand, the significant difference in size is also learned by assigning more importance to patches that contain image borders and corners. This explains why enlarging the patches improves performance: the larger the patches are, the more of the image border is contained in the patches and thus for the smaller background images the most discriminative patches are those showing large amounts of image border.

Although we may state that the algorithm in fact learns to effectively discriminate between background and foreground images, this is not the result we are trying to obtain. While we believe that the error rates are still valid results, we are interested in the performance of the algorithm if it cannot exploit the difference in size of the image classes.

To avoid the effect of learning the borders of background images, we scale all images to the common height of 225 pixels, approximately the mean height across the data. Repeating our experiments to determine the best patch size, we now obtain the error rates shown in Figure 7.13 for the discriminative model. Now larger patch sizes no longer perform better. The error rates for the smallest evaluated patch size of 11×11 are presented in Table 7.5 and compared to those from other publications and the best error rates from the first experiment. Performing further experiments with more cluster centers (thus using histograms with more bins) we observe that the error rate improves for the discriminative approach. The other methods, especially the generative Gaussian approach, improve only slightly, if at all.

Again, the discriminative approach performs best among the investigated methods and gives competitive results. Especially the error rate of 1.5% for the motorbike task is one of the lowest published error rates. The second best method now differs from task to task. Note that most other publications give ROC equal error rates. The error rates presented here do not involve any adjustment of a threshold but still are very close to this concept: the misclassifications within the two classes are 14:16 for airplanes, 13:18 for faces, and 7:13 for motorbikes.

In Figure 7.14 the top four discriminative patches are shown for each of the three tasks.

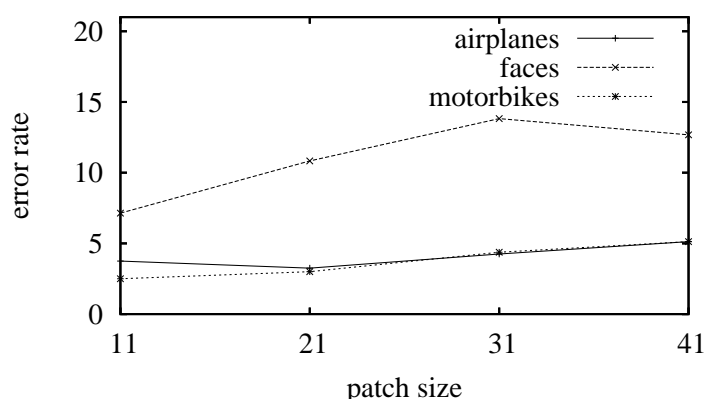


Figure 7.13: Discriminative model on the scaled data: effect of patch size on the error rate.

Table 7.5: Error rates on scaled Caltech data with 512/4096 clusters in comparison to results from other publications.

method	airp.	faces	mot.
512 clusters:			
patch search	4.8	8.5	21.5
nearest neighbor	9.4	18.7	5.5
naive Bayes	8.5	17.1	9.9
generative Gaussian	5.8	17.5	7.6
discriminative model	3.8	7.1	2.5
4096 clusters:			
nearest neighbor	11.6	19.9	14.5
naive Bayes	5.6	11.3	7.5
generative Gaussian	37.4	48.2	49.9
discriminative model	2.6	5.8	1.5
statistical model [Fergus & Perona ⁺ 03]	9.8	3.6	7.5
texture features [Deselaers & Keysers ⁺ 04b]	0.8	1.6	7.4
segmentation [Fussenegger & Opelt ⁺ 04]	2.2	0.1	10.4
discriminative model (Table 7.4)	1.4	1.8	2.4

We can observe that the patches for the foreground allow a meaningful visual interpretation in most of the cases: the airplane images contain more horizontal structures than the background images such that patches containing strong horizontal gradients are chosen to receive large weights. The first patch of the face class shows a patch that resembles an eye. This observation becomes clearer if we look at some patches from the training data that are assigned to this cluster as shown in Figure 7.15. Clearly, the algorithm has automatically learned that the eye is the visually most important patch type to distinguish faces from background images. The second face patch can be interpreted as a part of the hair/forehead line while the third and fourth are not easily interpreted. For the motorbike task, all four patches show diagonal wheel/rim structures, which typically do not occur in background images. The most discriminative background patches change for the three tasks, which is due to different training images and to the discriminative training: For example, the first two background patches in the faces task are strong indicators for background versus faces,

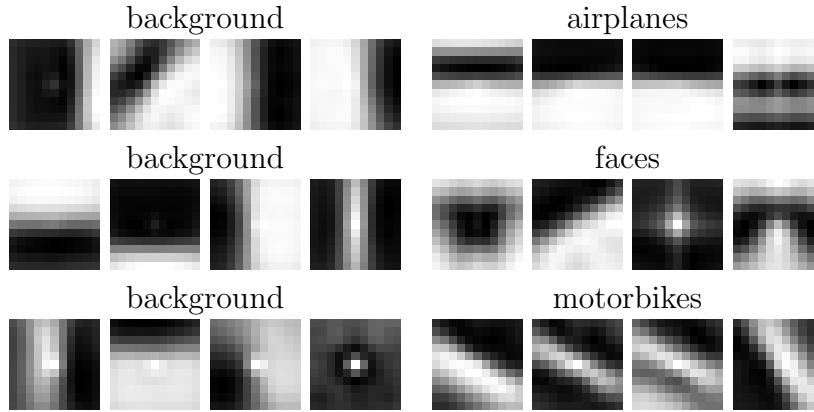


Figure 7.14: Most discriminative patches for the Caltech data (background and object class).

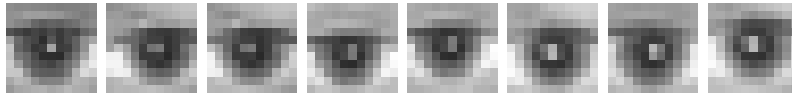


Figure 7.15: Patches from the cluster most discriminative for faces. The algorithm automatically learns that the eye is the most relevant feature for a human face.

Table 7.6: Error rates for the IRMA data in a key experiment.

method	ER[%]
discriminative patch-based model	23
image distortion model	18

but this would not be true in the airplanes task, because here vertical structures are indicators for airplanes. The bright centered dot in some of the patches is due to the PCA reconstruction and the mean image computed from patches supplied by the interest point detector, favoring images with strong gradients.

Figure 7.16 shows typical examples of each of the three tasks with those positions marked at which highly discriminative patches are extracted. We can observe that strong foreground indicators are located at horizontal structures for the airplanes task, at the eyes, hair/forehead, and clothes for the faces task, and at the wheel/rim and other diagonal structures for the motorbikes task. For the incorrectly classified images it can be observed that in the airplane image many vertical structures are found, that the face is too dark in comparison to the background, and that in the motorbike image a large amount of background patches are present.

To observe the performance of our method on a task with more than two classes, we also performed a key experiment using one setup of the IRMA data consisting of 2,832 training images and 1,016 test images and 24 classes which are very unevenly distributed. Table 7.6 shows the results using the above settings on the medical radiograph data. The error rate of 23% compares well to an error rate of 18% when using the image distortion model which is a method that is known to produce excellent results on this corpus. Note that the obtained error rate of 23% was obtained without any adaptation of parameters and only serves to show that the approach can also be used for tasks with more than one class. The very good



Figure 7.16: Typical examples of correct (top) and incorrect (bottom) classifications with positions of most discriminative patches for object (yellow) and background class (red).

performance of the patch-histogram method in the 2005 ImageCLEF competition that uses the larger IRMA data set underlines the applicability of the approach also for the IRMA data with a large number of classes. In the automatic annotation task of the 2005 ImageCLEF evaluation of content-based medical image retrieval using the IRMA data, the error rate of 12.6% obtained by the IDM was the best among 42 results submitted. The third best result with an error rate of 13.9% was achieved using the method based on local patch histograms and discriminative training as discussed here.

The experiments show that the approach works well for the data presented, where the foreground object forms a significant portion of the input image. It may be argued that it will be problematic for the approach to deal with cases where this is not the case. This (so far hypothetical) effect might be alleviated by using a significantly larger amount of training data. Furthermore, to our knowledge this problem will occur for all generic learning and recognition approaches with the possible exception of those approaches that are tuned toward a specific application like face detection.

Evaluation of the extensions

Using the discussed extensions to the patch-histogram approach with discriminative training we obtained 59% relative reduction of the error rate on average and were able to obtain a new best error rate of 1.1% on the Caltech motorbikes task [Deselaers & Keysers⁺ 05b].

In Table 7.7 we present the results for the basic patch-histogram method above and the results obtained with the proposed extensions in comparison to results from the literature. All experiments were carried out using 4096-dimensional histograms.

Comparing the results using multi-scale patches to the results from the baseline method where only patches of one size were extracted, a clear improvement can be seen in two of the three tasks. The result for the motorbikes task was not improved. These results can be explained by the fact that the scale of the motorbikes is very homogeneous and thus multi-scale patches cannot improve the results. Due to the positive results, all experiments

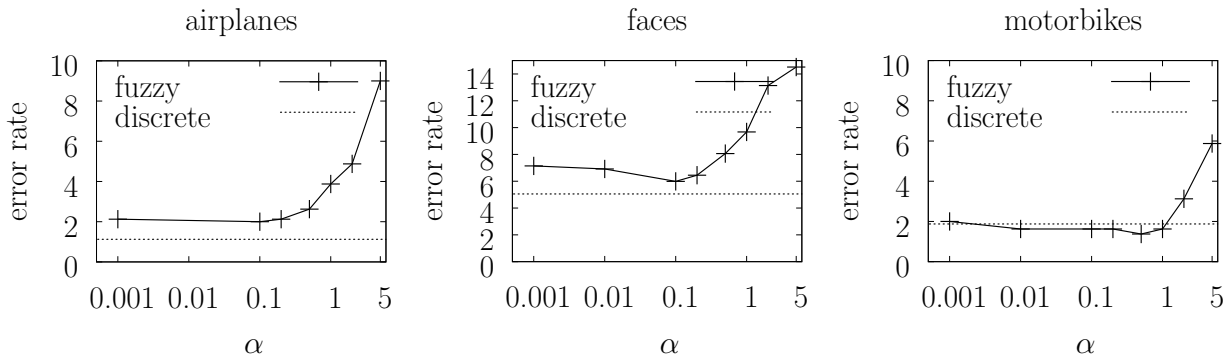


Figure 7.17: Error rates for the Caltech tasks depending on the smoothing factor α in smoothed histograms.

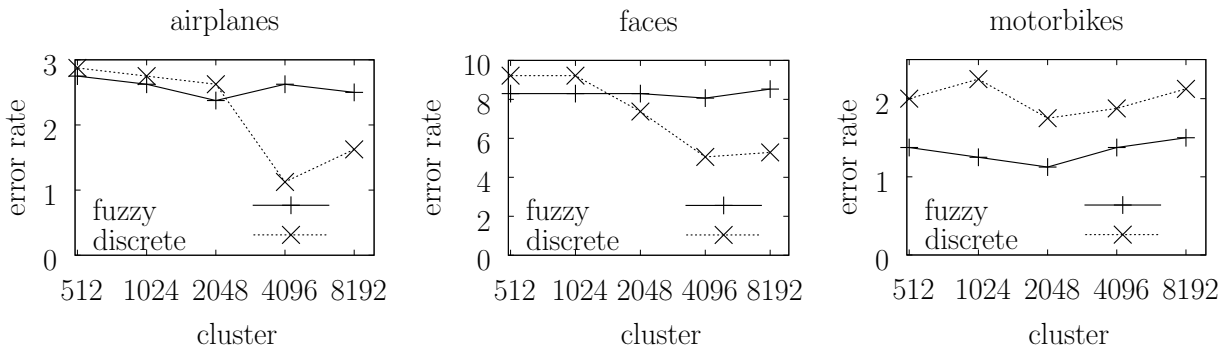


Figure 7.18: Error rates for the Caltech tasks depending on the number of clusters using smoothed histograms and discrete histograms.

Table 7.7: Summary of the results for the extensions to the histogram-based approach and comparison to results from other publications.

Method	error rate [%]		
	airplanes	faces	motorbikes
Discriminative Model [Deselaers & Keysers ⁺ 05a]	2.6	5.8	1.5
+ multi-scale patches	1.1	5.0	1.9
+ multi-scale & Sobel features	4.5	13.6	2.6
+ multi-scale feat. & smoothed hist.	2.6	8.1	1.4
+ multi-scale & brightness norm.	1.4	3.7	1.1
Statistical Model [Fergus & Perona ⁺ 03]	9.8	3.6	7.5
Texture features [Deselaers & Keysers ⁺ 04b]	0.8	1.6	7.4
Segmentation [Fussenegger & Opelt ⁺ 04]	2.2	0.1	10.4

in the following were performed using multi-scale patches.

The results where Sobel features were used are worse than those from the baseline method. This unexpected result may be due to the combined PCA transformation of brightness and contrast information.

In a next step, we evaluated the possible advantages of smoothed histograms. Figure 7.17 shows the effect of choosing different parameters α to smooth the image patch histograms. In these experiments we used 4096 clusters and multi-scale patches. The figures show that the smoothed histograms do not improve the results in this setting. In Figure 7.18 we compare smoothed histograms with discrete histograms using different numbers of histogram bins. It can be seen that smoothed histograms outperform discrete histograms in the case of only few clusters. As the clustering process is computationally very expensive, but the creation of smoothed histograms is not more expensive than the creation of discrete histograms given a cluster model, smoothed histograms can be used to obtain reasonable results when computing power for the training is limited. It can also be clearly seen that the number of clusters has less impact on the classification performance when smoothed histograms are used. The results in Table 7.7 show that the use of smoothed histograms does not yield a significant improvement over the baseline method.

Finally, we evaluated the proposed method for brightness normalization in the PCA domain. The results in Table 7.7 show that strong improvements are possible here. Especially for the faces task a significant improvement is observed as some of the images were taken indoors and some images were taken outdoors.

Nevertheless, the result presented in [Fussenegger & Opelt⁺ 04] is much better for the faces task, because a specialized method for face detection was applied to the data.

Results of the PASCAL Visual Object Classes Challenge

The results of the 2005 evaluation within the PASCAL Visual Object Classes Challenge underline the good performance of the approach patch-histogram approach using discriminative training¹.

The goal targeted in the PASCAL Visual Object Classes Challenge is to recognize objects from a number of visual object classes in realistic scenes (i.e. not pre-segmented objects). It is fundamentally a supervised learning problem in the sense that a training set of labeled images was provided. The task is to decide whether an object of a certain class is present in an image or not.

The members of the image understanding group at the Lehrstuhl für Informatik VI of the RWTH Aachen University participated in this evaluation using the patch-histogram approach with discriminative training and obtained very good results. The detailed results are given in Table 7.8 and 7.9. More information on the tasks and the submissions of other groups can be obtained from the PASCAL website at <http://www.pascal-network.org> and the document describing the results [Everingham & Gool⁺ 05]. The measures used in the tables are the equal classification rate (ECR) which corresponds to one minus the equal error rate (EER) and the area under the ROC curve (AUC). For both performance measures higher values are better. Note that in the tables below the results of INRIA are not exactly comparable to the other results because INRIA performed several runs on the test data and only submitted the best runs.

¹<http://www.pascal-network.org/challenges/VOC/>,
<http://www-i6.informatik.rwth-aachen.de/~deselaers/pascal-challenge05.html>

Table 7.8: Results of PASCAL challenge (Task 1). In this task, the goal was classification (is the object present or not), allowed training data included the ‘train’ and ‘eval’ data and tests were done on the ‘easier’ test set: “This test set is taken from the same distribution of images as the training and validation data, and is expected to provide an ‘easier’ challenge.”

Task 1.1: Classification of motorbikes			Task 1.2: Classification of bicycles		
ECR	AUC	Group	ECR	AUC	Group
0.977	0.998	INRIA	0.930	0.982	INRIA
0.972	0.994	Southampton	0.930	0.981	INRIA
0.968	0.997	INRIA	0.918	0.974	INRIA
0.964	0.996	INRIA	0.895	0.961	Southampton
0.949	0.989	Southampton	0.868	0.954	<i>Aachen</i>
0.940	0.987	<i>Aachen</i>	0.868	0.943	Southampton
0.940	0.985	Southampton	0.851	0.930	Southampton
0.926	0.979	<i>Aachen</i>	0.842	0.931	<i>Aachen</i>
0.921	0.974	Helsinki	0.816	0.895	Helsinki
0.917	0.970	Helsinki	0.795	0.891	Helsinki
0.912	0.952	Helsinki	0.781	0.864	Helsinki
0.903	0.966	Ankara	0.781	0.822	Ankara
0.898	0.960	Helsinki	0.767	0.880	Helsinki
0.875	0.945	MPI Tübingen	0.754	0.838	MPI Tübingen
0.856	0.882	Darmstadt	0.689	0.724	Edinburgh
0.829	0.919	Darmstadt			
0.722	0.765	Edinburgh			

Task 1.3: Classification of people			Task 1.4: Classification of cars		
ECR	AUC	Group	ECR	AUC	Group
0.917	0.979	INRIA	0.961	0.992	INRIA
0.917	0.972	INRIA	0.938	0.987	INRIA
0.901	0.965	INRIA	0.937	0.983	INRIA
0.881	0.943	Southampton	0.925	0.978	<i>Aachen</i>
0.861	0.936	<i>Aachen</i>	0.920	0.979	<i>Aachen</i>
0.861	0.928	<i>Aachen</i>	0.913	0.972	Southampton
0.857	0.921	Helsinki	0.909	0.971	Helsinki
0.850	0.927	Helsinki	0.908	0.968	Helsinki
0.845	0.919	Helsinki	0.901	0.961	Southampton
0.841	0.925	Southampton	0.898	0.959	Southampton
0.833	0.931	Helsinki	0.869	0.956	Helsinki
0.833	0.918	Southampton	0.847	0.934	Helsinki
0.803	0.816	Ankara	0.840	0.920	Ankara
0.731	0.834	MPI Tübingen	0.831	0.918	MPI Tübingen
0.571	0.597	Edinburgh	0.793	0.798	Edinburgh
			0.644	0.717	Darmstadt
			0.548	0.578	Darmstadt

Table 7.9: Results of PASCAL challenge (Task 2). In this task, the goal was classification (is the object present or not), allowed training data included the ‘train’ and ‘eval’ data and tests were done on the ‘harder’ test set: “This test set has been freshly collected for the challenge. It is not therefore expected to have the same distribution as the training data, and should provide a ‘harder’ challenge.”

