Erscheinungsbasierte 3-D Objekterkennung

von Andrea Wenning

Betreuer: Daniel Keysers

Inhalt

Zusammenfassung

In dieser Ausarbeitung wird ein Verfahren zum automatischen Lernen von Objektmodellen für die Erkennung und die Positionsbestimmung vorgestellt. Hierbei wird ein Objekt durch eine Menge von 2-D Bildern beschrieben. Da für ein Objekt in einem 2-D Bild die Ausrichtung und die Beleuchtungsrichtung variabel sind, kann das Objekt bezüglich dieser beiden Parameter kompakt dargestellt werden. Als erstes wird der Eigenraum bestimmt. In diesem wird das Objekt durch eine parametrische Mannigfaltigkeit beschrieben. Für die Erkennung eines Objektes in einem Bild, wird das Bild in den Eigenraum projiziert. Dann wird, um das Objekt zu bestimmen, die Mannigfaltigkeit gesucht, die am nächsten an dem projizierten Bild liegt. Die Ausrichtung des Objektes wird durch die genaue Position auf der Mannigfaltigkeit wiedergegeben.
Das in dieser Ausarbeitung vorgestellte Verfahren zeigt gute Ergebnisse, jedoch ist zum einen die geeignete Wahl der Grösse des Eigenraums nicht bekannt und zum anderen ist der Einsatz in einer realen Anwendung schwierig.

Keywords: Eigenraum, Mannigfaltigkeit, Erscheinung, Objekterkennung, Modellierung

1 Einleitung

Im medizinischen Image Retrieval werden große Mengen von medizinischen Bildern, z.B. Röntgen-, CT- oder MR-Bilder, in Datenbanken verwaltet. Dazu werden Methoden bereit gestellt, um Bilder zu finden, Merkmale zu extrahieren und zu vergleichen. Ein Objekt, z.B. der menschliche Schädel, wird nicht kompakt dargestellt, sondern unabhängig durch einzelne Bilder. Abbildung 1 stellt verschiedene Röntgenbilder von Schädeln dar. Betrachtet man den Schädel von unterschiedlichen Seiten, so stellt man fest, daß der Schädel von hinten anders aussieht als von vorne oder von der Seite. Dieses gibt die Frage auf, wie diese Informationen über das Objekt "Schädel" geeignet darstellt werden können, um zum einen das Objekt selber zu erkennen und zum anderen die Ausrichtung des Objektes zu bestimmen, beim Schädel z.B. frontal oder seitlich. Ein 3-D Objekt kann durch eine Menge von 2-D Bildern dargestellt werden. Diese Bilder werden in den Eigenraum projiziert. In dem das Objekt dann kompakt durch eine parametrische Mannigfaltigkeit repräsentiert wird, die von der Ausrichtung des Objektes im Bild und der Beleuchtungsrichtung abhängt. Für die Erkennung eines Bildes, wird das Bild in den Eigenraum projiziert. Dann wird, um das Objekt zu bestimmen, die Mannigfaltigkeit gesucht, die am nächsten an dem projizierten Bild liegt. Die Ausrichtung des Objektes wird durch die genaue Position auf der Mannigfaltigkeit wiedergegeben.

2 Ansatz zur Darstellung von 3-D Objekten

Ein Erkennungssystem für 3-D Objekte benötigt die Modelle der Objekte in seinem Speicher. In der Vergangenheit wurden z.B. geometrische Modelle erstellt, die die Gestalt der Objekte beschreiben. Hierbei wurden die Objekte mit Computer Aided Design (CAD) Modellen dargestellt. Das Problem dabei ist, daß man nicht alle Objekte, z.B. das Skelett eines Menschen, mit CAD-Modellen modellieren kann. Weiterhin muß für jedes Objekt ein Modell von Hand erstellt werden. Dieses ist aufwendig und unpraktisch, wenn man eine große Anzahl von Objekten darstellen möchte. Da jedes Modell erst von einem Menschen erstellt werden muß, ist ein System, welches auf diesem Ansatz basiert niemals in der Lage selbständig Modelle zu erstellen.
Betrachtet man einen Menschen, wie er sich ein 3 dimensionales Objekt visuell merkt, so beobachtet

