

Probabilistisches Tracking mit dem Condensation Algorithmus

von Axel Janßen

Betreuer: Daniel Keysers

Inhalt

1	Einleitung	2
2	Stochastische Grundlagen	2
2.1	Grundlegende Begriffe	3
2.2	Das Theorem von Bayes	4
3	Der Condensation Algorithmus	4
3.1	Vorbereitungen	5
3.2	Ein Schritt des Condensation Algorithmus	5
3.2.1	Vor dem ersten Schritt	5
3.2.2	Die eigentliche Iteration	5
3.2.3	Auffinden des Objektes	6
3.3	Modellierung der Objektdynamik	7
4	Exkurs: Der Kalman Filter	7
4.1	Der Kalman Filter	8
4.2	Vergleich des Condensation Algorithmus mit dem Kalman Filter	9
5	Zusammenfassung	9

Zusammenfassung

Der Condensation Algorithmus wurde von Michael Isard und Andrew Blake in der Veröffentlichung „Condensation - Conditional Density Propagation for Visual Tracking“ [1] vorgestellt. Der Algorithmus verfolgt mit Hilfe eines probabilistischen Ansatzes, der die konkurrierende Betrachtung verschiedener Handlungshypothesen erlaubt, ein Objekt in einer Videosequenz. Das Ziel ist, auch bei komplexer Objektumgebung eine möglichst sichere Erkennung in Echtzeitverarbeitung zu erreichen. An ausgewählten Samplestellen wird eine Wahrscheinlichkeitsverteilung repräsentiert. Durch einen Vergleich der Bilddaten mit diesen Stellen wird eine fortwährende Anpassung der Wahrscheinlichkeitsverteilung erreicht.

Keywords: Condensation, Visual Tracking, Video-Sequenz, Bayes-Theorem

1 Einleitung

Sehr viele Artikel der letzten Jahre verweisen auf die Ausarbeitung [1] der Autoren Michael Isard und Andrew Blake aus dem Jahre 1998. Der in der Ausarbeitung beschriebene Condensation Algorithmus ist ein populäres Verfahren der Bildverarbeitung der letzten Jahre.

Ziel dieses Algorithmus ist die Verfolgung von Objekten (*visual tracking*) innerhalb einer Videosequenz. Gefordert wird eine hohe Verarbeitungsgeschwindigkeit, die gelieferte Videobilder bei hoher Fehlertoleranz in Echtzeit verarbeiten kann.

Mögliche Anwendungen eines solchen Verfahrens in der Medizin könnte zum Beispiel die Ultraschallanalyse des menschlichen Herzens sein. Ultraschall bietet sich hierfür aufgrund der gelieferten Echtzeitvideosequenzen an, allerdings ist der Nachteil dieser Technologie nicht von der Hand zu weisen: Die von den Ultraschallgeräten gelieferten Sequenzen sind oft mit starkem Rauschen überlagert, so daß dem untersuchenden Arzt eine sofortige Analyse der gelieferten Bilder nicht immer leicht fällt. Hier könnte der Condensation Algorithmus gute Dienste leisten: Hat der Arzt das Objekt (das menschliche Herz) erst einmal detektiert, kann der Algorithmus während der folgenden Untersuchung das Herz permanent verfolgen und das Ergebnis seiner Analyse mit einer Umrandung sichtbar machen.

Grundsätzlich ist die erfolgreiche Verfolgung eines Objektes innerhalb eines Videos von verschiedenen Parametern abhängig. Zwei Arten von Fehlerquellen können das Ergebnis verfälschen oder den Algorithmus sogar überfordern: Veränderungen des Objektes und Veränderungen der Umgebung bzw. des Hintergrundes. Im ersten Fall kann sich das Objekt in Position und Lage und in Beschaffenheit und Form von Bild zu Bild auf teilweise nur sehr schwer vorhersagbare Art ändern; man denke beispielsweise an die Flugbewegung einer Vogelfeder bei starkem Wind. Auch könnte das gesuchte Objekt in manchen Bildern durch andere Objekte teilweise überlagert sein oder in einer Teilsequenz komplett verschwinden. Im zweiten Fall stellen ein wechselnder Hintergrund (man denke an die Personenverfolgung, bei der die Erkenntnisse des Algorithmus an den Steuermotor einer beobachtenden Kamera weitergegeben werden) oder eine dem Objekt sehr ähnliche Umgebung (beispielsweise die Verfolgung eines Blattes im Wind) Schwierigkeiten dar, die den Algorithmus ebenfalls überfordern können.