Task 2.1: Classification of motorbikes

ECR	AUC	Group
0.798	0.865	INRIA
0.769	0.829	<i>Aachen</i>
0.767	0.825	<i>Aachen</i>
0.698	0.765	MPI Tübingen
0.698	0.710	Edinburgh
0.683	0.716	Darmstadt
0.663	0.706	Darmstadt
0.635	0.675	Helsinki
0.624	0.693	Helsinki
0.614	0.666	Helsinki
0.594	0.637	Helsinki

Task 2.2: Classification of bicycles

ECR	AUC	Group
0.728	0.813	INRIA
0.667	0.724	<i>Aachen</i>
0.665	0.729	<i>Aachen</i>
0.616	0.654	MPI Tübingen
0.616	0.645	Helsinki
0.604	0.647	Helsinki
0.575	0.606	Edinburgh
0.527	0.567	Helsinki
0.524	0.546	Helsinki

Task 2.3: Classification of people

ECR	AUC	Group
0.719	0.798	INRIA
0.669	0.739	<i>Aachen</i>
0.663	0.721	<i>Aachen</i>
0.614	0.661	Helsinki
0.601	0.650	Helsinki
0.591	0.655	MPI Tübingen
0.587	0.630	Helsinki
0.574	0.618	Helsinki
0.519	0.552	Edinburgh

Task 2.4: Classification of cars

ECR	AUC	Group
0.720	0.802	INRIA
0.716	0.780	<i>Aachen</i>
0.703	0.767	<i>Aachen</i>
0.692	0.744	Helsinki
0.677	0.717	MPI Tübingen
0.676	0.740	Helsinki
0.655	0.709	Helsinki
0.644	0.694	Helsinki
0.633	0.655	Edinburgh
0.551	0.572	Darmstadt

7.5 Combination of patch-based classification and tangent distance

Statistical classification using tangent vectors and classification based on local patches are two successful methods for various image recognition problems. These two approaches tolerate global and local transformations of the images, respectively. Tangent vectors can be used to obtain global invariance with respect to small affine transformations and line thickness, for example. On the other hand, a classifier based on local representations admits the distortion of parts of the image. From these properties, a combination of the two approaches seems very likely to improve on the results of the individual approaches. In this section we show the benefits of this combination by applying it to the USPS task [Keysers & Paredes⁺ 02]. An error rate of 2.0% is obtained, which is among the best results published for this dataset.

The relevant transformations for handwritten digits include the two cases of

- global transformations of the image, e.g. scale, rotation, slant, and
- local transformations of the image, e.g. clutter or missing parts.



Figure 7.19: Examples of digits misclassified by the local patch approach, but correctly classified by the tangent distance classifier (first row, note the variation in line thickness and affine changes) and vice versa (second row, note the missing parts and clutter).

We have discussed two classification methods that are particularly suitable for the two types of transformations: the use of tangent vectors for global invariance and the use of local patches tolerating local changes (we use the baseline method here [Paredes & Perez-Cortes⁺ 01]). Because these two methods deal with different types of transformations it seems especially useful to combine the results of the classifiers. The combination of these two classifiers is evaluated here on the USPS database. For example, tangent distance is able to cope with different line thicknesses very well, while the local patch approach can tolerate missing parts (like segmentation errors) or clutter. Figure 7.19 shows some of the errors in which the two classifiers differed in the classification correctness of their decisions.

The experimental setup used here was comparably simple. The best result obtained in [Keysers & Dahmen⁺ 00b] (2.2% error rate) was already based on classifier combination on the basis of class posterior probabilities. Hence, it was only necessary to include the results of the local patch approach (which yielded an error rate of 3.0%) in the combination. Note that in this case, not the local variance but the grayvalue of the patch center pixel itself was used as a selection criterion for the local patches: dark pixels (with low gray value) are selected in order to determine points on the trace of the handwritten character. We used the decision based on the local patches with two votes, one classifier with one-sided tangent distance and two classifiers with two-sided tangent distance. Using majority vote as combination rule, ties were arbitrarily broken by choosing the class with the smallest class number. With this approach, we were able to improve the result from 2.2% to 2.0%. Table 3.2 on page 10 shows the error rates in comparison to those of other methods.

Note that the improvement from 2.2% to 2.0% is not statistically significant, as there are only 2,007 test samples in the test set (the 95% confidence interval for the error rate on this experiment is [1.4%, 2.8%]). Furthermore, it must be admitted that these improvements seem to result from “training on the testing data”. Against this impression we may state several arguments: On the one hand, only few experiments using classifier combination were performed here. Secondly, there exists no development test set for the USPS dataset. Therefore, all the results presented on this dataset (cp. Table 3.2) must be considered as training on the testing data to some degree and therefore a too optimistic estimation of the real error rate. This adds some fairness to the comparison. Despite these drawbacks, the presented results are interesting and important in our opinion, because the combination of two classifiers that are able to deal with different transformations of the input (cp. Figure 7.19), was able to improve on a result which was already very optimized.

7.6 Conclusion

We investigated enhancements for patch-based image classification. We showed that recognition is improved when using multi-scale patches for the IRMA database. Using kernel densities instead of direct voting also improves recognition on all three used databases. Finally, applying the tangent distance instead of the Euclidean distance leads to improvements as well. If the DCT is used instead of the PCA for feature dimensionality reduction, the result deteriorates slightly, but the PCA estimation step on the training data can be saved. The experiments show that the local feature-based approach is well suited to cope with partial occlusion. Observing that in each of the components feature extraction, distance measure, and probability model, improvements of the baseline method are possible, we may assume that the main point of the method (that already makes the baseline method very powerful) is the following: in the search for similar image parts, all patches of one class are hypothesized at the same time and not on a per-image basis, neglecting the position of the extracted patches.

We presented a method for object classification that uses image patches and fully automatically learns which patches are discriminative for the given object classes. We compared the method to other methods using the same patches. The obtained recognition performance compares favorably to those reported in other publications, in particular we observe a 1.5% error rate on the motorbikes task.

In the first series of experiments, the discriminative training learns that the size of the image is a very discriminative feature for the classification by assigning a large weight to border and corner patches, which is not intended. To avoid this effect, we scale the images to a common height in the second series of experiments. From the resulting clusters it can be observed that visually meaningful parts of the objects are learned, e.g. for faces the eyes and for motorbikes extracts from the wheels are most discriminative.

We extended the method for object classification in cluttered scenes toward different directions. We proposed to use multi-scale patches, Sobel patches, generalized histograms, and brightness normalization. We could show experimentally that multi-scale patches and brightness normalization strongly improve the results, and that generalized histograms can be used to reduce computation time in training with only slight degradation in classification performance. Using Sobel features did not improve the results. It might be an interesting option to apply PCA transformation to the gray values and Sobel features separately. Furthermore, we plan to explicitly model local variability in images. Another point where improvements are probably possible is to consider spatial information along with the extracted patches. Currently all spatial information is discarded.

We may also want to use local representations of the image content to detect more than one object in an image. The most straightforward idea to do so is to directly transfer the voting approach and set a threshold for the number of votes necessary to detect an object. We can also combine the approach with other methods and try to segment regions that contain votes for the same object or learn templates of local votes and detect these in a larger image. These approaches and their evaluations are described in [Deselaers & Keysers⁺ 03]. It seems that in this case more sophisticated methods as e.g. presented in [Leibe & Schiele 04] are more appropriate. Another alternative is the use of holistic models as presented in the following chapter.

Obviously, there exist alternatives to algorithmic choices made in the proposed method. For example, different interest point detectors can be used. However, experiments in other

domains suggest that the choice of the interest point detector is not critical and often the local gray value variance or entropy is already a sufficient criterion, provided that enough image patches are extracted [Paredes & Perez-Cortes⁺ 01, Paredes & Keysers⁺ 02]. Furthermore, the geometrical relation between the extracted patches is completely neglected in the approach presented here. While this relation could be used to improve classification accuracy, it remains difficult to achieve an effective reduction of the error rate in various situations by doing so.

8 Holistic scene analysis

Gordon's great insight was to design a program which allowed you to specify in advance what decision you wished it to reach, and only then to give it all the facts. The program's task, [...] was simply to construct a plausible series of logical-sounding steps to connect the premises with the conclusion.

– Douglas Adams, Dirk Gently's
Holistic Detective Agency, 1987

Up to now, approaches to classification, indexing, or retrieval are usually not based on the objects present in the image, but mostly on low-level features derived from color or texture. This is due to the still unsolved problem of automatic segmentation of objects in presence of an inhomogeneous background [Smeulders & Worring⁺ 00]. Approaches to image object recognition mostly rely on manually pre-segmented data for training. These algorithms also perform best for homogeneous or static background but do not take into account that ignoring background information in recognition can cause classification errors.

In this chapter, we present a holistic statistical model for the automatic analysis of complex scenes that addresses the analysis of images with complex background. Here, ‘holistic’ refers to an integrated approach that does not take local decisions about segmentation or object transformations. Starting from Bayes’ decision rule, which is the best we can do to minimize the error rate, we avoid explicit segmentation and determination of transformation parameters but instead consider these as integral parts of the decision problem. This is done to avoid incorrect local decisions. This holistic approach takes into consideration experiences from speech recognition, where explicit segmentation of ‘objects’ (words) and background is neither done in training, nor in recognition. Note, though, that treatment of distortions and transformations is computationally significantly more demanding in 2D (images) than in 1D (speech signals) as discussed in Section 6.4.

We develop an appearance-based approach that explains all pixels in the given scene using an explicit background model. This allows the training of object references from unsegmented data and recognition of complex scenes. We present empirical results on different databases obtaining state-of-the-art results on two databases where a comparison to other methods is possible. To obtain quantifiable results for object-based recognition, we introduce a new database with subsets of different difficulties [Keysers & Motter⁺ 03].

8.1 Introduction and related work

Automatic image analysis is the study of computer algorithms capable of determining the content of a given static digital image. While this problem remains unsolved for arbitrary images, solutions for many special tasks are available, e.g. the automated analysis of postal envelopes or the detection of human faces in images [Yang & Kriegman⁺ 02]. One application area is the analysis of medical images, e.g. images of blood cells or content-based



Figure 8.1: Example image of an object (a cup) in front of an inhomogeneous background from the COIL-RWTH data.

medical image retrieval [Keysers & Dahmen⁺ 03]. We discuss the use of a generative statistical model [Keysers & Dahmen⁺ 01b, Keysers & Dahmen⁺ 03, Keysers & Motter⁺ 03] that takes into account all image pixels (hence *holistic*) in the implicit decision making process when determining the content of an image. This approach should be contrasted with the ad hoc solutions to special tasks that often obtain very good results, like the use of two-step procedures that first separate object and background and then classify the segmented object. Although such specialized methods are often suitable, we believe that it is more appropriate to first formulate a holistic model and then make the adequate refinements explicit, as for example the assumption of a uniform background. Here we show the development and application of a holistic statistical model for image analysis.

While the human vision system is excellent in analyzing images, this task remains a hard problem for computers in the case of arbitrary images. Image analysis is one important subtopic of the discipline of computer vision. Here, by ‘image analysis’ we denote the following task: Given a digital image, determine a partitioning of the image and an assignment of each of the partitions to one of a finite set of classes. The assigned classes should denote the content of the partition. The set of classes will vary from task to task. In many cases we are only interested in the assigned classes and not in the image partitioning. For example, in a postal automation process, it may be sufficient to determine the postal code of the recipient’s address in an image retrieval task we may only want to know that the image contains an airplane, disregarding its position. In other words, we are interested in a decision rule $r : X \mapsto (M, c_1^M)$, which assigns to each image $X \in \mathbb{R}^{I \times J}$ of size $I \times J$ a sequence of M class labels $(k_1, k_2, \dots, k_M) =: k_1^M$.

Figure 8.1 shows an example image of an object in front of an inhomogeneous background. The example is from the COIL-RWTH data set, in which the object was artificially pasted in front of the background image. Note that conventional approaches that rely on segmentation will have problems because of the background clutter. Conventional approaches in this context fall into two categories: First, approaches to image object recognition often rely on the possibility to segment the objects from the background or on the presence of static background. Second, in image indexing or retrieval the standard approach is to use features that are derived from color or texture, and not directly from clues about the presence of objects. Note that the discriminative patch-based approach as described in Chapter 7 partly overcomes this restriction. In this chapter we discuss the holistic approach that allows us to automatically determine object references from only weakly labeled data (i.e. it is only known that an object is present in the image, not where it is located). This is achieved by employing an appearance-based holistic statistical model without explicit segmentation.

In the past years, statistical methods have been successfully applied to the task of ob-

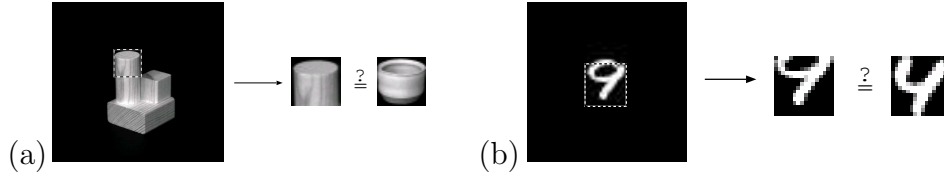


Figure 8.2: Problems with interdependence in recognition: (a) Only a small part of the original object is explained, possibly resulting in a misclassification. (b) Effect of small localization errors on the classification result.

ject recognition in images. In most cases, the methods used are either specialized on the recognition of single objects in a known position or on determining the position of a known object in the image. The approach presented here includes the approaches mentioned above as special cases and is furthermore designed for the recognition of multiple objects in an image, although the computational demand grows strongly with the number of objects and therefore the experiments usually consider the case of the detection of one object. This task is strongly related to the recognition of objects in the presence of varying background. Usually, this type of recognition is performed in combination with a separate segmentation step, which is inherently error-prone. To carry out scene recognition, an automatic system is faced with the interdependence of several operations, including segmentation, object detection and recognition and transformations of the objects. Therefore, the main concept of the approach presented here is to generalize the classical Bayes' decision rule to more complex object recognition tasks, i.e. to choose among all possible object and background configurations the one that maximizes the probability that the image was produced by this configuration. Thus, a meaningful segmentation of the image is implicitly determined. The need for such a holistic approach to image recognition (i.e. the necessity to explain the whole image, integrating segmentation and recognition) has been recognized before, e.g. [Hinton & Ghahramani⁺ 00]. It is also known that weaknesses of many current recognition systems are the reliance on a separate segmentation step and the removal of context information. These problems are illustrated in Figure 8.2 and further problems include recognition of partially occluded and very close objects in an image. In spite of the increased computational complexity of the holistic approach, the success of the holistic paradigm in speech recognition, where interdependence between time alignment, word boundaries and syntactic constraints exist, shows that the additional effort may well be justified [Ney & Ortmanns 00]. From the domain of automatic speech recognition, there are also various methods known to reduce the necessary amount of computations.

Recently, due to the increased computing power available and growing interest in the automatic analysis of images, different approaches related to the one presented here have been developed independently.

A statistical model for object recognition in the presence of heterogeneous background and occlusions is presented in [Reinhold & Paulus⁺ 01] which is then simplified to having local background/foreground decisions and thus amounts to a local thresholding approach. The authors use wavelet features to determine the local probabilities of a position in the image belonging to an object or to the background. The background is modeled by a uniform distribution. The assumption of statistical independence of the object features is reported to produce best results. The problem of automatic training in presence of heterogeneous

background is not addressed. The authors report 0% error rate on a classification task in the presence of rotation and translation.

A similar model to the one presented here has been independently proposed in [Frey & Jojic 03]. The authors introduce transformed mixtures of Gaussians that are used to learn representations on different databases of image data. They provide a detailed description of the statistical model. They consider only translations for an image database with background but do not present quantifiable results for this case. Instead, they only compare the results to a Gaussian mixture not regarding transformations. Error rates are only given for a set of synthetic 9×9 images in comparison to Gaussian mixtures. The emphasis of their work is on the automatic training of object models as opposed to classification of images.

Special attention to the subject of occlusion is paid in [Mardia & Qian⁺ 97], but in this work mainly object contours are considered for recognition, not the objects themselves. Some considerations with respect to a statistical model for multiple images can also be found in [Pösl 98]. Here, the author concentrates on determining the unknown 3D transformation parameters in the recognition process as well as improving feature extraction. It is shown that localization can be improved by explicit modeling of the background, although no global optimization of the posterior probability (8.2) is performed in the experiments. This approach has been extended in [Reinhold & Paulus⁺ 01]. Note that the framework presented here does not impose any restriction on the type of feature extraction used. This assures extensibility to multi-modal datasets, where aligned images can be treated as one image with an extended feature vector per pixel [Rybach & Keysers⁺ 05]. Furthermore, the extension to multi-dimensional images is straightforward but the search space that needs to be considered grows rapidly with the number of dimensions. However, the resulting difficulties may be handled by using efficient search strategies.

[Fergus & Perona⁺ 03] use a statistical model for the appearance of object parts that takes into account the relative position of these parts and achieves high recognition performance for object detection. [Tu & Chen⁺ 03] combine top-down and bottom-up processes for image analysis in a statistical context specialized on text and faces in complex images and impressive exemplary results on a few images are demonstrated. [Sullivan & Blake⁺ 01] also emphasize the necessity of considering the background in object localization and address the problem of efficient search. [Leonardis & Bischof 00] use an appearance-based hypothesize-and-test paradigm that admits noise and occlusion but only few tests regarding varying background are performed. Very recently [LeCun & Bottou⁺ 04] introduced a new large dataset for generic object recognition and reported high accuracies using convolutional artificial neural networks for recognition.