Abbildung 1: Röntgenbilder von Schädeln

Abbildung 2: Ein Objekt aus der Datenbank

man, daß der Mensch das Objekt rotiert und die Erscheinung des Objektes von unterschiedlichen Richtungen betrachtet. Weiterhin hat man festgestellt, daß der Mensch wahrscheinlich 3-D Objekte als eine Menge von 2-D Bildern repräsentiert. Versucht man diese Beobachtung auf ein automatisches visuelles Lernsystem umzusetzen, so muß man sich als erstes überlegen, was die Erscheinung eines Objektes ist. Erscheinung kann man als Kombination von Gestalt, Reflektion, Ausrichtung in der Szene und Beleuchtungsbedingungen auffassen. Gestalt und Reflektion sind feste Parameter, die sich für ein Objekt nicht ändern, während sich die Ausrichtung und die Beleuchtung verändern können. Dieses wird in Abbildung 2 am Beispiel einer Ente gezeigt. Ein Objekt wird daher in dem hier vorgestellten Ansatz aus einer Menge von Bildern dargestellt, die unter verschiedenen Positionen und Beleuchtungen aufgenommen wurden [1].

3 Erscheinungsbasiertes Lernen

Jedes Objekt wird in diesem Ansatz durch einen Menge von Bildern dargestellt. Jedes dieser Bilder wird normalisiert und durch einen Merkmalsvektor, der die Pixelwerte enthält, dargestellt. Um die Dimension zu reduzieren wird die Hauptachsentransformation verwendet. Es werden dabei zwei Eigenräume bestimmt. Der universelle Eigenraum dient zur Klassifikation und der Objekteigenraum zur Positionsbestimmung. Die Objekte werden durch parametrische Mannifaltigkeiten repräsentiert, die mittels Splines bestimmt werden. Abbildung 3 zeigt ein Objekt mit zugehöriger Mannigfaltigkeit. Die Mannigfaltigkeit wird zur Veranschaulichung im 3-D Raum dargestellt. Dieser wird durch die ersten drei Eigenvektoren des Eigenraums aufgespannt. Jedes in diesen Eigenraum projizierte Bild entspricht einem Punkt, welcher in diesem Fall nur die Ausrichtung als Parameter hat. Die Beleuchtungsrichtung wird hier als konstant angesehen. Interpoliert man diese Punkte so erhält man die dargestellte Kurve. Die Kurve ist geschlossen, da, wenn man das Objekt um 360 Grad gedreht hat, man wieder am Ausgangspunkt ankommt.

Abbildung 3: Ein Objekt mit zugehöriger Mannigfaltigkeit

3.1 Normalisierung

Jedes Objekt wird durch eine Menge von Bildern dargestellt. Um die Bilder unabhängig von der Größe und von der Helligkeit zu repräsentieren, werden die Bilder normiert.
Die Bilder werden als erstes in der Größe normiert. Hierbei werden sie in Hintergrund und Objektregion segmentiert. Dieses ist möglich, da vorrausgesetzt wird, daß der Hintergrund schwarz ist. Jedes normalisierte Bild wird als Merkmalsvektor aus Pixelwerten geschrieben. Damit die Variationen in der Helligkeit eines Bildes die Erkennung nicht beeinflußt, werden die Bilder in der Helligkeit normalisiert.

Die Energie in einem Bild wird dazu auf den Wert 1 normiert.
Jeder dieser normalisierten Bildervektoren wird durch $b_{r,l}^{(k)}$ beschrieben. Hierbei gibt r die Rotation oder die Ausrichtung, l die Beleuchtungsrichtung und k die Nummer des Objektes an. Die komplette Menge von Bildern für ein Objekt wird als Objektmenge bezeichnet:

$$\{b_{1,1}^{(k)}, ..., b_{R,1}^{(k)}, ..., b_{R,L}^{(k)}\} \tag{1}$$

R und L geben die Anzahl von diskreten Positionen und Beleuchtungsrichtung an. Fasst man alle Objekte in einer Menge zusammen, so erhält man die universelle Bildmenge, wobei K die Anzahl der Bilder darstellt:

$$\begin{aligned}\{&b_{1,1}^{(1)}, ..., b_{R,1}^{(1)}, ..., b_{R,L}^{(1)},\\ &b_{1,1}^{(2)}, ..., b_{R,1}^{(2)}, ..., b_{R,L}^{(2)},\\ &\quad ...\\ &b_{1,1}^{(K)}, ..., b_{R,1}^{(K)}, ..., b_{R,L}^{(K)}\}\end{aligned} \tag{2}$$

3.2 Eigenraum

Die Bildmenge ist sehr groß. Betrachtet man ein Bild der Größe 128×128, so erhält man einen normalisierten Bildvektor $b_{r,l}^{(k)}$ mit mehr als 10000 Komponenten. Um eine effiziente Berechnung durchführen zu können, wird als erstes jedes Bild in einen niedriger dimensionalen Unterraum abgebildet. Hierzu wird die Hauptachsentransformation (PCA) verwendet [2][3]. Es werden zwei Unterräume bestimmt, der universelle Eigenraum und der Objekteigenraum. Der universelle Eigenraum wird auf allen Bildern berechnet und dient zur Klassifizierung eines Objektes. Der Objekteigenraum wird auf Bildern von einem Objekt berechnet, welcher zur Positionsbestimmung verwendet wird.