Der Condensation Algorithmus verfolgt im Gegensatz zu anderen populären Verfahren einen probabilistischen Ansatz, der auf einer diskreten Menge von Zuständen operiert: Um das Objekt im aktuellen Bild detektieren zu können, werden die in den letzten Bildern über das Objekt gewonnene Erkenntnisse genutzt.

Wie dies vonstatten geht, wird in den folgenden Abschnitten näher vorgestellt. Zunächst werden in Abschnitt 2 die relevanten stochastischen Grundlagen erläutert und wieso dem Theorem von Bayes eine zentrale Rolle bezüglich der Idee des Condensation Algorithmus zukommt. Anschließend wird in Abschnitt 3 der Algorithmus selbst vorgestellt. In einem kurzem Exkurs (Abschnitt 4) wird auf den Kalman Filter, einen anderen Ansatz aus dem Gebiet der Bildverarbeitung, eingegangen. Hier werden auch Unterschiede der beiden Verfahren deutlich gemacht.

2 Stochastische Grundlagen

Dieser Abschnitt behandelt die stochastischen Grundlagen, die zum Verständnis des Condensation Algorithmus notwendig sind. Der erste Teil (Abschnitt 2.1) beschäftigt sich mit den relevanten Begriffen der Wahrscheinlichkeitsrechnung und führt anhand von Beispielen in diesen Bereich ein. Der zweite Teil



Abbildung 1: Mit *Splines* umrandetes Herz (Ultraschallaufnahme).

(Abschnitt 2.2) erklärt das Theorem von Bayes und stellt einen Zusammenhang des Theorems mit der dem Condensation Algorithmus zugrunde liegenden Idee her.

2.1 Grundlegende Begriffe

Für das Rechnen mit Wahrscheinlichkeiten wird im Allgemeinen das Kolmogorowsche Axiomensystem verwendet. Im Folgenden seien hier Eigenschaften von Wahrscheinlichkeiten für zwei Ereignisse a und b aufgeführt:

- Wahrscheinlichkeiten liegen immer zwischen 0 und 1:
 $0 \leq P(a) \leq 1$
- Die Summe der Wahrscheinlichkeiten der Ereignisse ist 1:
 $\sum_k P(a_k) = 1$
- Das komplementäre Ereignis zu a ist genau dann realisiert, wenn a nicht realisiert ist:
 $P(\bar{a}) = 1 - P(a)$

Eine *Zufallsvariable* repräsentiert ein Ereignis, deren Eintrittswahrscheinlichkeit bestimmt werden soll oder deren Wahrscheinlichkeit schon gegeben ist. Sie kann dabei vorgegebene Werte aus einem vorgegebenen Definitionsbereich annehmen.

Bisher wurden ausschließlich einfache Wahrscheinlichkeiten betrachtet, also Aussagen „wie groß ist die Wahrscheinlichkeit, daß ein Ereignis x auftritt?“ Der Begriff *a-priori Wahrscheinlichkeit* meint diese



Abbildung 2: Mit *Splines* umrandetes Blatt eines Busches.

unbedingte Wahrscheinlichkeit eines Ereignisses a : $P(a)$. Hier sei beispielhaft das Würfeln mit einem Würfel aufgeführt: $P(\text{Sechser beim Würfeln}) = 1/6$.