With respect to the necessity of integrating segmentation and recognition [Leibe & Schiele 03] write: “Historically, figure-ground segmentation has been seen as an important and even necessary precursor for object recognition. In that context, segmentation is mostly defined as a data driven, that is bottom-up, process. As for humans object recognition and segmentation are heavily intertwined processes, it has been argued that top-down knowledge from object recognition can and should be used for guiding the segmentation process.” Their approach uses a patch-based representation of the objects with the integration of position information, which yields impressive results on different tasks, especially when a detection of the position of the object is wanted. However, their method relies on the availability of hand-segmented data, from which the algorithm learns the segmentation of previously unseen images. Furthermore, their approach has problems when the objects contain holes, like e.g. bicycles.

We believe that there are also some important lessons for image analysis research that can be learned from the experience in the field of automatic speech recognition, which is better understood than image analysis at the moment, due to several reasons.

- The task of automatic speech recognition is well-defined, consisting of the conversion of an audio-signal to a sequence of characters. In comparison, there are many possible tasks within the analysis of images, which leads to many different solutions that sometimes are not systematic approaches but still lead to good results.
- Holistic analysis is computationally more expensive operating on two-dimensional input data (images). For example, matching techniques that are computationally feasible for one-dimensional input can be infeasible for two-dimensional input (cp. Section 6.4).
- The holistic approach is state-of-the-art in automatic speech recognition.

8.2 Statistical model and training

To classify an observation $X \in \mathbb{R}^{I \times J}$ we use Bayes' decision rule as introduced in Chapter 4

$$X \mapsto r(X) = \operatorname{argmax}_k \{p(k) p(X|k)\}, \quad (8.1)$$

where $p(k)$ is the prior probability of class k and $p(X|k)$ is the class-conditional probability for the observation X given class k .

For holistic recognition, we extend the elementary decision rule (8.1) into the following directions:

- We assume that the scene X contains an unknown number M of objects belonging to the classes $k_1, \dots, k_M =: k_1^M$. Reference models $p(X|\mu_k)$ exist for each of the classes $k = 1, \dots, K$, and μ_0 represents the background.
- We take decisions about object boundaries, i.e. the original scene is implicitly partitioned into $M + 1$ regions I_0^M , where $I_m \subset \{(i, j) : i = 1, \dots, I, j = 1, \dots, J\}$ is assumed to contain the m -th object and I_0 the background.
- The reference models may be subject to certain transformations (rotation, scale, translation, etc.). That is, given transformation parameters ϑ_1^M , the m -th reference is mapped to $\mu_{k_m} \rightarrow \mu_{k_m}(\vartheta_m)$. These transformations can be modeled in the class-conditional probabilities as introduced in Chapters 5 and 6.

The unknown parameters M, k_1^M, ϑ_1^M and (implicitly) I_0^M must be considered and the hypothesis which best explains the given scene is searched. This must be done considering the interdependence between the image partitioning, transformation parameters and hypothesized objects, where in the holistic concept partitioning is a part of the classification process. Note that this means that any pixel in the scene must be assigned either to an object or to the background class. This model has been introduced in [Keysers & Dahmen⁺ 03], where a restricted version was used in the experiments, only allowing horizontal shift. The resulting decision rule is

$$r(X) = \operatorname{argmax}_{M, k_1^M} \left\{ \max_{\vartheta_1^M} \left\{ p(\vartheta_1^M) \cdot p(k_1^M) \cdot \prod_{m=0}^M p(X_{I_m} | \mu_{k_m}(\vartheta_m)) \right\} \right\}, \quad (8.2)$$

where X denotes the scene to be classified and X_{I_m} is the feature vector extracted from region I_m . Instead of performing a summation over the parameters ϑ_1^M , we apply the common maximum approximation here. Invariance aspects can be directly incorporated into the models chosen for the density functions using a probabilistic model of variability. In (8.2), $p(k_1^M)$ is a prior over the combination of objects in the scene, which may depend on the transformation parameters and the combination of objects. If the regions I_m are allowed to be non-contiguous, the model can also describe part-based classification schemes [Ullman & Vidal-Naquet⁺ 02, Fergus & Perona⁺ 03] (cp. Chapter 7).

The maximizations involved in the decision process must be carefully implemented in a concrete computer algorithm, as the search space of all possible parameters may be large. Special care must also be taken in the implementation when the regions containing the objects do not have a simple rectangular form. For this case there are at least two possible solutions which both use special assumptions. We use an approach similar to the one used by [Frey & Jojic 03] that masks additional pixels within an object region as belonging to the background, thus allowing the object region to be rectangular. This approach is illustrated in Figure 8.5. Another method is to decompose the foreground/background decision into local decisions [Reinhold & Paulus⁺ 01].

In most practical cases expression (8.2) will be greatly simplified for the task at hand, which we illustrate for the case of classifying images with one object and variable background in the following. Note that only recently object recognition tasks that do not assume known or static background have been successfully addressed. We start to apply our approach to the consideration of the interdependence between segmentation and recognition. For the identification of one object in the presence of an inhomogeneous background we assume $M = 1$ and further that the image partitioning is determined by the transformation of the object reference, i.e. there is no occlusion. Note that the chosen approach means that any pixel in the scene must be assigned either to an object or to the background class. Furthermore, we assume that the transformation probability does not depend on the hypothesized class. Thus, (8.2) reduces to

$$r(X) = \operatorname{argmax}_k \left\{ \max_{\vartheta} \{p(\vartheta) p(k) p(X_{I_0}|\mu_0) p(X_{I_1}|\mu_k(\vartheta))\} \right\}. \quad (8.3)$$

Thus we arrive at a much simpler model for the classification of images with single objects and cluttered background in which we have made all the assumptions explicit. Note that there is a strong difference to two-step procedures that first try to segment the image and then classify the segmented region, which cannot recover from errors made in the segmentation step. Using the decision rule (8.3) means to consider the interdependence between segmentation and classification in one process.

We consider 2D-rotation, isotropic scaling, and translation as transformations. The priors $p(\vartheta)$ and $p(k)$ are assumed uniform. The object density $p(X|\mu_k)$ is modeled using Gaussian kernel densities or Gaussian mixture densities. The use of mixture models allows the implicit modeling of further transformations by mapping them to different densities if they are observed in the training data. The part of the image that is not assigned to any object is assigned to the class background. In the experiments, the set of background pixels is modeled by a univariate distribution on the pixel level, where individual pixel values are assumed to be statistically independent. I.e. we assume for the background model $p(X|\mu_0) = \prod_{x \in X} p(x|\mu_0)$. The local density $p(x|\mu_0)$ is chosen among univariate Gaussian, uniform distribution, or empirical histograms with different numbers of bins. Note that the

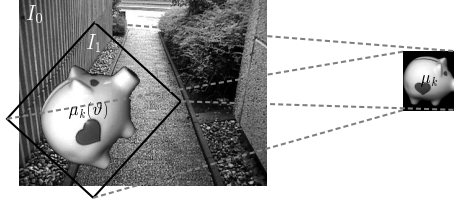


Figure 8.3: Implicit partitioning and comparison during the search.

correct normalization of the distributions is important because of the changing amount of pixels that are explained for different transformation parameters ϑ . One example partitioning is shown in Figure 8.3, two more examples are shown in Figure 8.5.

To illustrate the search or decision problem arising from the decision rule (8.3), we fix the hypothesized class k and assume the maximizing transformation parameters $\hat{\vartheta}$ are to be determined. E.g. considering Gaussian densities $p(X|\mu_k) = N(X|\mu_k, \sigma_1^2 I)$ for the objects and $p(x|\mu_0) = N(x|\mu_0, \sigma_0^2)$ for the background leads to the search

$$\begin{aligned} \hat{\vartheta} &= \operatorname{argmax}_{\vartheta} \{p(\vartheta) p(k) p(X_{I_0}|\mu_0) p(X_{I_1}|\mu_k(\vartheta))\} \\ &= \operatorname{argmin}_{\vartheta} \left\{ -\log p(\vartheta) - \log p(k) + \frac{1}{2}|I_0| \log(2\pi\sigma_0^2) + \frac{1}{2\sigma_0^2} \sum_{x \in X_{I_0}} (x - \mu_0)^2 \right. \\ &\quad \left. + \frac{1}{2}|S_1| \log(2\pi\sigma_1^2) + \frac{1}{2\sigma_1^2} \|X_{I_1} - \mu_k(\vartheta)\|^2 \right\} \end{aligned} \quad (8.4)$$

The maximization in the decision rule is then computationally very demanding and needs to be done for each hypothesized object reference, which may be many in a mixture density. Consider rotation, scaling, and translation as parameters and a suitable quantization of these parameters, then this may result in 72 (rotation) $\times 10$ (scaling) $\times 100 \cdot 100$ (translation) $\times 1,000$ (references) $= 7,200,000,000$ hypotheses to be tested for each scene. The large number of parameter settings ϑ makes the search for the maximizing arguments a complex problem. Therefore, optimization strategies should be considered, including:

- The squared Euclidean distances $\|X_{I_1} - \mu_k(\vartheta)\|^2$ for all translations can be efficiently calculated using the fast Fourier transform. To do so, we decompose the term into $\|X_{I_1} - \mu_k(\vartheta)\|^2 = \|X_{I_1}\|^2 - 2X_{I_1}\mu_k(\vartheta) + \|\mu_k(\vartheta)\|^2$. Then the first and third term can be efficiently computed using sums of squares and the second term can be viewed as a convolution, which can be computed efficiently using the convolution theorem for all translations. This procedure reduces the computation effort for this term in the order of $\log |I_0 \cup I_1| / |I_1|$.
- The sums of squares $\sum_{x \in X_{I_0}} (x - \mu_0)^2$ for all translations can be efficiently computed using precomputed sums of squares (or integral images). I.e. we compute $\sum_{x \in \{1, \dots, i\} \times \{1, \dots, j\}} (x - \mu_0)^2$ for all i, j once and then determine the desired sum by a few additions and subtractions of the (saved) results. This procedure reduces the effort for this term in the order of $1 / |I_1|$.
- The search space can be reduced by limiting the number of hypothesized transformations or by restricting the regions I_1 to square regions.

- A significant speedup can be gained by hierarchically pruning the search space using the results of a complete search in a down-scaled version of the scene.

Algorithms for single object recognition cannot be used to determine the model parameters without given segmentation. The following training algorithm is based on an expectation-maximization (EM) scheme, where the hidden variables are the parameters ϑ for each object in each training scene:

1. initialize model parameters
2. search maximizing transformation parameters ϑ in each scene using (8.4)
3. re-estimate model parameters (e.g. EM algorithm for mixtures)
4. repeat from 2 until convergence

For the training we assume exactly one object to be present in each image. Furthermore, objects are assumed to lie within a square region. The initial model parameters can be based on a constant graylevel estimated from a histogram of the training data or a small set of manually segmented objects. The latter approach facilitates convergence and still leads to a high reduction of manual preprocessing. The hypothesized transformations are translation, isotropic scaling, and 2D-rotation.

The presented probability model contains the transformation parameters and region partitioning as so-called hidden variables as they are not directly observable in the measurement X . This observation immediately leads to an estimation procedure that can be used for training of the chosen model distributions given a set of training images. The expectation-maximization algorithm [Dempster & Laird⁺ 77] iterates the following two steps: determine the best values for the hidden variables on the training data given the current models (expectation); update the model parameters assuming the values previously determined are correct (maximization).

A special case of the former model results if we can assume a uniform or static background for the images. In this case the holistic recognition reduces to the adequate (probabilistic) modeling of the foreground density $p(X|c, \theta)$ in presence of image variability. As discussed before, this approach is generally called ‘appearance-based’ as the appearance X of the image is considered directly [Beymer & Poggio 96].

8.3 Experiments and results


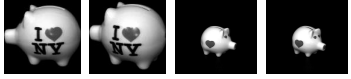
In the following we discuss the results obtained using the holistic model for the two domains of object recognition and medical image categorization.

8.3.1 Results for object learning and recognition

In this section we present results obtained using the holistic model on the data set created on the basis of the COIL-20 data [Keysers & Motter⁺ 03] and a few results for similar tasks.

For the recognition of general objects there is a well-known set of images of 20 objects named Columbia University Object Image Library [Murase & Nayar 95] (COIL-20). These images have been used by many researchers and the objects can be recognized well using different approaches. Unfortunately, different setups for training and test data are used.

Table 8.1: Error rates for the COIL-20 and ERLANGEN data sets.

name	ERLANGEN	COIL-20
# classes	5	20
# training images	90	720
# test images	85	180
example images		
other methods (ER [%])	[Reinhold & Paulus ⁺ 01] 0.0	[Keysers & Dahmen ⁺ 01b] 0.0
holistic model (ER [%])	0.0	0.0

We used a comparatively difficult setup with different lighting conditions, 3D-views, and image resolutions in training and test and the holistic model correctly classified all test images. On the original COIL-20 database, the holistic approach achieves a 0% error rate without further tuning than using a Gaussian background model with mean zero and low variance. This result seems not surprising, as the images are shown on a homogeneous black background. But as the training and test images are shown in different lighting conditions and on different scales, a nearest neighbor classifier is not sufficient for completely correct classification and it is necessary to extend it with elaborate techniques to achieve a 0% error rate [Keysers & Dahmen⁺ 01b]. Another set of objects is used in a database of images created at the University of Erlangen-Nürnberg, which can be recognized at 0% error using a statistical approach [Reinhold & Paulus⁺ 01]. We used the test set with heterogeneous background from the first database and the corresponding training set. The same error rate of 0% was achieved using the proposed holistic model with rectangular prototype models. Tables 8.1 and 8.2 show a summary of results for the two mentioned tasks. The error rates of 0% suggest that these tasks are not representatives of a general object recognition problem. We therefore created different classification tasks based on the COIL-20 data and used these to evaluate the classifier further [Keysers & Motter⁺ 03]. In summary, the holistic analysis achieves good error rates for a wide range of transformations and homogeneous background. If a heterogeneous background is combined with many transformations, the recognition rate is still very high.

To investigate the effect of different background models, we tested univariate single Gaussian densities, uniform distributions, and histograms with varying numbers of bins. In about 70% of the evaluated experiments, the univariate single Gaussian densities performed best among these models [Keysers & Motter⁺ 03]. In the following we therefore only discuss results obtained with this background model to separate its effect from the other parameters.

To observe the effect of known transformation parameters on the proposed training, we trained a Gaussian single density on all images with a fixed 3D-rotation angle of COIL-RWTH-2. The resulting mean images are shown in Table 8.3. It can be observed that the algorithm finds visually important parts of the object searched for. Note that this occurs although for the algorithm the only knowledge about the data is that all images contain an instance of the same object. The exact appearance of the mean image differs strongly depending on the information supplied to the training algorithm. To evaluate the proposed training algorithm further, we trained Gaussian mixture densities on COIL-RWTH-1 and used these models to classify the original COIL-20 dataset. This resulted in 7.8% error rate. Note that the mixture density now models the different 3D-rotation angles of the objects. If the correct 2D-rotation of the object is supplied to the training algorithm, this error rate

Table 8.2: Error rates for the COIL-RWTH database (20 classes, 180 test images each).


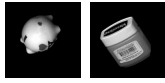



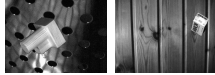

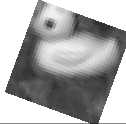

training data	COIL-20 720 images 	COIL-RWTH-1 5760 images 	COIL-RWTH-2 5760 images 
test data	COIL-RWTH-1 	COIL-RWTH-1 	COIL-RWTH-2 
kernel dens. (ER[%])	38.9	27.2	95.0
holistic (ER[%])	1.1	7.8	92.8

Table 8.3: Training results for Gaussian single densities on COIL-RWTH-2 with fixed 3D-rotation angle shown for one of the objects. The means were initialized with a constant gray value.

known	rotation	scaling	—
unknown	scaling, translation	rotation, translation	rotation, scaling, translation
result			

can be reduced to 4.4%. To separate the effect of unknown rotation from the other unknown parameters, in the following paragraphs we only present results, in which the 2D-rotation of the objects in the images is known to the classifier.

We evaluated the classification accuracy of the complete setup on the COIL-RWTH databases in three scenarios. The results are shown in Table 8.2. As no other results are available, we used a conventional kernel density classifier for comparison. This classifier was supplied with the same information and an object position compensation was implemented using the center of gravity of the images. The results show that the holistic model performs with acceptable error rates for homogeneous background. Recall that scale changes are handled automatically and segmentation is performed implicitly in the model. The high error rates of the kernel density classifier can be explained by the fact that it cannot cope with scale changes. This also explains the improving error rate for the COIL-RWTH-1 test data when switching from the COIL-20 to the COIL-RWTH-1 training data, because the latter already includes variations in scale.

The error rates for the inhomogeneous background are clearly unacceptable. The failure of the algorithm here is based on the coincidence of two problems: 1. Automatic object training with unknown segmentation and variable background is very difficult. The resulting mean vectors show strong blur due to the changing background but capture some characteristic information, which is not enough to achieve lower error rates. 2. Detection of objects of variable scale and position in large inhomogeneous images based on an incomplete object model of graylevels and backgrounds not seen in training is possible only in few cases.

A simplified recognition problem is that of object detection, where the task of the system is to decide if a specific object is present in an image or not. For this task different datasets have been assembled at the California Institute of Technology and been used in



Figure 8.4: Training images from the Caltech faces data set and prototypes learned by the holistic model. The first group shows five example images of the 218 training images. The second group shows the 18 images learned as object prototypes within a Gaussian mixture density if the prototype is initialized using the last image as starting prototype. The third group equally shows the 18 images learned as object prototypes within a Gaussian mixture density if the prototype is initialized using a constant grey-value only. Note that in this case the algorithm is capable of learning a visually satisfactory representation of the object class given only the information that a common object is present in all of the training images.

experiments with statistical classifiers [Fergus & Perona⁺ 03]. Interestingly, for the tasks of detecting faces and airplanes better results can be achieved using texture features calculated from the whole image [Deselaers & Keysers⁺ 04a]. This means that again the task is not a representative of a general object recognition problem. Also, very good results are obtained when using a patch-based approach as detailed in Chapter 7. Nevertheless, the learned object prototypes on the face detection task can be used to illustrate the object representations that are learned by the holistic model as depicted in Figure 8.4.