3.2.1 PCA

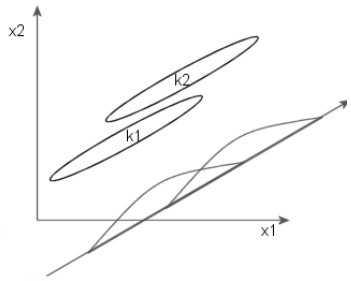

Abbildung 4: Hauptachsentransformation

Die Hauptachsentransformation projiziert die Merkmale in einen niedrieger dimensionalen Raum, so daß die Repräsentation der Klasse möglichst gut ist. Abbildung 4 zeigt dieses für einen 2 dimensionalen Fall. Dargestellt sind die Verteilungen der beiden Klassen $k1$ und $k2$. Mit der PCA wird der Unterraum bestimmt, der in diesem Fall durch die eingezeichnete Gerade repräsentiert wird. Werden die Elemente der Klassen in diesen Unterraum projiziert, so ergeben sich für die Klassen die eingezeichneten Verteilungen. Man sieht, daß die Gerade die Richtung der größten Streuung angibt.
Die Streuung der Merkmale ist durch die Streuungsmatrix gegeben:

$$S = \sum_{k=1}^{n} (x_k - m)(x_k - m)^T \tag{3}$$

$$m = \frac{1}{n} \sum_{k=1}^{n} x_k \tag{4}$$

n ist die Anzahl der Merkmalsvektoren und m ist der Mittelwert aller Merkmalsvektoren x_k. Gesucht wird nun eine Matrix V, die die Daten so in einen niedrig-dimensionalen Raum projiziert, daß die Streuung im Unterraum maximal wird. Die Dimension der Projektion wird als bekannt vorausgesetzt, bzw. ist variabel. Dazu maximiert man:

$$|V^T S V| \tag{5}$$

mit der Nebenbedingung:

$$V^T V = I \tag{6}$$

Dieses kann durch das Eigenwertproblem der Streuungsmatrix S gelöst werden. Die Eigenwerte und Eigenvektoren erfüllen dabei folgenden Zusammenhang:

$$\lambda e_i = S e_i \tag{7}$$

Die Eigenvektoren e_i bilden sortiert nach der Größe der zugehörigen Eigenwerte die Spalten der Matrix V. Die Stärke der Komprimierung ist variabel, je nach Wahl der Anzahl an Spalten von V, d.h. der Eigenvektoren. Werden wenige Eigenvektoren benutzt, so gehen viele Informationen verloren. Dadurch ist die Repräsentation nicht sehr genau. Nimmt man viele Eigenvektoren, so hat man viele Informationen und damit eine sehr genaue Beschreibung. Aber es sind nicht alle Informationen wichtig für die Repräsentation. Um eine geeignete Anzahl h von Eigenvektoren zu bestimmen, kann man den Quotienten aus der Summe der größten h Eigenwerte und der Summe aller Eigenwerte betrachten. In dem hier beschriebene Verfahren hat sich gezeigt, daß die Anzahl $h = 20$ ausreichend ist, um eine gut Erkennung zu erreichen.

$$V = [e_1, ..., e_h] \tag{8}$$

3.2.2 Universeller- und Objekteigenraum

Der universelle Eigenraum wird auf der gesamten Bildmenge berechnet. Die Streuungsmatrix der Bildmenge ist gegeben durch:

$$S_u = \sum_{k=1}^{K} \sum_{r=1}^{R} \sum_{l=1}^{L} (x_{r,l}^{(k)} - m)(x_{r,l}^{(k)} - m)^T \tag{9}$$

$$m = \frac{1}{n} \sum_{k=1}^{K} \sum_{r=1}^{R} \sum_{l=1}^{L} x_{r,l}^{(k)} \tag{10}$$