Von zentraler Bedeutung dagegen ist in diesem Fall der Begriff der *a-posteriori Wahrscheinlichkeit*, der die *bedingte* Wahrscheinlichkeit eines Ereignisses a unter Beobachtung von b (also wenn b gilt): $P(a|b)$: $P(\text{Sechser beim Würfeln} | \text{Gerade Augenzahl}) = 1/3$ meint, also Aussagen der Form: „Wie groß ist die Wahrscheinlichkeit, daß Ereignis x auftritt, unter Bedingung von y ?“

2.2 Das Theorem von Bayes

Die Gleichung für bedingte Wahrscheinlichkeiten besagt:

$$P(x|y) = \frac{P(x \wedge y)}{P(y)} = \frac{P(y|x) \cdot P(x)}{P(y)} \quad (1)$$

Das folgende Beispiel erläutert diese Gleichung näher:

- x : Hypothese (es ist sonnig)
- y : Daten, Ursache (es sind 25 Grad Celsius)
- $P(y)$: Wahrscheinlichkeit, daß die Temperatur mindestens 25 Grad beträgt
- $P(x)$: Wahrscheinlichkeit, daß es sonnig ist.
- $P(x|y)$: Wahrscheinlichkeit, daß es sonnig ist, unter der Bedingung, daß die Temperatur mindestens 25 Grad beträgt.
- $P(y|x)$: Wahrscheinlichkeit, daß es an einem sonnigen Tag mindestens 25 Grad warm ist.
- $P(x \wedge y) = P(y|x) \cdot P(x)$
- $P(y|x) \cdot P(x) = P(x|y) \cdot P(y)$

Für die Berechnungen bezüglich des Condensation Algorithmus gelten im folgenden diese Variablen:

- x sei eine mögliche Position des Objektes (Hypothese, *Sample*)
- y sei die gelieferte Bildinformation

Das Theorem von Bayes wird nun benötigt, um die schwierige direkte Bestimmung der Objektposition $P(x|y)$ („befindet sich das Objekt an dieser Stelle im Bild“) durch die einfachere Bestimmung, ob die Bildinformation der vermuteten Position entspricht $P(y|x)$ („entspricht die Bildinformation der Hypothese?“) auszudrücken.

$$P(x|y) = \frac{P(y|x) \cdot P(x)}{P(y)} = k \cdot P(y|x) \cdot P(x) \quad (2)$$

Der Quotient $\frac{1}{P(y)}$ ist unabhängig von x und wird daher durch die Konstante k ausgedrückt. Die Berechnung von $P(x|y)$ durch $P(y|x)$ stellt die zentrale Idee des Condensation Algorithmus dar.

Im folgenden Abschnitt 3 soll nun erklärt werden, wie aus den Hypothesen über die Position eines Objektes in einem bestimmten Bild der Videosequenz $P(x_{t-1})$ Aussagen über die Wahrscheinlichkeit, die Position des Objektes im nächsten Bild der Videosequenz $P(x_t)$ betreffend, gewonnen werden können.

3 Der Condensation Algorithmus

Das Ziel des Condensation Algorithmus ist die möglichst korrekte Identifizierung eines Objektes in jedem Zeitschritt einer Videosequenz und dies möglichst in Echtzeit. Der Condensation Algorithmus basiert, wie schon erwähnt, auf einem probabilistischen Ansatz, der auf einer diskreten Menge von Zuständen operiert, um anhand von gewichteten Hypothesen (im folgenden als *Samples* bezeichnet) Vorhersagen über das Verhalten des beobachteten Objektes im nächsten Zeitschritt machen zu können. Im folgenden soll nun dargestellt werden, was dies genau bedeutet, und wie der Algorithmus abläuft.

3.1 Vorbereitungen

Ein Objekt in einem Bild wird im einfachsten Fall durch seine Position beschrieben. Aber auch andere Informationen können betrachtet werden, wie die Lage im Raum, Form des Objektes etc. Im folgenden werden diese Variablen wie angegeben verwendet:

- x_t ist der Zustand des modellierten Objektes im Zeitschritt t
- y_{t-1} beschreibt den vorangehenden Zeitschritt (das vorherige Bild der Videosequenz)
- y_t beschreibt den aktuellen Zeitschritt (das aktuelle Bild der Videosequenz).

Nach jedem Zeitschritt gibt der Algorithmus eine gewichtete Menge von Hypothesen (im folgenden als *Sample-Set* bezeichnet) $\{x_t^{(1)}, \pi_t^{(1)}, \dots, x_t^{(N)}, \pi_t^{(N)}\}$ aus, die die bedingte Zustandsdichte $p(x_t|y_t)$ in Zeitschritt t repräsentieren. Ein Sample n zum Zeitschritt t ist allgemein mit $x_t^{(n)}$ bezeichnet und $\pi_t^{(n)}$ meint die Gewichtung eines solchen Samples. Der folgende Abschnitt erklärt, wie man an dieses Sample-Set gelangt.