In the following we will briefly discuss experiments that extend the basic holistic modeling approach described above in several directions.

Starting from the error rate of 92.8% as reported in [Keysers & Motter⁺ 03], we observed that some of the learned prototypes models characterize uniformly colored regions of the image background. Inspired by the use of variance thresholds [Paredes & Perez-Cortes⁺ 01] or entropy thresholds [Fergus & Perona⁺ 03] in patch-based classification approaches of images with background clutter, we imposed a variance threshold on the prototypes during the training. That is, in the EM algorithm any density center for which the variance of the pixel values falls below a given threshold is pruned. This procedure reduced the error rate on the COIL-RWTH corpus from 92.8% to 88.3%.

The second improvement was inspired by the use of adaptive background noise modeling in speech recognition. Instead of using a fixed Gaussian background model, we let the background model adapt to the image under consideration. After a first determination of foreground and background region using the holistic model, we estimate the mean and variance of the background using the maximum likelihood estimators. Using this new background model that is adapted to the image under consideration, we apply the holistic

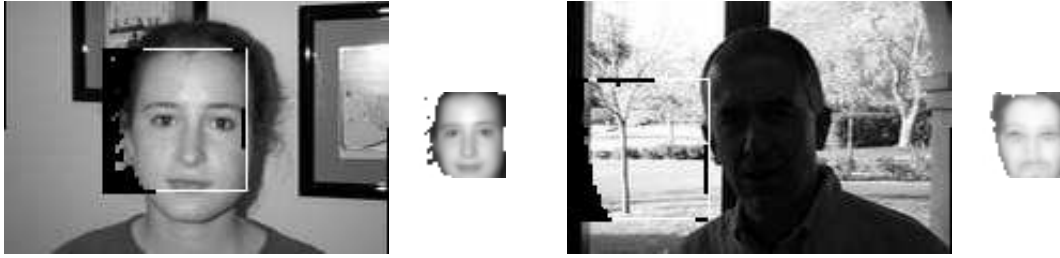


Figure 8.5: Illustration of the implicit detection process on the Caltech faces data set. Each pair shows an input image with a frame showing the position of the detected face followed by the best-fitting prototype within the Gaussian mixture density. The first pair shows a successful detection, while the second pair shows a failure in a case with difficult illumination.



Figure 8.6: Example images and resulting object representations for the data from [Frey & Jojic 03].

decision rule again. This procedure further reduced the error rate from 88.3% to 82.2%.

In separate experiments on the Caltech faces database, we observed a reduction of the error rate from 42% to 16% using the variance threshold for the prototypes as described above. Additionally, we could improve the error rate by allowing non-square prototypes from 16% to 12%. This was done by automatically choosing those 20% of the pixels that are connected to the image border which exhibit the highest variance during training. These pixels were then assumed to belong to the background and not to the foreground. Thus, we can model foreground prototypes of very different shapes. Figure 8.5 shows two results of the implicit detection process in the holistic model using these non-square prototypes. Note that unfortunately these results are still not competitive with other error rates published for this data set as presented in Table 3.13. We were not able to obtain lower error rates using the holistic model. On the other hand, much lower error rates could be obtained using a patch-based approach [Deselaers & Keysers⁺ 05a] as described in Chapter 7.

Figure 8.6 shows example images of the data¹ used by [Frey & Jojic 03] and the resulting learned means using the holistic approach. The images are taken from video sequences with added ‘snow’. In total, 200 images are available and were used to train the mixture model using the holistic approach, resulting in the density centers as shown. No quanti-

¹We would like to thank B. Frey for making available this data set.

Table 8.4: Error rates [%] on the IRMA database using the statistical approach with kernel densities and invariant distance measures. Error rates are determined using the leaving-one-out method.

distance measure	thresholding	
	no	yes
Mahalanobis distance	14.0	11.2
tangent distance	13.3	11.1
image distortion model	12.1	9.0
tangent distance + image distortion model	10.4	8.0

tative results for these data are available, but the results are visually comparable to those presented in [Frey & Jojic 03]. We can observe that the algorithm learns visually satisfactory representations of the image set and correctly captures the faces as the most prominent feature.

8.3.2 Results for the IRMA task

The holistic approach for image analysis has obtained one of the lowest error rates among a variety of methods on the IRMA-1,167 database of medical images [Keysers & Dahmen⁺ 03]. Increased performance has then been obtained by using the appearance-based model of variability leading to an error rate of only 5.3% as described in Chapter 6.

The experimental results on the IRMA database are summarized in Table 8.4 [Keysers & Dahmen⁺ 03]. As the images are scaled to the same height, the possible regions that are hypothesized in (8.2) are restricted to rectangular areas of the same height, which allows efficient maximization. The class-conditional densities are modeled using kernel densities here.

The baseline results were obtained using the Mahalanobis distance, resulting in an error rate of 14.0%. Using the presented IDM with a warp range of 3×3 pixels and no image context features (only the pixel values were used), the error rate was reduced to 12.1%. Although the distortion model is straightforward, it effectively compensates for local variations in radiographs. Using only the tangent distance, the error rate was 13.3%. This gain was not as large as for the distortion model, but still remarkable. In another experiment, it was investigated whether the improvements of tangent distance and the image distortion model are additive. This sounds reasonable, as tangent distance compensates for global image transformations, whereas the image distortion model deals with local perturbations. Indeed, using the distorted tangent distance, the error rate was further reduced to 10.4%. As few large differences in pixel values can mislead classifiers based on squared error distances (e.g. [Vasconcelos & Lippman 98]), a local threshold was introduced to limit the maximum contribution of a single pixel difference to the distance between two images. This is especially justified here, because the images may be subject to artifacts that do not affect class-membership, like noise or changing scribor position in radiographs. Applying this thresholding approach, the error rate was reduced to 8.0%.

To make sure that no overfitting occurred in the experiments, the 332 previously unseen radiographs were used as test images and the 1,617 images of the IRMA database as ref-

erences. Using the optimal parameters for the database determined by the leaving-one-out strategy, the algorithm misclassified 30 of the new radiographs (9.0%) which means that the classifier proposed here generalizes very well.

In the course of this work, various other experiments were carried out, some of which are worth mentioning. For example, several tangents based on other transformations (e.g. projective) were tested in experiments, but no improvement over the combination of affine and brightness transforms was obtained. Furthermore, experiments concerning the creation of virtual data (a method that was very successful for the task of optical character recognition [Keysers & Dahmen⁺ 00b]) did not yield improvements on this particular dataset.

The result of tangent distance using a local threshold is only slightly better than that of Mahalanobis distance (11.1% vs. 11.2%). A possible explanation for this behavior is that using the thresholding approach may mimic the behavior of tangent distance in this particular application, because the subspace projection minimizes the sum of squared pixel differences. It should also be noted that in previous experiments all IRMA images were scaled down to a common size of 32×32 pixels prior to classification [Dahmen & Theiner⁺ 00, Dahmen & Keysers⁺ 01b]. In these experiments, the tangent distance significantly outperformed the Mahalanobis distance (with and without the thresholding approach). Thus, it seems possible that the main effect of tangent distance is the compensation of image shifts (which is now inherent to the classification approach by optimizing over all possible image positions). Interestingly, the background model with independent pixel assignments used in [Reinhold & Paulus⁺ 01] also results in local thresholding and can be interpreted as its probabilistic justification.

8.4 Conclusion

We presented a holistic statistical model for appearance-based training and recognition of objects in complex scenes. Experiments on existing databases show the algorithm to be competitive with other known approaches. A further database with a higher level of difficulty that can be used by other researchers was introduced. The gained results underline the difficulty of training and recognition in the presence of an inhomogeneous background. The fact that the presented method achieves 0% error rates on two databases used in the literature, but fails on a database of images with highly misleading background shows that the databases on which 0% error rates can be reported are by far not representative for the complexity of the general object-based scene analysis problem.

Most improvements to the presented method can be expected from the modeling of local variations of the objects using tangent distance or appropriate distortion models.

Why is this approach in images so much harder than a direct comparison with speech recognition would suggest? One possible reason is that the comparison is not a fair one. Trying to learn a model from a set of images is not the same thing as trying to learn pronunciation models from a set of sentences, because the background is not easily separated from the foreground. The comparison therefore should rather be made to the problem of learning pronunciations from data in which constantly two or more people are talking, but only one person is transcribed. In another way, the problem in image prototype learning is more like the following problem you could think of in speech recognition: You are given a set of sentences with about 10 words each that all contain one common word. Now the task

is to learn an acoustic representation of that word, which might additionally have different pronunciations.

Although the approach has theoretical benefits, the complexity of two-dimensional inputs and strong variability of image material showing the same object seems to suggest that component- or patch-based approaches are of greater usefulness and performance. For the task of face recognition [Heisele & Ho⁺ 01] write with respect to a comparison of algorithms that use the whole face versus a system that uses components that “The component system clearly outperformed both global systems on all tests.” Although this work does not aim at a detailed comparison between these approaches, the results for the patch-based approach as presented in Chapter 7 consistently outperform the holistic approach presented in this chapter on all data sets that were examined.

When comparing the classification results obtained with the method presented in this chapter with the results that can be obtained by using patch-based approaches as discussed in Chapter 7, it becomes clear that the holistic approach in the form presented here is not competitive with current patch-based approaches, despite its theoretical benefits. We feel that this result is caused by the difficulty to incorporate efficient models of variability into the model as presented here, which involves the evaluation of a large number of hypotheses. This problem is currently alleviated by the methods used in the patch-based classification approaches. At the same time, it is difficult for the holistic model to benefit from the explicit modeling of the background information. We have seen that an increase in performance can be obtained by using an adaptive background model. Nevertheless, the main problem seems to be to effectively distinguish background from foreground image parts. We feel that future systems that may learn from larger amounts of images may overcome these problems and possibly incorporate for example an explicit background model into a patch-based recognition system to increase its performance.

We may further conclude that although statistical decision theory is well-founded and explored, practical algorithms for holistic image analysis are only recently being applied. Thus, there is a growing need for international standard tasks that can be used to evaluate and compare these approaches. One immediate application of these techniques is the use for object-based image retrieval from large databases as e.g. in medical applications. Here the conventional approach is to use global features based on color or texture, while object-based retrieval is more appropriate for a narrow domain.

9 Conclusion

Es reicht nicht, keine Gedanken zu haben, man muss auch unfähig sein, sie auszudrücken.

– K. Kraus

In the preceding chapters, we discussed several methods to effectively model the variability that can be found in image data with the goal of reducing the recognition error rate. We based our methods on the appearance-based paradigm that operates on image intensities directly without attempting a segmentation of the input.

We described the tangent distance as a linear model of variability and observed that it can lead to substantial improvements in recognition performance. We saw that a probabilistic treatment of the tangent distance allows us to estimate the transformation derivatives from a training data set. This estimation allows us to employ the approach for data without a priori knowledge of the variability that is present.

We discussed discrete models of image variability that assign pixel positions of a test image to those of a reference image in order to determine which reference best explains the test image. Here, we saw that the computation of a match under complete two-dimensional constraints is an NP-hard problem. We used various models with fewer constraints on the matching and could observe that the pseudo two-dimensional hidden Markov distortion model performed very well. The much simpler image distortion model, which disregards the displacement of neighboring pixels, performed almost equally well while reducing computational cost. In both cases it was observed that it is most important to include a suitable representation of the local image context (using gradients and small patches) to achieve low recognition error rates. The use of the Hungarian algorithm to achieve a more homogeneous match could improve the result of the image distortion model, but only at a much higher computational cost. From the experiments we conclude that the simple image distortion model using local image context should be considered as a baseline method for image distances in various scenarios because it combines ease of implementation, low computational complexity, and very good recognition results.

As an extreme case of a model of variability we discussed the use of local image patches while disregarding the position of these patches. This approach is especially suited for images with occlusion and background clutter. Starting from the baseline method of Paredes and colleagues, we could show that several improvements were possible: the use of kernel densities in the probability model, the use of the tangent distance for the comparison of patches, and the use of multi-scale patch extraction. When using a vector quantization of the patch set, which is equivalent to the use of histograms, we could achieve very good performance by employing discriminative maximum entropy training. This approach allows the computer to determine the importance of a given class of patches for the recognition problem. For example, in the case of human face recognition it was automatically determined that patches containing an eye are the most discriminative. The histogram-based method could be improved by using a brightness normalization directly integrated into the principal component analysis. Patch-based approaches for visual object recognition are a topic of

Table 9.1: Error rates [%] achieved on various data sets using the techniques described in this work in comparison to the best other published error rates.

name	this work	page	best ‘other’	reference
USPS	1.9	127	2.2	[Dong & Krzyzak ⁺ 02b]
MNIST	0.5	127	0.4	[Simard & Steinkraus ⁺ 03]
UCI	0.8	127	1.5	[Kim & Kim ⁺ 02]
MCEDAR	3.3	127	4.6	[Tipping & Bishop 99]
ETL6A	0.5	133	0.5	[Uchida & Sakoe 03a]
IRMA-1,617	5.3	127	9.3	[Paredes & Keysers ⁺ 02]
IRMA-10,000	12.6	133	14.1	[Marée & Geurts ⁺ 05]
RBC	13.5	157	15.3	[Dahmen & Hektor ⁺ 00]
COIL-20	0.0	183	0.0	[Baker & Nayar 96]
ERLANGEN-50-2	0.6	158	4.8	[Reinhold & Paulus ⁺ 01]
CALTECH-A	1.1	168	0.8	[Deselaers & Keysers ⁺ 04b]
CALTECH-F	3.7	168	0.1	[Fussenegger & Opelt ⁺ 04]
CALTECH-M	1.1	168	3.8	[Gao & Vasconcelos 05]
LTI	1.4	140	4.3	[Pelkmann 99]

intensive research at the time of writing this document and we can expect more interesting results from this area of research.

We presented a discussion of a holistic model for image analysis that is directly derived from Bayes’ decision rule and takes into account the complete image during the classification process. We were able to achieve state-of-the-art results for different object recognition tasks using the holistic model. The corresponding training procedure was able to recover a visual representation of a given object (e.g. a human face) from a set of images containing the object in front of an inhomogeneous background with the only knowledge being that each image contains one of the objects. However, the computational demand of this technique is very high and the use of patch-based approaches seems a more promising approach.

In the course of discussing the discriminative maximum entropy framework, we were able to analyze the relationship between the resulting models and Gaussian densities. This relationship allowed us to discriminatively estimate the equivalent of full covariance matrices for an image recognition task, which is problematic using other methods. Furthermore, the relationship enabled us to derive a new linear feature reduction technique: maximum entropy linear discriminant analysis, which showed superior performance in comparison to conventional linear discriminant analysis when the number of features in relation to the number of classes was high.

In the course of the experiments using the different techniques, we used a variety of data sets and could obtain very good results for some of these, mostly achieving or even advancing the state-of-the-art results. An overview of the best results obtained in comparison to other published error rates is presented in Table 9.1. Due to the use of several data sets, some parts of this work could be regarded as case studies. However, one important overall result is that we could observe improvements of the appearance-based approach using appropriate models of variability, e.g. tangent distance and deformation models, in all experiments. We observed excellent results for handwritten character recognition and a high generalization ability across tasks without parameter tuning. Furthermore, improvements for other image data like medical images and image sequences were obtained. In the recent 2005 PASCAL evaluation and the 2004 and 2005 ImageCLEF evaluations, excellent results in comparison to other approaches were obtained.