m ist der Mittelwert aller Bilder und $n = K \cdot R \cdot L$ die Anzahl aller Bilder. Von dieser Matrix S_u werden mittels PCA die Eigenvektoren e_i bestimmt. Die Eigenvektoren spannen den universellen Eigenraum auf. Der Objekteigenraum wird auf der Objektmenge bestimmt. Hierzu wird die Streuungsmatrix von allen Bildern zu einem Objekt bestimmt. Die Matrix ist gegeben durch:

$$S_o^{(k)} = \sum_{r=1}^{R} \sum_{l=1}^{L} (x_{r,l}^{(k)} - m^{(k)})(x_{r,l}^{(k)} - m^{(k)})^T \tag{11}$$

$$m^{(k)} = \frac{1}{n_k} \sum_{r=1}^{R} \sum_{l=1}^{L} x_{r,l}^{(k)} \tag{12}$$

Hierbei ist $m^{(k)}$ der Mittelwert der Bilder von einem Objekt k und $n_k = R \cdot L$ die Anzahl der Bilder von einem Objekt. Von dieser Matrix werden die Eigenvektoren e_i^k mittels PCA berechnet. Die Eigenvektoren spannen den Objekteigenraum auf. In dem hier beschriebenen Experiment wurde die Anzahl der Eigenvektoren $h = 20$ gewählt.

3.3 Repräsentation der Erscheinung

Die Repräsentation der Objekte findet in den Eigenräumen statt. In dem universellen Eigenraum wird die Erscheinungscharakteristik der Objekte beschrieben. Dazu wird jedes Bild aus der Bildmenge in den universellen Eigenraum projiziert. Als erstes wird von jedem Bild $x_{r,l}^{(k)}$ der Mittelwert m der Bildmenge abgezogen und dieses wird dann mit den Eigenvektoren multipliziert.

$$g_{r,l}^{(k)} = [e_1, ..., e_h]^T (x_{r,l}^{(k)} - m) = V^T (x_{r,l}^{(k)} - m) \tag{13}$$

Dadurch erhält man diskrete Punkte. Die Punkte beschreiben eine Mannigfaltigkeit.

$$g^{(k)}(\Theta_1, \Theta_2, ..., \Theta_m) \tag{14}$$

$\Theta_1, \Theta_2, ..., \Theta_m$ sind kontinuierliche Parameter für Ausrichtung und Beleuchtungsrichtung. Die Anzahl der Parameter kann variieren. Davon abhängig beschreibt das Mannigfaltigkeit eine Kurve, Oberfläche oder ein Volumen. Da im folgenden beschriebenen Experiment nur zwei Parameter, einer für die Ausrichtung und einer für die Beleuchtungrichtung, benutzt werden, wird die Mannigfaltigkeit gegeben durch:

$$g^{(k)}(\Theta_1, \Theta_2) \tag{15}$$

Die Mannigfaltigkeit wird mittels Cubic Spline bestimmt. Die Idee der Splines ist, daß man stückweise mit Polynomen eines festen Grades interpoliert. Hierbei wird nicht der Polynomgrad sondern die Anzahl der Stücke erhöht. Die diskreten Punkte dienen als Stützpunkte für die Interpolation. Da hier Cubic Splines verwendet werden, haben die Polynome den Grad drei.
Jedes Bild $x_{r,l}^{(k)}$ aus der Objektmenge wird in den Objekteigenraum projiziert. Dieses ist genauso wie bei der Projektion in den universellen Eigenraum. Es werden nur die Bilder zu einen Objekt betrachtet. Von diesen wird der Mittelwert $m^{(k)}$ der Objektbilder abgezogen.

$$f_{r,l}^{(k)} = [e_1^{(k)}, ..., e_h^{(k)}]^T (x_{r,l}^{(k)} - m^{(k)}) = {V^{(k)}}^T (x_{r,l}^{(k)} - m^{(k)}) \tag{16}$$

Von den diskreten Punkten wird mittels Cubic Spline Interpolation eine Mannigfaltigkeit bestimmt, die wie oben erwähnt hier nur von zwei Parametern abhängt, der Beleuchtungsrichtung und der Ausrichtung.

$$f^{(k)}(\Theta_1, \Theta_2) \tag{17}$$

Das Bild 5 zeigt Mannigfaltigkeiten, die zur Veranschaulichung nur bezüglich der ersten drei Eigenvektoren zu den größten Eigenwerten dargestellt sind. Jedes in diesen Eigenraum projizierte Bild entspricht einem Punkt, welcher hier die Ausrichtung Θ_1 und die Beleuchtungsrichtung Θ_2 als Parameter hat. Die Punkte werden interpoliert. Dadurch erhält man die dargestellten Kurven. Die Kurve ist entlang Θ_1 geschlossen, da, wenn man das Objekt um 360 Grad gedreht hat, man wieder am Ausgangspunkt ankommt. Da es für jede Ausrichtung mehrere Beleuchtungsrichtungen gibt, hat die Kurve eine gewisse Breite.