List of Acronyms

2DW	2-Dimensional Warping
ANN	Artificial Neural Network
AUC	Area Under (ROC) Curve
CBIR	Content-Based Image Retrieval
CC	Classifier Combination
CEDAR	Center of Excellence for Document Analysis and Recognition (handwritten digits) database
CENPARMI	Centre for Pattern Recognition and Machine Intelligence
CLEF	Cross Language Evaluation Forum
CNN	Convolutional Neural Network
COIL	Columbia University Object Image Library
DCT	Discrete Cosine Transformation
DER	Digit Error Rate
DFT	Discrete Fourier Transformation
DICOM	Digital Imaging and Communications in Medicine
DNF	Disjunctive Normal Form
DP	Dynamic Programming
ECR	Equal Classification Rate
EER	Equal Error Rate
ER	Error Rate
ETL	Electrotechnical Laboratory (character) database
FFT	Fast Fourier Transform
FIRE	Flexible Image Retrieval System
GIFT	GNU Image Finding Toolkit
GIS	Generalized Iterative Scaling
GMD	Gaussian Mixture Density
EM	Expectation-Maximization
HDM	Hungarian Distortion Model
HMM	Hidden Markov Model
i6	Chair of Computer Science VI (Lehrstuhl für Informatik VI) of RWTH Aachen University
IDM	Image Distortion Model
IDIAP	Institut Dalle Molle d'Intelligence Artificielle Perceptive
INRIA	Institut National de Recherche en Informatique
ITI	Instituto Tecnológico de Informática, Universidad Politécnica de Valencia, Spain
IRMA	Image Retrieval in Medical Applications
KD	Kernel Density
L1O	Leaving-One-Out
LBG	Linde-Buzo-Gray Clustering

IRMA	Image Retrieval in Medical Applications
LDA	Linear Discriminant Analysis
LMB	Lehrstuhl für Mustererkennung und Bildverarbeitung of the University of Freiburg
LTI	Lehrstuhl für Technische Informatik of RWTH Aachen University
MCEDAR	Modified CEDAR digit recognition task
ME	Maximum Entropy
MELDA	Maximum Entropy Linear Discriminant Analysis
MI	Institute of Medical Informatics of RWTH Aachen University
ML	Maximum Likelihood
MMI	Maximum Mutual Information
MNIST	Modified National Institute of Standards and Technology (handwritten digits) database
MPI	Max-Planck-Institut Tübingen
NN	Nearest Neighbor
OCR	Optical Character Recognition
ORL	Olivetti Research Laboratory face database
P2DHMM	Pseudo-2-Dimensional Hidden Markov Model
P2DHMDM	Pseudo-2-Dimensional Hidden Markov Distortion Model
PASCAL	Pattern Analysis, Statistical Modelling and Computational Learning
PCA	Principal Components Analysis
PDA	Penalized Discriminant Analysis
POI	Probability of Improvement; Point of Interest
RBC	Red Blood Cell
RBF	Radial Basis Function
ROC	Receiver Operating Characteristic
RST	Rotation, Translation, Scaling (Invariance)
RWTH	Rheinisch-Westfälische Technische Hochschule Aachen (RWTH Aachen University)
SAT	Satisfiability
SIMBA	Search Images by Appearance
SVM	Support Vector Machine
TD	Tangent Distance
TREC	Text REtrieval Conference
UCI	University of California, Irvine (handwritten digits) database
USPS	US Postal Service (handwritten digits) database
VOC	Visual Object Classes
VTs	Virtual Test Sample Method
WER	Word Error Rate

List of Tables

3.1	Corpus and image sizes and example images.	7
3.2	USPS error rates	10
3.3	MNIST error rates	13
3.4	Significance of improvements for best MNIST classifiers	17
3.5	Results for the UCI task, error rates [%].	18
3.6	Error rates [%] for the MCEDAR data set.	19
3.7	Error rates [%] for the CEDAR ‘goodbs’ data set.	19
3.8	Results for ETL6A, error rates [%].	20
3.9	IRMA-1,617 error rates	23
3.10	Overview of ImageCLEF 2005 results	26
3.11	Results for the red blood cells corpus.	28
3.12	Error rates for the Erlangen task	31
3.13	Error rates for the Caltech task	32
4.1	USPS error rates for penalized discriminant analysis	47
5.1	USPS results using Gaussian models	71
5.2	Effect of the type of tangent vectors on the USPS error rate	73
5.3	USPS results, kernel densities and tangent distance	73
5.4	Summary of Results for the RBC data	74
5.5	Overview of maximum entropy results on USPS	82
5.6	Error rates for the maximum entropy method and different features	84
5.7	Corpus statistics for MONK, DNA, and LETTER	85
5.8	Comparison ME, different orders on UCI/STATLOG	86
5.9	Comparison between ME and other approaches	88
5.10	Results of first comparison between LDA and ME	94
5.11	Summary of data statistics for the MELDA experiments	94
5.12	MELDA experimental results	95
6.1	Overview of constraints for the nonlinear deformation models	102
6.2	USPS error rates for IDM with local tangent distance	109
6.3	Effect of beam search for 2DW on USPS	115
6.4	Open questions about matching complexity	124
6.5	USPS error rates using single prototypes	128
6.6	Results for LTI gesture recognition data	140
6.7	Results for the BOSTON50 sign language data	141
6.8	Summary of results for the nonlinear models	142
7.1	Results for multi-scale patch extraction on the IRMA data.	151
7.2	Results for kernel densities in patch-based approach	158

7.3	Patch-based results on Erlangen data	159
7.4	Error rates, patch-histograms, 61×61	164
7.5	Error rates on scaled Caltech data	165
7.6	IRMA error rates for the patch-histograms	166
7.7	Results for the extensions to the histogram-based approach	168
7.8	Results of PASCAL challenge (Task 1)	170
7.9	Results of PASCAL challenge (Task 2)	171
8.1	Error rates for the COIL-20 and ERLANGEN data sets.	183
8.2	Error rates for the COIL-RWTH database	184
8.3	Training results for COIL-RWTH-2	184
8.4	Results on the IRMA task using the holistic model	187
9.1	Best error rates on various data sets	192

List of Figures

1.1	Examples of image variability	3
3.1	Example images from the USPS data set	9
3.2	Difficult USPS test samples	9
3.3	USPS 1-NN recognition examples	11
3.4	Example images from the MNIST data set	12
3.5	Difficult MNIST test samples	14
3.6	One example of each class of the ETL6A data.	20
3.7	One image from each of the six IRMA-1,617 classes.	21
3.8	Several images from class ‘chest’ from the IRMA-1,617 database.	21
3.9	IRMA 1-NN recognition examples	24
3.10	Examples of red blood cell images	27
3.11	Examples of the COIL-20 database	29
3.12	Example images from the COIL-i6 data	30
3.13	Examples from the Erlangen corpus	31
3.14	Examples from the Caltech task	32
4.1	Structure of a recognition system	37
4.2	Illustration of conditional densities	39
4.3	PDA error rates on USPS	47
4.4	Penalized LDA projection vectors	47
4.5	Illustration of the need for invariant distance measures	50
4.6	Examples of 2D smoothed random distortions	56
5.1	Examples, use of tangent vectors for handwritten characters	63
5.2	Examples, use of tangent vectors for medical data	63
5.3	Illustration of the tangent distance	64
5.4	Example images for closest subspace point	64
5.5	Density for tangents on observation side	66
5.6	USPS error rates with tangent vectors	72
5.7	Use of tangent vectors for continuous text recognition	76
5.8	Examples of posterior from the GIS algorithm	78
5.9	Examples of posterior for ML and MMI training	79
5.10	Visualization of maximum entropy coefficients	82
5.11	Eigenvalue distribution for the maximum entropy approach	83
5.12	Comparison of second- and third-order features for ME on USPS	84
5.13	Convergence of heuristic speed-up GIS	91
5.14	Relative performance MELDA / LDA	96
6.1	2-dimensional interdependence of local displacements.	103
6.2	Examples of nonlinear matching applied to face images	104
6.3	USPS P2DHMM error rate with respect to gradient weight	106
6.4	Local context extraction, 3×3 sub images of gradients	107
6.5	Results for PCA context extraction on USPS	108

6.6	Padding to relax image border constraints	110
6.7	Example of up-scaling using spline and bilinear interpolation	111
6.8	2DW constraints for neighboring pixels	112
6.9	Illustration of the warped wake concept in 2DW	113
6.10	Illustration of 2DW mapping restrictions	113
6.11	2DW dynamic programming algorithm	114
6.12	Examples, 2DW compared to Euclidean distance	116
6.13	Straight connector component	119
6.14	Corner connector component	120
6.15	Variable component	120
6.16	Clause component	122
6.17	Crossing connector component	122
6.18	Schematic view of a pseudo 2-D HMM	126
6.19	USPS prototypes using matching	128
6.20	Image retrieval for medical images using FIRE	134
6.21	Construction of the bipartite graph for Hungarian matching	135
6.22	Examples of pixel displacement for IDM and HDM	138
6.23	HDM error rates on USPS	139
6.24	Example images from the LTI gesture database	140
6.25	Example images from the BOSTON50 sign language data	141
6.26	Summary of the IDM distance computation	142
7.1	Local patch examples, USPS database	146
7.2	Local patch examples, IRMA database	146
7.3	Schematic view of the basic patch-based approach	148
7.4	Patch extraction: salient points and uniform grid	150
7.5	Extraction of multi-scale patches	151
7.6	Examples of PCA for brightness tolerance	152
7.7	PCA and DCT vectors	152
7.8	Impact of approximate nearest neighbor search.	156
7.9	Examples of changing line thickness for red blood cells.	157
7.10	Discrete and smoothed patch histograms	163
7.11	Patch size influence for patch-based discriminative model	163
7.12	Most discriminative image patches on unscaled Caltech data	164
7.13	Patch size influence for patch-based discriminative model	165
7.14	Most discriminative patches for Caltech data	166
7.15	Most discriminative patches for faces	166
7.16	Classification examples, patch-based, Caltech data	167
7.17	Influence of histogram smoothing	168
7.18	Influence of the number of clusters	168
7.19	Misclassifications on USPS suggesting the use of classifier combination	172
8.1	Example image, holistic recognition	176
8.2	Interdependence in recognition	177
8.3	Implicit partitioning and comparison during the search	181
8.4	Illustration of the holistic model results for the Caltech faces data	185
8.5	Illustration of implicit holistic detection process	186
8.6	Examples and results for the data from [Frey & Jojic 03]	186

References

- [Abend & Harley⁺ 65] K. Abend, T. Harley, L. Kanal: Classification of Binary Random Patterns. *IEEE Trans. Information Theory*, Vol. 11, pp. 538–544, 1965.
- [Agazzi & Kuo 93] O. Agazzi, S. Kuo: Pseudo Two-Dimensional Hidden Markov Models for Document Recognition. *AT&T Technical J.*, Vol. 72, No. 5, pp. 60–72, Sept. 1993.
- [Akyol & Canzler⁺ 00] S. Akyol, U. Canzler, K. Bengler, W. Hahn: Gesture Control for Use in Automobiles. In *IAPR Workshop on Machine Vision Applications*, pp. 349–352, Tokyo, Japan, Nov. 2000.
- [Alpaydi & Kaynak 98] E. Alpaydi, C. Kaynak: Cascading Classifiers. *Kybernetika*, Vol. 4, pp. 369–374, 1998.
- [Arya & Mount⁺ 98] S. Arya, D.M. Mount, N.S. Netanyahu, R. Silverman, A.Y. Wu: An Optimal Algorithm for Approximate Nearest Neighbor Searching. *J. ACM*, Vol. 45, No. 6, pp. 891–923, Nov. 1998.
- [Athistos & Alon⁺ 05] V. Athistos, J. Alon, S. Sclaroff: Efficient Nearest Neighbor Classification Using a Cascade of Approximate Similarity Measures. In *CVPR 2005, Int. Conf. on Computer Vision and Pattern Recognition*, Vol. I, pp. 486–493, San Diego, CA, June 2005.
- [Baker & Nayar 96] S. Baker, S. Nayar: Pattern Rejection. In *CVPR 96, Int. Conf. on Computer Vision and Pattern Recognition*, pp. 544–549, San Francisco, CA, June 1996.
- [Belhumeur & Hespanha⁺ 97] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman: Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 711–720, July 1997.
- [Belongie & Malik⁺ 01] S. Belongie, J. Malik, J. Puzicha: Shape Context: A New Descriptor for Shape Matching and Object Recognition. In T.K. Leen, T.G. Dietterich, V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pp. 831–837. MIT Press, April 2001.
- [Belongie & Malik⁺ 02] S. Belongie, J. Malik, J. Puzicha: Shape Matching and Object Recognition Using Shape Contexts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 24, No. 4, pp. 509–522, April 2002.
- [Berger & Della Pietra⁺ 96] A.L. Berger, S.A. Della Pietra, V.J. Della Pietra: A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, Vol. 22, No. 1, pp. 39–72, March 1996.
- [Bernado-Mansilla & Ho 04] E. Bernado-Mansilla, T.K. Ho: On Classifier Domains of Competence. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. I, pp. 136–139, Cambridge, UK, Aug. 2004.
- [Beymer & Poggio 96] D. Beymer, T. Poggio: Image Representations for Visual Learning. *Science*, Vol. 272, No. 5270, pp. 1905–1909, June 1996.
- [Bisani & Ney 04] M. Bisani, H. Ney: Bootstrap Estimates for Confidence Intervals in ASR Performance Evaluation. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vol. 1, pp. 409–412, Montreal, Canada, May 2004.

- [Bishop & Winn 00] C.M. Bishop, J.M. Winn: Non-linear Bayesian Image Modelling. In *Proc. 6th European Conf. on Computer Vision*, Vol. LNCS 1842, pp. 3–17, Dublin, Ireland, June 2000.
- [Bishop 95] C.M. Bishop: *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- [Bishop 99] C.M. Bishop: Bayesian PCA. In M. Kearns, S. Solla, D. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pp. 332–388. MIT Press, 1999.
- [Bottou & Cortes⁺ 94] L. Bottou, C. Cortes, J.S. Denker, H. Drucker, I. Guyon, L. Jackel, Y. Le Cun, U. Müller, E. Säckinger, P. Simard, V.N. Vapnik: Comparison of Classifier Methods: A Case Study in Handwritten Digit Recognition. In *Proc. of the Int. Conf. on Pattern Recognition*, pp. 77–82, Jerusalem, Israel, Oct. 1994.
- [Bredno & Brandt⁺ 00] J. Bredno, S. Brandt, J. Dahmen, B. Wein, T. Lehmann: Kategorisierung von Röntgenbildern mit aktiven Konturmodellen. In *Bildverarbeitung in der Medizin, München*, pp. 356–360, March 2000.
- [Bregler & Omohundro 95] C. Bregler, S.M. Omohundro: Nonlinear Image Interpolation using Manifold Learning. In G. Tesauro, D. Touretzky, T. Leen, editors, *Advances in Neural Information Processing Systems 7*, pp. 973–980. MIT Press, July 1995.
- [Breuel 93] T. Breuel: Recognition of Handprinted Digits using Optimal Bounded Error Matching. In *Proc. 2nd Int. Conf. on Document Analysis and Recognition*, pp. 493–496, Tsukuba City, Japan, Oct. 1993.
- [Brown & Lowe 02] M. Brown, D. Lowe: Invariant Features from Interest Point Groups. In *British Machine Vision Conf.*, pp. 656–665, Cardiff, Wales, UK, Sept. 2002.
- [Burkhardt & Fenske⁺ 92] H. Burkhardt, A. Fenske, H. Schulz-Mirbach: Invariants for the Recognition of Planar Contour and Gray-Scale Images. In *Invariants for Recognition, ESPRIT Basic Research Workshop at ECCV 92*, pp. 1–26, Santa Margherita Ligure, Italy, May 1992.
- [Burkhardt & Siggelkow 01] H. Burkhardt, S. Siggelkow. Invariant Features in Pattern Recognition – Fundamentals and Applications. In C. Kotropoulos, I. Pitas, editors, *Nonlinear Model-Based Image/Video Processing and Analysis*, chapter 7, pp. 269–307. Wiley, 2001.
- [Burr 81] D.J. Burr: Elastic Matching of Line Drawings. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 3, No. 6, pp. 708–713, Nov. 1981.
- [Cai & Liu 99] J. Cai, Z.Q. Liu: Integration of Structural and Statistical Information for Unconstrained Handwritten Numeral Recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 21, No. 3, pp. 263–270, March 1999.
- [Dahmen & Beulen⁺ 00] J. Dahmen, K. Beulen, M.O. Güld, H. Ney: A Mixture Density Based Approach to Object Recognition for Image Retrieval. In *Proc. of the 6th Int. RIAO Conf. on Content-Based Multimedia Information Access, Paris, France*, pp. 1632–1647, April 2000.
- [Dahmen & Hektor⁺ 00] J. Dahmen, J. Hektor, R. Perrey, H. Ney: Automatic Classification of Red Blood Cells using Gaussian Mixture Densities. In *Bildverarbeitung für die Medizin*, pp. 331–335, Munich, March 2000.
- [Dahmen & Keysers⁺ 00a] J. Dahmen, D. Keysers, M.O. Güld, H. Ney: Invariant Image Object Recognition using Mixture Densities. In *Proc. 15th Int. Conf. on Pattern Recognition*, Vol. 2, pp. 614–617, Barcelona, Spain, Sept. 2000.
- [Dahmen & Keysers⁺ 00b] J. Dahmen, D. Keysers, M. Pitz, H. Ney: Structured Covariance Matrices for Statistical Image Object Recognition. In *22. DAGM Symposium Mustererkennung*, pp. 99–106, Kiel, Germany, Sept. 2000. Springer.

-
- [Dahmen & Keyzers⁺ 01a] J. Dahmen, D. Keyzers, H. Ney: Combined Classification of Handwritten Digits using the 'Virtual Test Sample Method'. In *MCS 2001, 2nd Int. Workshop on Multiple Classifier Systems*, Vol. 2096 of *Lecture Notes in Computer Science*, pp. 109–118, Cambridge, UK, May 2001. Springer.
- [Dahmen & Keyzers⁺ 01b] J. Dahmen, D. Keyzers, H. Ney, M.O. Güld: Statistical Image Object Recognition using Mixture Densities. *J. Mathematical Imaging and Vision*, Vol. 14, No. 3, pp. 285–296, May 2001.
- [Dahmen & Schlüter⁺ 99] J. Dahmen, R. Schlüter, H. Ney: Discriminative Training of Gaussian Mixture Densities for Image Object Recognition. In *21. DAGM Symposium Mustererkennung*, pp. 205–212, Bonn, Germany, Sept. 1999.
- [Dahmen & Theiner⁺ 00] J. Dahmen, T. Theiner, D. Keyzers, H. Ney, T. Lehmann, B. Wein: Classification of Radiographs in the 'Image Retrieval in Medical Applications' System (IRMA). In *Proc. of the 6th Int. RIAO Conf. on Content-Based Multimedia Information Access, Paris, France*, pp. 551–566, April 2000.
- [Dahmen 01] J. Dahmen. *Invariant Image Object Recognition using Gaussian Mixture Densities*. PhD thesis, RWTH Aachen University, Aachen, Germany, Oct. 2001.
- [Darroch & Ratcliff 72] J.N. Darroch, D. Ratcliff: Generalized Iterative Scaling for Log-Linear Models. *Annals of Mathematical Statistics*, Vol. 43, No. 5, pp. 1470–1480, 1972.
- [DeCoste & Schölkopf 02] D. DeCoste, B. Schölkopf: Training Invariant Support Vector Machines. *Machine Learning*, Vol. 46, No. 1-3, pp. 161–190, 2002.
- [Della Pietra & Della Pietra⁺ 97] S. Della Pietra, V. Della Pietra, J. Lafferty: Inducing Features of Random Fields. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 4, pp. 380–393, April 1997.
- [DeMenthon & Doermann⁺ 00] D. DeMenthon, D. Doermann, M.V. Stükelberg: Image Distance Using Hidden Markov Models. In *Proc. 15th Int. Conf. on Pattern Recognition*, Vol. 3, pp. 143–146, Barcelona, Spain, Sept. 2000.
- [Dempster & Laird⁺ 77] A. Dempster, N. Laird, D. Rubin: Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. Royal Statistical Society Series B*, Vol. 39, pp. 1–38, 1977.
- [Deriche & Giraudon 93] R. Deriche, G. Giraudon: A Computational Approach to Corner and Vertex Detection. *Int. J. Computer Vision*, Vol. 10, pp. 101–124, 1993.
- [Deselaers & Keyzers⁺ 03] T. Deselaers, D. Keyzers, R. Paredes, E. Vidal, H. Ney: Local Representations for Multi-Object Recognition. In *DAGM 2003, Pattern Recognition, 25th DAGM Symposium*, Vol. 2781 of *Lecture Notes in Computer Science*, pp. 305–312, Magdeburg, Germany, Sept. 2003. Springer.
- [Deselaers & Keyzers⁺ 04a] T. Deselaers, D. Keyzers, H. Ney: Classification Error Rate for Quantitative Evaluation of Content-based Image Retrieval Systems. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. II, pp. 505–508, Cambridge, UK, Aug. 2004.
- [Deselaers & Keyzers⁺ 04b] T. Deselaers, D. Keyzers, H. Ney: Features for Image Retrieval: A Quantitative Comparison. In *DAGM 2004, Pattern Recognition, 26th DAGM Symposium*, Vol. 3175 of *Lecture Notes in Computer Science*, pp. 228–236, Tübingen, Germany, Aug. 2004.
- [Deselaers & Keyzers⁺ 04c] T. Deselaers, D. Keyzers, H. Ney: FIRE - Flexible Image Retrieval Engine: ImageCLEF 2004 Evaluation. In *CLEF 2004*, Vol. 3491 of *Lecture Notes in Computer Science*, pp. 688–698, Bath, UK, Sept. 2004.

- [Deselaers & Keyzers⁺ 05a] T. Deselaers, D. Keyzers, H. Ney: Discriminative Training for Object Recognition using Image Patches. In *CVPR 2005, Int. Conf. on Computer Vision and Pattern Recognition*, Vol. II, pp. 157–162, San Diego, CA, June 2005.
- [Deselaers & Keyzers⁺ 05b] T. Deselaers, D. Keyzers, H. Ney.: Improving a Discriminative Approach to Object Recognition using Image Patches. In *DAGM 2005, Pattern Recognition, 27th DAGM Symposium*, Vol. LNCS 3663, pp. 326–333, Vienna, Austria, Aug. 2005.
- [Deselaers 03] T. Deselaers. Features for Image Retrieval. Diploma thesis, Lehrstuhl für Informatik VI, RWTH Aachen University, Aachen, Germany, Dec. 2003.
- [Deuticke & Grebe⁺ 90] B. Deuticke, R. Grebe, C. Haest. Action of Drugs on the Erythrocyte Membrane. In J. Harris, editor, *Blood Cell Biochemistry*, Vol. 1, pp. 475–529. Plenum Press, New York, 1990.
- [Devijver & Kittler 82] P.A. Devijver, J.V. Kittler: *Pattern Recognition. A Statistical Approach*. Prentice-Hall, Englewood Cliffs, NJ, 1982.
- [Devijver 86] P. Devijver: Probabilistic Labeling in a Hidden Second Order Markov Mesh. In *Pattern Recognition in Practice II*, pp. 113–123, 1986.
- [di Battista & Eades⁺ 99] G. di Battista, P. Eades, R. Tamassia: *Graph Drawing*. Prentice Hall, Upper Saddle River, NJ, 1999.
- [Dong & Krzyzak⁺ 01] J.X. Dong, A. Krzyzak, C.Y. Suen. Statistical Results of Human Performance on USPS database. Technical report, CENPARMI, Concordia University, Montreal, Canada, Oct. 2001.
- [Dong & Krzyzak⁺ 02a] J.X. Dong, A. Krzyzak, C.Y. Suen: Local learning framework for handwritten character recognition. *Engineering Applications of Artificial Intelligence*, Vol. 15, No. 2, pp. 151–159, April 2002.
- [Dong & Krzyzak⁺ 02b] J.X. Dong, A. Krzyzak, C.Y. Suen: A Practical SMO Algorithm. In *Proc. Int. Conf. on Pattern Recognition*, Vol. 3, Quebec City, Canada, Aug. 2002.
- [Dong & Krzyzak⁺ 05] J.X. Dong, A. Krzyzak, C.Y. Suen: Fast SVM Training Algorithm with Decomposition on Very Large Data Sets. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 27, No. 4, pp. 603–618, April 2005. Additional results at <http://www.cenparmi.concordia.ca/~people/jdong/HeroSvm.html>.
- [Dorko & Schmid 03] G. Dorko, C. Schmid: Selection of Scale-Invariant Parts for Object Class Recognition. In *International Conference on Computer Vision*, Vol. 1, pp. 634–640, Nice, France, Oct. 2003.
- [Dreuw & Keyzers⁺ 05] P. Dreuw, D. Keyzers, T. Deselaers, H. Ney: Gesture Recognition Using Image Comparison Methods. In *GW 2005, 6th Int. Workshop on Gesture in Human-Computer Interaction and Simulation*, Vannes, France, May 2005.
- [Dreuw 05] P. Dreuw. Appearance-Based Gesture Recognition. Diploma thesis, Lehrstuhl für Informatik VI, RWTH Aachen University, Aachen, Jan. 2005.
- [Drucker & Schapire⁺ 93] H. Drucker, R. Schapire, P. Simard: Boosting Performance in Neural Networks. *Int. J. Pattern Recognition Artificial Intelligence*, Vol. 7, No. 4, pp. 705–719, 1993.
- [Duda & Hart 73] R. Duda, P. Hart: *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.
- [Duda & Hart⁺ 01a] R.O. Duda, P.E. Hart, D.G. Stork: *Pattern Classification*. John Wiley & Sons, Inc., New York, 2nd edition, 2001.

-
- [Duda & Hart⁺ 01b] R.O. Duda, P.E. Hart, D.G. Stork: *Pattern Classification*. New York: John Wiley & Sons, 2nd edition, 2001.
- [Eppshtein & Ullman 05] B. Eppshtein, S. Ullman: Identifying Semantically Equivalent Object Fragments. In C. Schmid, S. Soatto, C. Tomasi, editors, *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 2–9, San Diego, CA, USA, June 2005. IEEE.
- [Everingham & Gool⁺ 05] M. Everingham, L.V. Gool, C. Williams, A. Zisserman. Pascal Visual Object Classes Challenge Results, April 2005. Available at http://www.pascal-network.org/challenges/VOC/voc/results_050405.pdf, accessed July 2005.
- [Everson & Roberts 00] R. Everson, S. Roberts: Inferring the eigenvalues of covariance matrices from limited, noisy data. *IEEE Trans. Signal Processing*, Vol. 48, No. 7, pp. 2083–2091, July 2000.
- [Fergus & Perona⁺ 03] R. Fergus, P. Perona, A. Zisserman: Object Class Recognition by Unsupervised Scale-Invariant Learning. In *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pp. 264–271, Madison, WI, June 2003.
- [Fergus & Perona⁺ 05] R. Fergus, P. Perona, A. Zisserman: A Sparse Object Category Model for Efficient Learning and Exhaustive Recognition. In C. Schmid, S. Soatto, C. Tomasi, editors, *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, pp. 380–389, San Diego, CA, USA, June 2005. IEEE.
- [Fischer & Modersitzki 01] B. Fischer, J. Modersitzki: A Super Fast Registration Algorithm. In *Proc. Bildverarbeitung für die Medizin*, pp. 169–173, Lübeck, Germany, March 2001. Springer.
- [Fitzgibbon & Zisserman 03] A. Fitzgibbon, A. Zisserman: Joint Manifold Distance: a New Approach to Appearance Based Clustering. In *Proc. Conf. Computer Vision and Pattern Recognition*, Vol. 1, pp. 26–33, Madison, WI, June 2003.
- [Forsyth & Ponce 03] D.A. Forsyth, J. Ponce: *Computer Vision: A Modern Approach*. Prentice Hall, Upper Saddle River, NJ, 2003.
- [Frey & Jojic 03] B.J. Frey, N. Jojic: Transformation-invariant clustering using the EM algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 25, No. 1, pp. 1–17, Jan. 2003.
- [Fukunaga 90] K. Fukunaga: *Introduction to Statistical Pattern Recognition*. Computer Science and Scientific Computing Academic Press Inc., San Diego, CA, 2nd edition, 1990.
- [Fussenegger & Opelt⁺ 04] M. Fussenegger, A. Opelt, A. Pinz, P. Auer: Object Recognition Using Segmentation for Feature Detection. In *ICPR*, Vol. 3, pp. 41–48, Cambridge, UK, aug 2004.
- [Gao & Vasconcelos 05] D. Gao, N. Vasconcelos. Discriminant Saliency for Visual Recognition from Cluttered Scenes. In *Advances in Neural Information Processing Systems 17*, pp. 481–488. MIT Press, Cambridge, MA, 2005.
- [Garey & Johnson 79] M.R. Garey, D.S. Johnson: *Computers and Intractability, A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.
- [Glasbey & Mardia 98] C.A. Glasbey, K.V. Mardia: A Review of Image Warping Methods. *J. Applied Statistics*, Vol. 25, No. 2, pp. 155–171, 1998.
- [Gollan 03] C. Gollan. Nichtlineare Verformungsmodelle für die Bilderkennung. Diploma thesis, Lehrstuhl für Informatik VI, RWTH Aachen University, Aachen, Sept. 2003.
- [Ha & Bunke 97] T.M. Ha, H. Bunke: Off-Line, Handwritten Numeral Recognition by Perturbation Method. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 5, pp. 535–539, May 1997.

- [Haasdonk & Halawani⁺ 04] B. Haasdonk, A. Halawani, H. Burkhardt: Adjustable Invariant Features by Partial Haar-Integration. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, pp. 769–774, Cambridge, UK, Aug. 2004.
- [Haasdonk & Keysers 02] B. Haasdonk, D. Keysers: Tangent Distance Kernels for Support Vector Machines. In *ICPR 2002, 16th Int. Conf. on Pattern Recognition*, Vol. II, pp. 864–868, Quebec City, Canada, Sept. 2002.
- [Haasdonk & Vossen⁺ 05] B. Haasdonk, A. Vossen, H. Burkhardt.: Invariance in Kernel Methods by Haar-Integration Kernels. In *Proc. 14th Scandinavian Conf. on Image Analysis*, pp. 841–851, June 2005.
- [Haralick & Shanmugam⁺ 73] R. Haralick, K. Shanmugam, I. Deinstein: Textural Features for Image Classification. *IEEE Trans. Systems, Man and Cybernetics*, Vol. 3, No. 6, pp. 610–621, Nov. 1973.
- [Hastie & Buja⁺ 95] T. Hastie, A. Buja, R. Tibshirani: Penalized Discriminant Analysis. *Annals of Statistics*, Vol. 23, No. 1, pp. 73–102, Jan. 1995.
- [Hastie & Simard⁺ 95] T. Hastie, P. Simard, E. Säcker: Learning Prototype Models for Tangent Distance. In G. Tesauro, D. Touretzky, T. Leen, editors, *Advances in Neural Inf. Proc. Systems*, Vol. 7, pp. 999–1006. MIT Press, July 1995.
- [Hastie & Simard 98] T. Hastie, P. Simard: Metrics and Models for Handwritten Character Recognition. *Statistical Science*, Vol. 13, No. 1, pp. 54–65, Jan. 1998.
- [Hastie & Tibshirani 96] T. Hastie, R. Tibshirani: Discriminative Adaptive Nearest Neighbor Classification. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, pp. 607–616, June 1996.
- [Hastie & Tibshirani⁺ 01] T. Hastie, R. Tibshirani, J. Friedman: *The Elements of Statistical Learning*. New York: Springer, 2001.
- [Heisele & Ho⁺ 01] B. Heisele, P. Ho, T. Poggio: Face Recognition with Support Vector Machines: Global versus Component-based Approach. In *Proc. 8th Int. Conf. on Computer Vision*, Vol. 2, pp. 688–694, Vancouver, Canada, 2001.
- [Hinton & Dayan⁺ 97] G.E. Hinton, P. Dayan, M. Revow: Modeling the Manifolds of Images of Handwritten Digits. *IEEE Trans. on Neural Networks*, Vol. 8, No. 1, pp. 65–74, Jan. 1997.
- [Hinton & Ghahramani⁺ 00] G. Hinton, Z. Ghahramani, Y. Teh: Learning to Parse Images. In S. Solla, T. Leen, K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pp. 463–469. MIT Press, June 2000.
- [Hinton & Revow⁺ 95] G.E. Hinton, M. Revow, P. Dayan: Recognizing Handwritten Digits Using Mixtures of Linear Models. In G. Tesauro, D. Touretzky, T. Leen, editors, *Advances in Neural Information Processing Systems 7*, pp. 1015–1022. MIT Press, 1995.
- [Hinton & Williams⁺ 92] G.E. Hinton, C.K.I. Williams, M. Revow: Adaptive Elastic Models for Hand-Printed Character Recognition. In *Advances in Neural Information Processing Systems 4*, pp. 512–519, 1992.
- [Holub & Perona 05] A. Holub, P. Perona: A Discriminative Framework for Modelling Object Classes. In C. Schmid, S. Soatto, C. Tomasi, editors, *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, pp. 663–670, San Diego, CA, USA, June 2005. IEEE.
- [Hu 62] M.K. Hu: Visual Pattern Recognition by Moment Invariants. *IEEE Trans. Information Theory*, Vol. 8, pp. 179–187, Feb. 1962.

- [Huttenlocher & Lilien⁺ 99] D.P. Huttenlocher, R.H. Lilien, C.F. Olson: View-Based Recognition Using an Eigenspace Approximation to the Hausdorff Measure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 21, No. 9, pp. 951–955, Sept. 1999.
- [Jaakkola & Meila⁺ 00] T. Jaakkola, M. Meila, T. Jebara: Maximum Entropy Discrimination. In *Advances in Neural Information Processing Systems 12*, pp. 470–476, Cambridge, MA, 2000. MIT Press.
- [Jaynes 82] E.T. Jaynes: On the Rationale of Maximum Entropy Models. *Proc. of the IEEE*, Vol. 70, No. 9, pp. 939–952, Sept. 1982.
- [Kambhatla & Leen 97] N. Kambhatla, T.K. Leen: Dimension reduction by local principal component analysis. *Neural Computation*, Vol. 9, No. 7, pp. 1493–1516, 1997.
- [Keijsper & Pendavingh 98] J. Keijsper, R. Pendavingh: An Efficient Algorithm for Minimum-Weight Bibranching. *J. Combin. Theory Ser. B*, Vol. 73, No. 2, pp. 130–145, 1998.
- [Keysers & Celik⁺ 02] D. Keysers, S. Celik, H. Braess, J. Dahmen, H. Ney: Parameter Estimation for Automatic Dose Control in Radioscopy. In *Bildverarbeitung für die Medizin*, pp. 279–282, Leipzig, Germany, March 2002. Springer.
- [Keysers & Dahmen⁺ 00a] D. Keysers, J. Dahmen, H. Ney: A Probabilistic View on Tangent Distance. In *22. DAGM Symposium Mustererkennung*, pp. 107–114, Kiel, Germany, Sept. 2000. Springer.
- [Keysers & Dahmen⁺ 00b] D. Keysers, J. Dahmen, T. Theiner, H. Ney: Experiments with an Extended Tangent Distance. In *Proc. 15th Int. Conf. on Pattern Recognition*, Vol. 2, pp. 38–42, Barcelona, Spain, Sept. 2000.
- [Keysers & Dahmen⁺ 01a] D. Keysers, J. Dahmen, H. Ney: Invariant Classification of Red Blood Cells. In *Bildverarbeitung für die Medizin*, pp. 367–371, Lübeck, Germany, March 2001. Springer.
- [Keysers & Dahmen⁺ 01b] D. Keysers, J. Dahmen, H. Ney, M.O. Güld: A Statistical Framework for Multi-Object Recognition. In *Informatiktage 2001 der Gesellschaft für Informatik*, pp. 73–76, Bad Schussenried, Germany, Oct. 2001. Konradin Verlag.
- [Keysers & Dahmen⁺ 03] D. Keysers, J. Dahmen, H. Ney, B. Wein, T. Lehmann: Statistical Framework for Model-based Image Retrieval in Medical Applications. *J. Electronic Imaging*, Vol. 12, No. 1, pp. 59–68, Jan. 2003.
- [Keysers & Deselaers⁺ 04] D. Keysers, T. Deselaers, H. Ney: Pixel-to-Pixel Matching for Image Recognition using Hungarian Graph Matching. In *DAGM 2004, Pattern Recognition, 26th DAGM Symposium*, Vol. 3175 of *Lecture Notes in Computer Science*, pp. 154–162, Tübingen, Germany, Aug. 2004. Received DAGM prize.
- [Keysers & Gollan⁺ 04a] D. Keysers, C. Gollan, H. Ney: Classification of Medical Images using Non-linear Distortion Models. In *Proc. BVM 2004, Bildverarbeitung für die Medizin*, pp. 366–370, Berlin, Germany, March 2004.
- [Keysers & Gollan⁺ 04b] D. Keysers, C. Gollan, H. Ney: Local Context in Non-linear Deformation Models for Handwritten Character Recognition. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. IV, pp. 511–514, Cambridge, UK, Aug. 2004.
- [Keysers & Macherey⁺ 01] D. Keysers, W. Macherey, J. Dahmen, H. Ney: Learning of Variability for Invariant Statistical Pattern Recognition. In *12th European Conf. on Machine Learning*, Vol. 2167 of *Lecture Notes in Computer Science*, pp. 263–275, Freiburg, Germany, Sept. 2001. Springer.