4 Objekterkennung und Bestimmung der Position

Die Erkennung und Positionsbestimmung eines Bildes findet in den Eigenräumen statt. Um das Bild zu erkennen, wird es in den universellen Eigenraum projiziert. Es wird dann das Objekt gesucht, dessen Mannigfaltigkeit am nächsten an dem projizierten Bild liegt. Hat man das Objekt bestimmt, so wird das Bild in den Objekteigenraum projiziert. Die Ausrichtung des Objektes wird durch die Position auf der Mannigfaltigkeit gegeben, die am nächsten an dem projizieren Bild liegt.

4.1 Abstände

Gegeben sei eine Bildmenge mit K Objekten. Man möchte ein Bild x klassifizieren. Gesucht wird die Klasse k, die bei einer gegebenen Verteilung p und gegebenen x am wahrscheinlichsten ist. Das bedeutet, es wird das k gesucht, welches die Verteilung $p(k|x)$ maximiert.

$$x \to r(x) = \underset{k}{\text{argmax}}\{p(k|x)\} \tag{18}$$

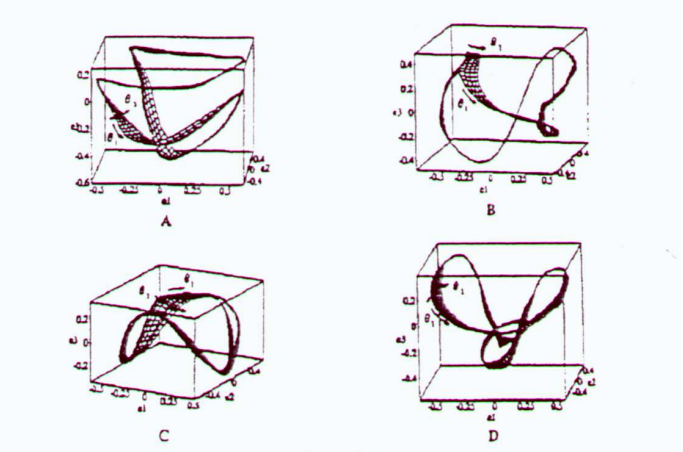

Abbildung 5: Mannigfaltigkeiten, die nur bezüglich der ersten drei wichtigsten Eigenvektoren dargestellt sind.

Der Ausdruck ist auch bekannt als die Bayes'sche Entscheidungsregel [4]. Für die Bayes'sche Entscheidungsregel ist bekannt, daß sie die Fehlerrate minimiert, falls die tatsächliche Verteilungsfunktion bekannt ist.
In dem vorgestellten Ansatz werden zwei Voraussetzungen gemacht. Die a-priori Wahrscheinlichkeit ist für alle Klassen gleich, d.h. jede Klasse ist gleich wahrscheinlich:

$$p(k) = \frac{1}{K} \tag{19}$$

Des weiteren ist $p(x|k)$ Gauß-verteilt.
Man kann die Bayes'sche Entscheidungsregel folgendermaßen umschreiben:

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmax}}\{\frac{p(k) \cdot p(x|k)}{p(x)}\} \tag{20}$$

Da $p(x)$ nicht von k abhängt, kann es bei der Berechnung vernachläßigt werden.

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmax}}\{p(k) \cdot p(x|k)\} \tag{21}$$

Durch die Annahme, daß alle Klassen gleich verteilt sind, kann $p(k)$ aus dem Term weggelassen werden.

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmax}}\{p(x|k)\} \tag{22}$$

Nach Annahme ist $p(x|k)$ Gauss-verteilt. Weiterhin wird für alle Klassen angenommen, daß die Kovarianzmatrizen gleich sind, d.h. $\Sigma_k = \Sigma$ für alle k.

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmax}}\{\exp[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)]\} \tag{23}$$

Da die Exponentialfunktion monoton ist, braucht nur der Exponent betrachtet werden.

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmax}}\{-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\} \tag{24}$$

Die Konstante $\frac{1}{2}$ spielt bei der Maximierung keine Rolle. Weiterhin wird der Ausdruck maximal, wenn $\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)$ minimal wird.

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmin}}\{(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\} \tag{25}$$

Ist Σ die Einheitsmatrix, so ergibt das den Euklidischen Abstand. Die Einheitsmatrix kann unter der Bedingung $\Sigma_k = \Sigma$ immer erreicht werden, indem die Merkmalsvektoren einer sogenannten "Whitening-Transformation" unterzogen werden. Die PCA hat hier die Eigenschaft, daß die entstandenen Merkmale unkorreliert sind [4].

$$\underset{k}{\text{argmax}}\{p(k|x)\} = \underset{k}{\text{argmin}}\{\|x - \mu_k\|^2\} \tag{26}$$

Die Erkennung kann dann auf Grund von Distanzberechnungen durchgeführt werden.
Ein Bild x_m kann durch seine Projektion angenähert werden.

$$x_m \approx \sum_{i=1}^{k} g_{m_i} e_i + c \tag{27}$$

Benutzt man in der Distanzberechnung zweier Bilder x_m und x_n diese Gleichung, so ergibt sich:

$$\begin{aligned} \|x_m - x_n\|^2 &\approx \|\sum_{i=1}^{h} g_{m_i} e_i - \sum_{i=1}^{h} g_{n_i} e_i\|^2 \\ &= \|\sum_{i=1}^{h} (g_{m_i} - g_{n_i}) e_i\|^2 \\ &= \sum_{i=1}^{h}\sum_{j=1}^{h} e_i^T e_j \cdot (g_{m_i} - g_{n_i})^2 \\ &= \|g_m - g_n\|^2 \end{aligned} \tag{28}$$

Die letzte Vereinfachung erfolgt durch die Tatsache, daß die Eigenvektoren orthogonal zueinander sind, d.h. $e_i^T e_j = 1$ wenn $i = j$, und 0 sonst. Die Distanz zweier Bilder kann also durch die Distanz ihrer Projektionen approximiert werden werden.

4.2 Erkennung

Das zu bestimmendes Bild x wird als erstes in den universellen Eigenraum projiziert.

$$x_u = [e_1, ..., e_h]^T (x - m) = V^T (x - m) \tag{29}$$

Gesucht wird das Objekt, dessen Mannigfaltigkeit am nächsten an x_u ist.

$$k^* = argmin_k \{ \min_{\Theta_1, \Theta_2} \|x_u - g^{(k)}(\Theta_1, \Theta_2)\| \} \tag{30}$$

Ist die minimale Distanz grösser als ein festgelegter Schwellwert, so wird angenommen, daß das Bild x von keinem Objekt ist, die das System gelernt hat. Liegt die Distanz unterhalb des Schwellwertes, so ist das Bild von dem Objekt k^*.
Steht das Objekt fest, so wird x in den Objekteigenraum projiziert.

$$x_o = [e_1^{(k^*)}, ..., e_h^{(k^*)}]^T (x - m^{(k^*)}) = {V^{(k^*)}}^T (x - m^{(k^*)}) \tag{31}$$

Um die Position des Bildes zu bestimmen, wird die Lage auf der Mannigfaltikeit gesucht welche am nächsten zu x_o liegt.

$$i = argmin_{\Theta_1, \Theta_2} \|x_o - f^{(k)}(\Theta_1, \Theta_2)\| \} \tag{32}$$

Die Lage auf der Mannigfaltigkeit gibt dann die Parameter des Bildes x an. Θ_1 steht für die Ausrichtung und Θ_2 für die Beleuchtungrichtung.