- [Keyzers & Macherey⁺ 04] D. Keyzers, W. Macherey, H. Ney, J. Dahmen: Adaptation in Statistical Pattern Recognition using Tangent Vectors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 26, No. 2, pp. 269–274, Feb. 2004.
- [Keyzers & Motter⁺ 03] D. Keyzers, M. Motter, T. Deselaers, H. Ney: Training and Recognition of Complex Scenes using a Holistic Statistical Model. In *DAGM 2003, Pattern Recognition, 25th DAGM Symposium*, Vol. 2781 of *Lecture Notes in Computer Science*, pp. 52–59, Magdeburg, Germany, Sept. 2003. Springer.
- [Keyzers & Ney 04] D. Keyzers, H. Ney: Linear Discriminant Analysis and Discriminative Log-linear Modeling. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. I, pp. 156–159, Cambridge, UK, Aug. 2004.
- [Keyzers & Och⁺ 02a] D. Keyzers, F.J. Och, H. Ney: Maximum Entropy and Gaussian Models for Image Object Recognition. In *Pattern Recognition, 24th DAGM Symposium*, Vol. 2449 of *Lecture Notes in Computer Science*, pp. 498–506, Zürich, Switzerland, Sept. 2002. Springer.
- [Keyzers & Och⁺ 02b] D. Keyzers, F.J. Och, H. Ney: Efficient Maximum Entropy Training for Statistical Object Recognition. In *Informatiktage 2002*, pp. 342–345, Bad Schussenried, Germany, Nov. 2002. Konradin Verlag.
- [Keyzers & Paredes⁺ 02] D. Keyzers, R. Paredes, H. Ney, E. Vidal: Combination of Tangent Vectors and Local Representations for Handwritten Digit Recognition. In *SPR 2002, Int. Workshop on Statistical Pattern Recognition*, Vol. 2396 of *Lecture Notes in Computer Science*, pp. 538–547, Windsor, Ontario, Canada, Aug. 2002. Springer.
- [Keyzers & Paredes⁺ 03] D. Keyzers, R. Paredes, E. Vidal, H. Ney: Comparison of Log-Linear Models and Weighted Dissimilarity Measures. In *IbPRIA 2003, 1st Iberian Conf. on Pattern Recognition and Image Analysis*, Vol. 2652 of *Lecture Notes in Computer Science*, pp. 370–377, Puerto de Andratx, Spain, June 2003. Springer.
- [Keyzers & Unger 03] D. Keyzers, W. Unger: Elastic Image Matching is NP-complete. *Pattern Recognition Letters*, Vol. 24, No. 1–3, pp. 445–453, Jan. 2003.
- [Keyzers 00] D. Keyzers. Approaches to Invariant Image Object Recognition. Diploma thesis, Lehrstuhl für Informatik VI, RWTH Aachen University, Aachen, June 2000.
- [Kim & Kim⁺ 02] H.J. Kim, D. Kim, S.Y. Bang: A numeral character recognition using the PCA mixture model. *Pattern Recognition Letters*, Vol. 23, No. 1–3, pp. 103–111, Jan. 2002.
- [Kittler 98] J. Kittler: On Combining Classifiers. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 20, No. 3, pp. 226–239, March 1998.
- [Knuth 94] D.E. Knuth: *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison-Wesley, Reading, MA, 1994.
- [Kohnen & Schubert⁺ 01] M. Kohnen, H. Schubert, B.B. Wein, R.W. Günther, J. Bredno, T.M. Lehmann, J. Dahmen: Qualität von DICOM-Informationen in Bilddaten aus der klinischen Routine. In H. Handels, A. Horsch, T. Lehmann, H.P. Meinzer, editors, *Bildverarbeitung für die Medizin 2001*, pp. 419–423, Lübeck, Germany, March 2001. Springer.
- [Kölsch & Keyzers⁺ 04] T. Kölsch, D. Keyzers, R. Paredes, H. Ney: Enhancements for Local Feature Based Image Classification. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. I, pp. 248–251, Cambridge, UK, Aug. 2004.
- [Kölsch 03] T. Kölsch. Local Features for Image Classification. Diploma thesis, Chair of Computer Science VI, RWTH Aachen University, Aachen, Germany, Nov. 2003.

-
- [Kuo & Agazzi 94] S. Kuo, O. Agazzi: Keyword Spotting in Poorly Printed Documents using Pseudo 2-D Hidden Markov Models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 16, No. 8, pp. 842–848, Aug. 1994.
- [Lazebnik & Schmid⁺ 05] S. Lazebnik, C. Schmid, J. Ponce: A Maximum Entropy Framework for Part-Based Texture and Object Recognition. In *ICCV 2005, Int. Conf. on Computer Vision*, Beijing, China, Oct. 2005. In press.
- [LeCun & Boser⁺ 89] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel: Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, Vol. 1, No. 4, pp. 541–551, 1989.
- [LeCun & Boser⁺ 90] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, L. Jackel: Handwritten Digit Recognition With a Back-Propagation Network. In *Advances in Neural Information Processing Systems 2*, pp. 396–404. Morgan Kaufmann, 1990.
- [LeCun & Bottou⁺ 98] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner: Gradient-Based Learning Applied to Document Recognition. *Proc. of the IEEE*, Vol. 86, No. 11, pp. 2278–2324, Nov. 1998.
- [LeCun & Bottou⁺ 04] Y. LeCun, L. Bottou, J. HuangFu: Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting. In *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pp. 97–104, Washington, D.C., June 2004. IEEE.
- [Lehmann & Güld⁺ 04] T. Lehmann, M. Güld, C. Thies, B. Fischer, K. Spitzer, D. Keysers, H. Ney, M. Kohnen, H. Schubert, B. Wein: Content-based Image Retrieval in Medical Applications. *Methods of Information in Medicine*, Vol. 43, No. 4, pp. 354–361, Oct. 2004.
- [Lehmann & Güld⁺ 05] T. Lehmann, M. Güld, T. Deselaers, D. Keysers, H. Schubert, K. Spitzer, H. Ney, B. Wein: Automatic Categorization of Medical Images for Content-based Retrieval and Data Mining. *Computerized Medical Imaging and Graphics*, Vol. 29, pp. 143–155, 2005. In press.
- [Lehmann & Schubert⁺ 03] T. Lehmann, H. Schubert, D. Keysers, M. Kohnen, B. Wein: The IRMA code for unique classification of medical images. In *Proc. Medical Imaging 2003*, Vol. 5033 of *Proc. SPIE*, pp. 109–117, San Diego, CA, May 2003.
- [Lehmann & Wein⁺ 00] T. Lehmann, B. Wein, J. Dahmen, J. Bredno, F. Vogelsang, M. Kohnen: Content-Based Image Retrieval in Medical Applications: A Novel Multi-Step Approach. In *Proc. SPIE*, Vol. 3972(32), pp. 312–320, Feb. 2000.
- [Lei & Govindaraju 04] H. Lei, V. Govindaraju: Direct Image Matching by Dynamic Warping. In *Proc. of the 2004 CVPR Workshop on Face Processing in Video (FPIV'04)*, pp. 76–80, Washington, DC, June 2004.
- [Leibe & Schiele 03] B. Leibe, B. Schiele: Interleaved Object Categorization and Segmentation. In *Proc. British Machine Vision Conf. (BMVC'03)*, Vol. II, pp. 264–271, Norwich, UK, Sept. 2003.
- [Leibe & Schiele 04] B. Leibe, B. Schiele: Scale Invariant Object Categorization Using a Scale-Adaptive Mean-Shift Search. In *DAGM 2004, Pattern Recognition, 26th DAGM Symposium*, Vol. 3175 of *Lecture Notes in Computer Science*, pp. 145–153, Tübingen, Germany, Aug. 2004.
- [Leonardis & Bischof 00] A. Leonardis, H. Bischof: Robust Recognition Using Eigenimages. *Computer Vision and Image Understanding*, Vol. 78, No. 1, pp. 99–118, April 2000.
- [Levin & Pieraccini 92] E. Levin, R. Pieraccini: Dynamic Planar Warping for Optical Character Recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 3, pp. 149–152, San Francisco, CA, March 1992.
- [Li & Lu 99] S.Z. Li, J. Lu: Face Recognition Using the Nearest Feature Line Method. *IEEE Trans. Neural Networks*, Vol. 10, No. 2, pp. 439–443, March 1999.

- [Li & Najmi⁺ 00] J. Li, A. Najmi, R.M. Gray: Image Classification by a Two-Dimensional Hidden Markov Model. *IEEE Trans. Signal Processing*, Vol. 48, No. 2, pp. 517–533, feb 2000.
- [Linde & Buzo⁺ 80] Y. Linde, A. Buzo, R. Gray: An Algorithm for Vector Quantizer Design. *IEEE Trans. Communications*, Vol. 28, No. 1, pp. 84–95, 1980.
- [Lindwurm & Breuer⁺ 96] R. Lindwurm, T. Breuer, K. Kreutzer: Multi Expert System for Hand-print Recognition. In *Progress in Handwriting Recognition*, pp. 293–298, Colchester, UK, Sept. 1996. World Scientific.
- [Liu & Dellaert 98] Y. Liu, F. Dellaert: A Classification Based Similarity Metric for 3D Image Retrieval. In *Procs. Int. Conf. Computer Vision and Pattern Recognition*, pp. 800–805, Santa Barbara, CA, June 1998.
- [Liu & Nakashima⁺ 03] C.L. Liu, K. Nakashima, H. Sako, H. Fujisawa: Handwritten Digit Recognition: Benchmarking of State-of-the-Art Techniques. *Pattern Recognition*, Vol. 36, No. 10, pp. 2271–2285, Oct. 2003.
- [Loupas & Sebe⁺ 00] E. Loupas, N. Sebe, S. Bres, J. Jolion: Wavelet-based Salient Points for Image Retrieval. In *International Conference on Image Processing*, Vol. 2, pp. 518–521, Vancouver, Canada, sep 2000.
- [Lowe 99] D.G. Lowe: Object Recognition from Local Scale-Invariant Features. In *Int. Conf. on Computer Vision*, pp. 1150–1–157, Corfu, Sept. 1999.
- [Macherey & Keysers⁺ 01] W. Macherey, D. Keysers, J. Dahmen, H. Ney: Improving Automatic Speech Recognition Using Tangent Distance. In *Eurospeech 2001, 7th European Conf. on Speech Communication and Technology*, Vol. III, pp. 1825–1828, Aalborg, Denmark, Sept. 2001.
- [Mardia & Qian⁺ 97] K. Mardia, W. Qian, D. Shah, K. de Souza: Deformable Template Recognition of Multiple Occluded Objects. *IEEE Trans. Pattern Analysis Machine Intelligence*, Vol. 19, No. 9, pp. 1035–1042, Sept. 1997.
- [Martinez & Kak 01] A. Martinez, A. Kak: PCA versus LDA. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 228–233, Feb. 2001.
- [Marée & Geurts⁺ 04] R. Marée, P. Geurts, J. Piater, L. Wehenkel: A Generic Approach for Image Classification Based on Decision Tree Ensembles and Local Sub-Windows. In K.S. Hong, Z. Zhang, editors, *Proc. of the 6th Asian Conf. on Computer Vision*, Vol. 2, pp. 860–865, Jeju Island, Korea, Jan. 2004.
- [Marée & Geurts⁺ 05] R. Marée, P. Geurts, J. Piater, L. Wehenkel: Random Subwindows for Robust Image Classification. In C. Schmid, S. Soatto, C. Tomasi, editors, *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 34–40, San Diego, CA, USA, June 2005. IEEE.
- [Matsumoto & Uchida⁺ 04] N. Matsumoto, S. Uchida, H. Sakoe: Prototype Setting for Elastic Matching-based Image Pattern Recognition. In *ICPR 2004, 17th Int. Conf. on Pattern Recognition*, Vol. I, pp. 224–227, Cambridge, UK, Aug. 2004.
- [Mayraz & Hinton 02] G. Mayraz, G. Hinton: Recognizing Handwritten Digits Using Hierarchical Products of Experts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 24, No. 2, pp. 189–197, Feb. 2002.
- [Meinicke & Ritter 99] P. Meinicke, H. Ritter: Local PCA Learning with Resolution-Dependent Mixtures of Gaussians. In *Proc. 9th Int. Conf. on Artificial Neural Networks*, pp. 497–502, Edinburgh, UK, 1999.

-
- [Merz & Murphy⁺ 97] C.J. Merz, P.M. Murphy, D.W. Aha. UCI Repository of Machine Learning Databases. University of California, Department of Information and Computer Science, Irvine CA, 1997. <http://www.ics.uci.edu/~mlearn/MLRepository.html>.
- [Messer & Kittler⁺ 03] K. Messer, J. Kittler, M. Sadeghi, S. Marcel, C. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, J. Czyz, L. Vandendorpe, S. Srisuk, M. Petrou, W. Kurutach, A. Kadyrov, R. Paredes, B. Kepenekci, F.B. Tek, G.B. Akar, F. Deravi, N. Mavity: Face Verification Competition on the XM2VTS Database. In *4th Int. Conf. on Audio and Video Based Biometric Person Authentication*, pp. 964–974, June 2003.
- [Michie & Spiegelhalter⁺ 94] D. Michie, D.J. Spiegelhalter, C.C. Taylor, editors: *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, 1994. Available at <http://www.amsta.leeds.ac.uk/~charles/statlog/>, datasets at <http://www.liacc.up.pt/ML/statlog/datasets.html>.
- [Milgram & Sabourin⁺ 05] J. Milgram, R. Sabourin, M. Cheriet: Combining Model-based and Discriminative Approaches in a Modular Two-stage Classification System: Application to Isolated Handwritten Digit Recognition. *Electronic Letters on Computer Vision and Image Analysis*, Vol. 5, No. 2, pp. 1–15, 2005.
- [Minka 00] T.P. Minka: Automatic Choice of Dimensionality for PCA. In T.K. Leen, T.G. Dietterich, V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pp. 598–604. MIT Press, 2000.
- [Moghaddam & Nastar⁺ 96] B. Moghaddam, C. Nastar, A. Pentland: A Bayesian Similarity Measure for Direct Image Matching. In *Proc. 13th Int. Conf. on Pattern Recognition*, pp. 350–358, Vienna, Austria, Aug. 1996.
- [Moghaddam & Pentland 97] B. Moghaddam, A. Pentland: Probabilistic Visual Learning for Object Representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 696–710, July 1997.
- [Mohan & Papageorgiou⁺ 01] A. Mohan, C. Papageorgiou, T. Poggio: Example-based Object Detection in Images by Components. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 4, pp. 349–361, April 2001.
- [Moore 79] R.K. Moore: A Dynamic Programming Algorithm for the Distance Between Two Finite Areas. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 1, No. 1, pp. 86–88, Jan. 1979.
- [Murase & Nayar 95] H. Murase, S. Nayar: Visual Learning and Recognition of 3-D Objects from Appearance. *Int. J. Computer Vision*, Vol. 14, No. 1, pp. 5–24, Jan. 1995.
- [Ney & Mergel⁺ 92] H. Ney, D. Mergel, A. Noll, A. Paeseler: Data Driven Search Organization for Continuous Speech Recognition. *IEEE Trans. Signal Processing*, Vol. 40, No. 2, pp. 271–281, Feb. 1992.
- [Ney & Ortmanns 00] H. Ney, S. Ortmanns: Progress in Dynamic Programming Search for LVCSR. *Proc. of the IEEE*, Vol. 88, No. 8, pp. 1224–1240, Aug. 2000.
- [Ney 95] H. Ney: On the Probabilistic Interpretation of Neural Net Classifiers and Discriminative Training Criteria. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, pp. 107–119, Feb. 1995.
- [Ney 99] H. Ney. Mustererkennung und Neuronale Netze. Script to the lecture on Pattern Recognition and Neural Networks held at RWTH Aachen, 1999.

- [Nigam & Lafferty⁺ 99] K. Nigam, J. Lafferty, A. McCallum: Using Maximum Entropy for Text Classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*, pp. 61–67, Stockholm, Sweden, Aug. 1999.
- [Niyogi & Girosi⁺ 98] P. Niyogi, F. Girosi, T. Poggio: Incorporating Prior Information in Machine Learning by Creating Virtual Examples. *Proc. of the IEEE*, Vol. 86, No. 11, pp. 2196–2209, Nov. 1998.
- [Normandin 96] Y. Normandin: Maximum Mutual Information Estimation of Hidden Markov Models. In C.H. Lee, F.K. Soong, K.K. Paliwal, editors, *Automatic Speech and Speaker Recognition*, pp. 57–81, Norwell, MA, 1996. Kluwer Academic Publishers.
- [Paredes & Keysers⁺ 02] R. Paredes, D. Keysers, T. Lehmann, B. Wein, H. Ney, E. Vidal: Classification of Medical Images using Local Representations. In *Bildverarbeitung für die Medizin*, pp. 171–174, Leipzig, Germany, March 2002. Springer.
- [Paredes & Perez-Cortes⁺ 01] R. Paredes, J. Perez-Cortes, A. Juan, E. Vidal.: Local Representations and a Direct Voting Scheme for Face Recognition. In *Workshop on Pattern Recognition in Information Systems*, pp. 71–79, Setúbal, Portugal, July 2001.
- [Paredes & Vidal 00] R. Paredes, E. Vidal: A class-dependent weighted dissimilarity measure for nearest-neighbor classification problems. *Pattern Recognition Letters*, Vol. 21, pp. 1027–1036, 2000.
- [Paredes & Vidal⁺ 02] R. Paredes, E. Vidal, D. Keysers: An Evaluation of the WPE Algorithm using Tangent Distance. In *ICPR 2002, 16th Int. Conf. on Pattern Recognition*, Vol. IV, pp. 48–51, Quebec City, Canada, Sept. 2002.
- [Pelkmann 99] A. Pelkmann. Entwicklung eines Klassifikators zur videobasierten Erkennung von Gesten. Diploma thesis, RWTH Aachen University, Aachen, Germany, Feb. 1999.
- [Perrey 00] R. Perrey. Affin-invariante Merkmale für die 2D-Bildererkennung. Diploma thesis, Chair of Computer Science VI, RWTH Aachen University, Aachen, Germany, Feb. 2000.
- [Perronnin & Dugelay⁺ 03] F. Perronnin, J.L. Dugelay, K. Rose: Iterative Decoding of Two-Dimensional Hidden Markov Models. In *ICASSP 2003, Int. Conf. Acoustics, Speech, and Signal Processing*, Vol. 3, pp. 329–332, Hong Kong, China, April 2003.
- [Pösl 98] J. Pösl. *Erscheinungsbasierte statistische Objekterkennung*. PhD thesis, Universität Erlangen Nürnberg, Erlangen, 1998. Shaker Verlag, Aachen.
- [Press & Teukolsky⁺ 92] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery: *Numerical Recipes in C*. Cambridge University Press, Cambridge, second edition, 1992.
- [Reinhold & Paulus⁺ 01] M. Reinhold, D. Paulus, H. Niemann: Appearance-Based Statistical Object Recognition by Heterogenous Background and Occlusions. In *Pattern recognition. 23rd DAGM Symposium*, number 2191 in Lecture Notes in Computer Science, pp. 254–261, Munich, Germany, Sept. 2001. Springer.
- [Revow & Williams⁺ 96] M. Revow, C.K.I. Williams, G.E. Hinton: Using Generative Models for Handwritten Digit Recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, pp. 592–606, 1996.
- [Ronee & Uchida⁺ 01] M.A. Ronee, S. Uchida, H. Sakoe: Handwritten character recognition using piecewise linear two-dimensional warping. In *Proc. 6th Int. Conf. on Document Analysis and Recognition*, pp. 39–43, Seattle, WA, Sept. 2001.
- [Roweis & Ghahramani 99] S. Roweis, Z. Ghahramani: A Unifying Review of Linear Gaussian Models. *Neural Computation*, Vol. 11, No. 2, pp. 305–345, 1999.