4.3 Suchverfahren

Für die Suche nach dem nächsten Punkt auf der Mannigfaltigkeit wurden in der betrachteten Arbeit zwei verschiedene Verfahren verwendet. Das erste Verfahren basiert auf der binären Suche im hochdimensionalen Raum, während bei dem zweite Verfahren ein dreischichtiges neuronales Netz verwendet wurde.
Das erste Verfahren ist eine binäre Suche im hochdimensionalen Raum [5]. Um das Prinzip zu veranschaulichen, wird der 3-D Raum betrachtet. Es wird eine Punktmenge bestimmt, deren Punkte innerhalb eines Würfels der breite 2ϵ liegen. Das Zentrum dieses Würfels ist das in den Eigenraum projizierte Bild. In dieser Punktmenge wird dann im Sinne des Euklidischen Abstandes der am nächsten gelegene Punkt gesucht.
Die Bestimmung der Punktmenge erfolgt folgendermaßen. Zuerst werden die Punkte bestimmt, die zwischen zwei parallelen Ebenen X_1 und X_2 liegen. Diese Ebenen haben zu dem projizierten Bildpunkt jeweils den Abstand ϵ und sind senkrecht zur x-Achse. Diese Punkte werden in eine Kandidatenliste eingetragen. Als nächstes werden die Punkte aus der Kandidatenliste gestrichen, die nicht zwischen zwei weiteren Ebenen Y_1 und Y_2 liegen. Die Ebenen sind senkrecht zur y-Achse und haben ebenfalls den Abstand ϵ zu dem projizierte Bildpunkt. Dieses wird dann für die Ebenen Z_1 und Z_2 wiederholt. In der Kandidatenliste sind jetzt nur noch die Punkte enthalten, die innerhalb des Würfels mit der Seitenlänge 2ϵ und dem projizierten Bild als Zentrum.
Da die bestimmte Punktmenge klein ist, sind die Kosten der Suche nach dem am nächsten liegenden Punkt klein. Die Hauptkosten entstehen bei der Erstellung der Kandidatenliste. Um die Kandidatenliste zu erstellen wird in [5] eine Datenstruktur vorgestellt auf der man mit 1-D binärer Suche effizient die Punkte zwischen zwei parallelen Hyperebenen bestimmen kann. Hierbei werden die Punkte in k 1-D Arrays gespeichert. k ist die Dimension des Eigenraums. Das j-te Array enthält die j-te Komponente der Punkte. Jedes Array wird sortiert, wobei nachgehalten wird, welche Stelle im sortierten Array zu welchen Punkt gehört. In dem Array mit der ersten Koordinate jeden Punktes werden die Punkte mit binärer Suche bestimmt, dessen erste Koordinate innerhalb der ϵ-Umgebung der Koordinate des projizierten Bildpunktes liegen. Diese werden in die Kandidatenliste eingetragen. Für die entstandene Kandidatenliste wird dann dieses Verfahren für die weiteren Koordinaten iteriert. Dadurch erhält man die Kadidatenliste, die die Punkte in der ϵ-Umgebung des projizierten Bildpunktes enthält. Die Kosten liegen in $O(k \log_2 n)$, wobei k die Dimension des Eigenraumes und n die Anzahl der Punkte der Mannigfaltigkeit ist.
Bei dem zweiten Verfahren wird ein neuronales Netz [6] verwendet. Dabei wird das Zuordnen der Eingabepunkte und der Parameter der Mannigfaltigkeit gelernt. Dieses Netz basiert auf der Regularisationstechnik. Das Netzwerk hat die Eigenschaft die diskreten Bildpunkte implizit zu interpolieren. Dadurch wird die Cubic Spline Interpolation zur Bestimmung der Mannigfaltigkeit bei diesem Verfahren nicht benötigt. Durch die implizite Interpolation erreicht dieses Verfahren etwas genauere Ergebnisse in der Positionsbestimmung.

5 Experiment

Die Bilder der Objekte wurden in Abhängigkeit von der Ausrichtung und der Beleuchtungsrichtung mit einer Kamera aufgenommen.

5.1 Aufbau

Für die Repräsentation des Objektes werden 2-D Bilder des Objektes benötigt. Um diese Bilder zu erhalten, wird ein Objekt auf eine motorisierte Drehplatte gestellt. Diese Platte kann sich nur um eine Achse drehen. Daher wird nur ein Parameter für die Ausrichtung betrachtet. Die Beleuchtungsrichtung wurde mit einem Roboterarm variiert. Diese wurde durch den zweiten Parameter angegeben. Von einem Objekt werden dann mit einer Kamera Bilder unter verschiedenen Ausrichtungen und Beleuchtungsrichtungen aufgenommen.

5.2 Ergebnisse

Das System wurde auf zwei Objektmengen trainiert und getestet. Die Objektmenge A enthielt 4 Objekte die gleiche Reflektionseigenschaften haben, aber deren Gestalt in gewissen Positionen sehr ähnlich sind. Die Objektmenge B enthielt 4 Objekte mit einer komplexen Erscheinungscharakteristik. Zu jeder Objektmenge wurden zwei Trainingsmengen mit unterschiedlicher Anzahl an verwendeten Ausrichtungen

Abbildung 6: Objekte, die in diesem Experiment verwendet wurden

definiert. Die Trainingsmenge I enthält für jedes Objekt 450 Bilder. Es wurden 5 Beleuchtungsrichtungen und 90 verschiedene Ausrichtungen gewählt. Die Trainingsmenge II enthält für jedes Objekt 90 Bilder. Für diese Trainingsmenge wurden wieder 5 Beleuchtungsrichtungen aber nur 18 Ausrichtungen verwendet. Die Testmengen bestanden wieder aus diesen 4 Objekten, wobei hier 3 Beleuchtungsrichtungen und 90 Positionen benutzt wurden. Die Positionen im Test waren andere als im Training.
Tabelle 1 zeigt die Testergebnisse. Die Ergebnisse stellen den durchschnittlichen absoluten Ausrichtungsfehler in Grad dar. Die beiden Objektmengen erziehlen auf der Trainingsmenge I eine durchschnittliche Abweichung von 0.5 Grad. Es zeigt, daß die Positionsbestimmung sehr genau ist. Selbst auf der Trainingsmenge II mit deutlich weniger Ausrichtungen, ist die Bestimmung der Ausrichtung nicht erheblich schlechter. Sie fällt auf ca. 1 Grad ab.

Tabelle 1: Durchschnittlicher absoluter Ausrichtungsfehler in Grad

Durchschnittlicher absoluter Ausrichtungsfehler in Grad		
	I	II
A	0.5	1.0
B	0.5	1.2

Abbildung 6 zeigt Objekte mit denen das System weiter getestet wurde. Die Erscheinung dieser Objekte wurde im 20 dimensionalen universellen Eigenraum dargestellt. Sowohl die Erkennung als auch die Positionsbestimmung fanden im universellen Eigenraum statt. Der komplette Erkennungsprozess, mit Segmentierung, Normalisierung, Projektion des Bildes in den universellen Eigenraum und die Suche nach dem nächsten Objekt und der Position, wurde in weniger als einer Sekunde auf einer Sun Workstation durchgeführt. Die Erkennung läuft also fast in Echtzeit ab. Die Bilder wurden alle richtig erkannt. Dieses Ergebnis ist subjektiv sehr gut, da wenn man sich die Objekte anschaut, man feststellt, daß einige Objekte, wie z.B. die Autos, sich sehr stark ähneln.

6 Zusammenfassung

Es wurde ein Verfahren zur erscheinungsbasierten 3-D Objekterkennung vorgestellt. Ein 3-D Objekt wird durch eine Menge von 2-D Bildern dargestellt. Es werden zwei Eigenräume mit der PCA bestimmt, in die die Bilder projiziert werden. Die projizierten Bilder eines Objektes werden mit Splines interpoliert. Die entstandene Kurve beschreibt eine Mannigfaltigkeit, auf der die Erkennung und Bestimmung der Ausrichtung stattfindet.
Das vorgestellte System hat auf den benutzten Bildern gute Ergebnisse erzielt. Es wird jedoch vorausgesetzt, daß die Bilder segmentiert sind. Des weiteren ist die geeignete Größe des Eigenraums unbekannt. Ein Objekt wurde bezüglich der zwei Parameter Ausrichtung und Beleuchtungsrichtung dargestellt. Um aber die Ausrichtung eines Objektes in drei Dimensionen zu beschreiben, benötigt man zwei Parameter für die Ausrichtung und ein Parameter für die Beleuchtungsrichtung. Obendrein ist es schwierig diesen Ansatz in einer realen Anwendung einzusetzen, da man Bilder von den Objekten benötigt, die man erkennen möchte. Man kann z.B. in dem Bereich der Gesichtserkennung, nicht jeden Menschen auf eine Drehplatte setzen und die Bilder von seinem Gesicht machen. Im Bereich der Medizin wäre dieses Verfahren z.B. bei CT Bildern einsetzbar, da diese Bilder schon drei dimensional vorliegen.

Literatur

[1] H. Murase, S.K. Nayar: Visual Learning and Recognition of 3D Objects from Appearance. International Journal of Computer Vision, 1995; 14(1): 5–24.

[2] R.O. Duda, P.E. Hart, D.G. Storck: Pattern Classification. J. Wiley, New York, 2001.

[3] T. Lehmann, W. Oberschelp, E. Pelikan, R. Repges: Bildverarbeitung für die Medizin. Springer, 1997.

[4] H. Ney: Skript zur Vorlesung: Mustererkennung und Neuronale Netze. RWTH Aachen, 2002.

[5] S.A. Nene, S.K. Nayar: A Simple Algorithm for Nearest Neighbor Search in High Dimensions. Technical Report No. CUCS-030-95, 1995.

[6] T. Poggio, F. Girosi: A Theory of Networks for Approximation and Learning. 1989.