- [Roweis 98] S. Roweis: EM algorithms for PCA and SPCA. In M. Kearns, M. Jordan, S. Solla, editors, *Advances in Neural Information Processing Systems 10*, pp. 626–632, Cambridge, MA, 1998. MIT Press.
- [Rybach & Keyzers⁺ 05] D. Rybach, D. Keyzers, H. Ney: Erweiterung eines holistischen statistischen Bilderkenners zur Verwendung von mehreren Merkmalen. In *Informatiktag 2005*, pp. 155–158, St. Augustin, Germany, April 2005.
- [Salzberg 97] S.L. Salzberg: On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach. *Data Mining and Knowledge Discovery*, Vol. 1, No. 3, 1997.
- [Samaria 94] F.S. Samaria. *Face Recognition Using Hidden Markov Models*. PhD thesis, University of Cambridge, Cambridge, UK, Oct. 1994.
- [Schmid & Mohr 97] C. Schmid, R. Mohr: Local Grayvalue Invariants for Image Retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 5, pp. 530–535, May 1997.
- [Schnörr & Weickert 00] C. Schnörr, J. Weickert: Variational Image Motion Computation: Theoretical Framework, Problems and Perspectives. In *Proc. 22. DAGM Symposium Mustererkennung*, pp. 476–487, Kiel, Germany, Sept. 2000. Springer.
- [Schölkopf & Simard⁺ 98] B. Schölkopf, P. Simard, A. Smola, V. Vapnik: Prior Knowledge in Support Vector Kernels. In M.I. Jordan, M.J. Kearns, S.A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pp. 640–646. MIT Press, June 1998.
- [Schölkopf 97] B. Schölkopf: *Support Vector Learning*. Oldenbourg Verlag, Munich, 1997.
- [Schönfeld & Grebe 89] M. Schönfeld, R. Grebe: Automatic Shape Quantification of Freely Suspended Red Blood Cells by Isodensity Contour Tracing and Tangent Counting. *Computer Methods and Programs in Biomedicine*, Vol. 28, pp. 217–224, 1989.
- [Schulz-Mirbach 92] H. Schulz-Mirbach: On the Existence of Complete Invariant Feature Spaces in Pattern Recognition. In *Proc. of the 11th Int. Conf. on Pattern Recognition, Conf. B: Pattern Recognition Methodology and Systems*, Vol. II, pp. 178–182, Den Haag, The Netherlands, Aug. 1992.
- [Schwenk & Milgram 96] H. Schwenk, M. Milgram: Constraint Tangent Distance for On-line Character Recognition. In *12th Int. Conf. on Pattern Recognition*, Vol. D, pp. 515–519, 1996.
- [Short & Fukunaga 81] R.D. Short, K. Fukunaga: The Optimal Distance Measure for Nearest Neighbor Classification. *IEEE Trans. Information Theory*, Vol. 27, No. 5, pp. 422–427, Sept. 1981.
- [Siggelkow 02] S. Siggelkow. *Feature Histograms for Content-Based Image Retrieval*. Ph.D. thesis, University of Freiburg, Institute for Computer Science, Freiburg, Germany, 2002.
- [Simard & Le Cun⁺ 92] P. Simard, Y. Le Cun, J. Denker, B. Victorri: An Efficient Algorithm for Learning Invariances in Adaptive Classifiers. In *Proc. 11th Int. Conf. on Pattern Recognition*, pp. 651–655, The Hague, The Netherlands, Aug. 1992.
- [Simard & Le Cun⁺ 93] P. Simard, Y. Le Cun, J. Denker: Efficient Pattern Recognition Using a New Transformation Distance. In S. Hanson, J. Cowan, C. Giles, editors, *Advances in Neural Information Processing Systems 5*, pp. 50–58, San Mateo, CA, 1993. Morgan Kaufmann.
- [Simard & Le Cun⁺ 98a] P. Simard, Y. Le Cun, J. Denker, B. Victorri: Transformation Invariance in Pattern Recognition — Tangent Distance and Tangent Propagation. In G. Orr, K.R. Müller, editors, *Neural Networks: Tricks of the Trade*, Vol. 1524 of *Lecture Notes in Computer Science*, pp. 239–274, Heidelberg, Nov. 1998. Springer.

- [Simard & Le Cun⁺ 98b] P. Simard, Y. Le Cun, J. Denker, B. Victorri: Transformation Invariance in Pattern Recognition — Tangent Distance and Tangent Propagation. In G. Orr, K.R. Müller, editors, *Neural networks: tricks of the trade*, Vol. 1524 of *Lecture Notes in Computer Science*, pp. 239–274, Heidelberg, 1998. Springer.
- [Simard & Steinkraus⁺ 03] P. Simard, D. Steinkraus, J.C. Platt: Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis. In *7th Int. Conf. Document Analysis and Recognition*, pp. 958–962, Edinburgh, Scotland, Aug. 2003.
- [Simard & Victorri⁺ 92] P. Simard, B. Victorri, Y. Le Cun, J. Denker: Tangent Prop—A Formalism for Specifying Selected Invariances in an Adaptive Network. In J.E. Moody, S.J. Hanson, R.P. Lippmann, editors, *Advances in Neural Information Processing Systems 4*, pp. 895–903, San Mateo, CA, April 1992. Morgan Kaufmann Publishers, Inc.
- [Simard 94] P. Simard: Efficient Computation of Complex Distance Metrics Using Hierarchical Filtering. In J.D. Cowan, G. Tesauero, J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pp. 168–175. Morgan Kaufmann Publishers, Inc., April 1994.
- [Smeulders & Worring⁺ 00] A.W.M. Smeulders, M. Worring, S. Santint, A. Gupta, R. Jain: Content-Based Image Retrieval at the End of the Early Years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22, pp. 1349–1380, Dec. 2000.
- [Smith & Bourgoïn⁺ 94] S.J. Smith, M.O. Bourgoïn, K. Sims, H.L. Voorhees: Handwritten Character Classification Using Nearest Neighbor in Large Databases. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 16, No. 9, pp. 915–919, Sept. 1994.
- [Sohn 99] S.Y. Sohn: Meta Analysis of Classification Algorithms for Pattern Recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 21, No. 11, pp. 1137–1144, Nov. 1999.
- [Steinbiss & Ullrich⁺ 88] V. Steinbiss, R. Ullrich, H. Ney: Curve Fitting for the Recognition of Line Drawings. In *Signal Processing IV: Theories and Applications, Fourth European Signal Processing Conf.*, Vol. III, pp. 1449–1452, Grenoble, France, Sept. 1988.
- [Sullivan & Blake⁺ 01] J. Sullivan, A. Blake, M. Isard, J. MacCormick: Bayesian object localisation in images. *Int. J. Computer Vision*, Vol. 44, No. 2, pp. 111–135, 2001.
- [Teow & Loe 00] L.N. Teow, K.F. Loe: Handwritten Digit Recognition with a Novel Vision Model that Extracts Linearly Separable Features. In *Proc. CVPR 2000, Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 76–81, Hilton Head, SC, June 2000.
- [Teow & Loe 02] L.N. Teow, K.F. Loe: Robust Vision-Based Features and Classification Schemes for Off-Line Handwritten Digit Recognition. *Pattern Recognition*, Vol. 35, No. 11, pp. 2355–2364, Nov. 2002.
- [Tipping & Bishop 99] M.E. Tipping, C.M. Bishop: Mixtures of Probabilistic Principal Component Analysers. *Neural Computation*, Vol. 11, No. 2, pp. 443–482, 1999.
- [Tipping 00] M.E. Tipping: The Relevance Vector Machine. In S. Solla, T. Leen, K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pp. 332–388, Cambridge, MA, 2000. MIT Press.
- [Toselli & Juan⁺ 04] A.H. Toselli, A. Juan, J. González, I. Salvador, E. Vidal, F. Casacuberta, D. Keysers, H. Ney: Integrated Handwriting Recognition and Interpretation using Finite State Models. *Int. J. Pattern Recognition and Image Analysis*, Vol. 18, No. 4, pp. 519–539, June 2004.
- [Tu & Chen⁺ 03] Z. Tu, X. Chen, A. Yuille, S.C. Zhu: Image Parsing: Segmentation, Detection, and Object Recognition. In *Proc. 9th IEEE Int. Conf. on Computer Vision*, pp. 18–25, Nice, France, Oct. 2003. IEEE.

-
- [Turk & Pentland 91] M. Turk, A. Pentland: Eigenfaces for recognition. *J. Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71–86, 1991.
- [Uchida & Ronee⁺ 02] S. Uchida, M.A. Ronee, H. Sakoe: Using Eigen-Deformations in Handwritten Character Recognition. In *ICPR 2002, 16th Int. Conf. on Pattern Recognition*, Vol. I, pp. 572–575, Quebec City, Canada, Sept. 2002.
- [Uchida & Sakoe 98] S. Uchida, H. Sakoe: A monotonic and continuous two-dimensional warping based on dynamic programming. In *Proc. 14th Int. Conf. on Pattern Recognition*, Vol. 1, pp. 521–524, Brisbane, Australia, Aug. 1998.
- [Uchida & Sakoe 99a] S. Uchida, H. Sakoe: An Efficient Two-Dimensional Warping Algorithm. *IEICE Trans. Information & Systems*, Vol. E82-D, No. 3, pp. 693–700, March 1999.
- [Uchida & Sakoe 99b] S. Uchida, H. Sakoe: Handwritten Character Recognition Using Monotonic and Continuous Two-Dimensional Warping. In *Proc. 5th Int. Conf. on Document Analysis and Recognition*, pp. 499–502, Bangalore, India, Sept. 1999.
- [Uchida & Sakoe 00a] S. Uchida, H. Sakoe: An Approximation Algorithm for Two-Dimensional Warping. *IEICE Trans. Information & Systems*, Vol. E83-D, No. 1, pp. 109–111, Jan. 2000.
- [Uchida & Sakoe 00b] S. Uchida, H. Sakoe: Piecewise Linear Two-Dimensional Warping. In *Proc. of the 15th Int. Conf. on Pattern Recognition, Barcelona, Spain*, Vol. 3, pp. 538–541, Sept. 2000.
- [Uchida & Sakoe 02] S. Uchida, H. Sakoe: A handwritten character recognition method based on unconstrained elastic matching and eigen-deformations. In *Proc. of the 8th Int. Workshop on Frontiers of Handwriting Recognition*, pp. 72–77, Niagara-on-the-Lake, Ontario, Canada, Aug. 2002.
- [Uchida & Sakoe 03a] S. Uchida, H. Sakoe: Eigen-Deformations for Elastic Matching based Handwritten Character Recognition. *Pattern Recognition*, Vol. 36, No. 9, pp. 2031–2040, Sept. 2003.
- [Uchida & Sakoe 03b] S. Uchida, H. Sakoe: Handwritten character recognition using elastic matching based on a class-dependent deformation model. In *7th Int. Conf. Document Analysis and Recognition*, pp. 163–167, Edinburgh, Scotland, Aug. 2003.
- [Ullman & Vidal-Naquet⁺ 02] S. Ullman, M. Vidal-Naquet, E. Sali: Visual features of intermediate complexity and their use in classification. *Nature Neuroscience*, Vol. 5, No. 7, pp. 692–687, July 2002.
- [Vapnik 95] V. Vapnik: *The Nature of Statistical Learning Theory*. Springer, New York, 1995.
- [Vapnik 98] V. Vapnik: *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.
- [Vapnik 99] V. Vapnik: An Overview of Statistical Learning Theory. *IEEE Trans. Neural Networks*, Vol. 10, No. 5, pp. 988–999, Sept. 1999.
- [Vasconcelos & Lippman 98] N. Vasconcelos, A. Lippman: Multiresolution Tangent Distance for Affine-invariant Classification. In M.I. Jordan, M.J. Kearns, S.A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pp. 843–849. MIT Press, June 1998.
- [Veltkamp & Hagedoorn 01] R. Veltkamp, M. Hagedoorn: State-of-the-art in Shape Matching. In M. Lew, editor, *Principles of Visual Information Retrieval*, pp. 87–119. Springer, 2001.
- [Viola & Wells 97] P. Viola, W. Wells: Alignment by Maximization of Mutual Information. *Int. J. Computer Vision*, Vol. 24, No. 2, pp. 137–154, 1997.
- [Wang & Srihari 88] C. Wang, S. Srihari: A Framework for Object Recognition in a Visually Complex Environment and its Application to Locating Address Blocks on Mail Pieces. *Int. J. Computer Vision*, Vol. 2, pp. 125–151, 1988.

- [Weber & Welling⁺ 00] M. Weber, M. Welling, P. Perona: Unsupervised Learning of Models for Recognition. In *European Conference on Computer Vision*, Vol. 1, pp. 18–32, Dublin, Ireland, June 2000.
- [Wiskott & Fellous⁺ 97] L. Wiskott, J. Fellous, N. Kruger, C.V. der Malsburg.: Face Recognition by Elastic Bunch Graph Matching. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 775–779, July 1997.
- [Wood 96] J. Wood: Invariant Pattern Recognition: A Review. *Pattern Recognition*, Vol. 29, No. 1, pp. 1–17, Jan. 1996.
- [Würtz 97] R.P. Würtz: Object Recognition Robust Under Translations, Deformations, and Changes in Background. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 769–775, July 1997.
- [Yang & Kriegman⁺ 02] M.H. Yang, D. Kriegman, N. Ahuja: Detecting Faces in Images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 24, No. 1, pp. 34–58, Jan. 2002.
- [Zahedi & Keysers⁺ 05a] M. Zahedi, D. Keysers, T. Deselaers, H. Ney.: Combination of Tangent Distance and Image Distortion for Appearance-Based Sign Language Recognition. In *DAGM 2005, Pattern Recognition, 27th DAGM Symposium*, Vol. LNCS 3663, pp. 401–408, Vienna, Austria, Aug. 2005.
- [Zahedi & Keysers⁺ 05b] M. Zahedi, D. Keysers, H. Ney: Appearance-Based Recognition of Words in American Sign Language. In *IbPRIA 2005, 2nd Iberian Conf. on Pattern Recognition and Image Analysis*, Vol. LNCS 3522, pp. 513–520, Estoril, Portugal, June 2005.
- [Zahedi & Keysers⁺ 05c] M. Zahedi, D. Keysers, H. Ney: Pronunciation Clustering and Modeling of Variability for Appearance-Based Sign Language Recognition. In *GW 2005, 6th Int. Workshop on Gesture in Human-Computer Interaction and Simulation*, Vannes, France, May 2005.
- [Zhang & Huang⁺ 05] H. Zhang, W. Huang, Z. Huang, B. Zhang: A Kernel Autoassociator Approach to Pattern Classification. *IEEE Trans. Systems, Man and Cybernetics, Part B*, Vol. 35, No. 3, pp. 593–606, June 2005.

Lebenslauf — Curriculum Vitae

Angaben zur Person

Name	Daniel Martin Keyzers
Geburtsdatum	14. Februar 1975
Geburtsort	Düsseldorf
Staatsangehörigkeit	deutsch
Familienstand	verheiratet, ein Sohn

Schulbildung

August 1981 – Juni 1985	Grundschule Heerdt, Düsseldorf
August 1985 – Juni 1994	Mataré-Gymnasium, Meerbusch, Abitur (Note: 1,0)
August 1991 – Februar 1992	Perris High School, Perris, CA

Zivildienst

Juli 1994 – September 1995	Pflegedienst, Seniorenheim Haus Lörriek, Düsseldorf
----------------------------	---

Studium

Oktober 1995 – Juni 2000	Informatikstudium an der RWTH Aachen, Stipendien der Studienstiftung des deutschen Volkes und der DaimlerChrysler Studienförderung Forschung und Technologie Abschluss: Diplom in Informatik mit Auszeichnung
Oktober 1997 – Juli 1998	Informatikstudium an der Univ. Complutense de Madrid
seit Juli 2000	Promotionsstudent an der RWTH Aachen
in 2002	Gastwissenschaftler am Instituto Tecnológico de Informática, Universidad Politecnica de Valencia (3 Monate)

Arbeitstätigkeiten

Oktober 1998 – April 2000	Studentische Hilfskraft am Lehrstuhl für Informatik VII der RWTH Aachen
April 1999 – Dezember 1999	Studentische Hilfskraft am Lehrstuhl für Informatik VI der RWTH Aachen
August 2000 – April 2005	Wissenschaftlicher Mitarbeiter am Lehrstuhl für Informatik VI der RWTH Aachen
seit Mai 2005	Researcher am Deutschen Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Arbeitsgruppe Image Understanding and Pattern Recognition