3.2 Ein Schritt des Condensation Algorithmus

3.2.1 Vor dem ersten Schritt

Für jeden Zeitschritt t der Videosequenz wird eine Iteration gestartet, um an das vorgenannte Sample-Set zu gelangen. Die Iteration jedes Zeitschrittes benötigt dazu das Sample-Set aus dem vorherigen Zeitschritt.

Also ist klar, daß vor der ersten Iteration bereits eine Anfangsverteilung für Samples erzeugt werden und ein Sample-Set $\{x_0^{(n)}\}$ vorliegen muß. Eine Menge N von Samples, also Hypothesen möglicher Zustände des Objektes, wird im Bild verteilt (Schritt 0 in Abbildung 3). Dies kann durch eine Gleichverteilung geschehen. Hat man allerdings schon eine Vermutung, wo sich das Objekt befinden könnte, kann man die Samples in dieser Bildregion mit einer geeigneten Verteilung $P(x)$ enger streuen. Weil man von vornherein die Menge von Samples auf N beschränkt, kann ein Ablauf innerhalb gegebener Rechenressourcen garantiert werden.

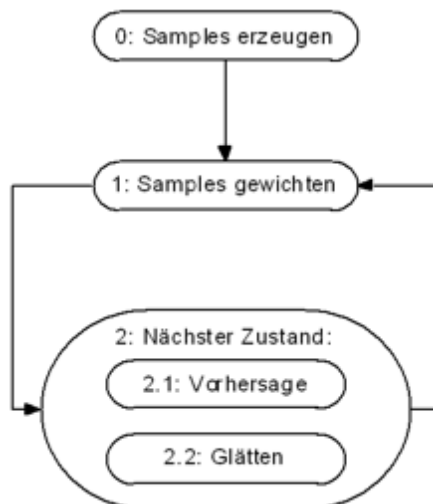


Abbildung 3: Schematischer Ablauf des Condensation Algorithmus.

3.2.2 Die eigentliche Iteration

An dieser Stelle beginnt nun der erste Schritt der Abbildung 4. Er ist mit $P(x_{t-1})$ bezeichnet. Abbildung 4 beschreibt die Iteration des Condensation Algorithmus zu einem beliebigen Zeitpunkt t . Es soll nun aus dem Sample-Set des letzten Zeitschrittes $t - 1$ ein neues Sample-Set für den aktuellen Zeitschritt t gewonnen werden.

Die schraffierten Kreise stellen die Samples zum Zeitpunktschritt $t - 1$ dar. Der Mittelpunkt eines Kreises meint das jeweilige Sample und die Größe der schraffierten Fläche um diesen Mittelpunkt gibt die Gewichtung des Samples an.

Der erste Schritt in Abbildung 4, das *Sampling*, geht folgendermaßen vonstatten: Aus der bisherigen Samplemenge $\{x_{t-1}^{(1)}, \pi_{t-1}^{(1)}, \dots, x_{t-1}^{(N)}, \pi_{t-1}^{(N)}\}$ des letzten Zeitschrittes werden N Samples $x_{t-1}^{(n)}$ entnommen. Man kann dies durch wiederholtes Ziehen von Samples (mit Zurücklegen) erreichen. Einige Samples, vor allem diejenigen mit höherer Gewichtung $\pi_{t-1}^{(n)}$, werden häufiger gezogen, wohingegen manche Samples vielleicht gar nicht gezogen werden. Dies läuft schematisch folgendermaßen ab:

1. Ziehe ein Sample $x_{t-1}^{(n)}$.
2. Übernehme Sample $x_{t-1}^{(n)}$ mit der assoziierten Wahrscheinlichkeit $\pi_{t-1}^{(n)}$ in das neue Sample-Set.
3. Gehe solange zu Schritt 1 zurück, bis die Anzahl der Ziehungen gleich der Anzahl N Elemente der alten Samplemenge ist.

Die Sampleverteilung entspricht nun wieder der gesuchten Wahrscheinlichkeitsverteilung $P(x_{t-1})$. Jedes Element des neuen Sample-Sets wird nun dem nächsten Schritt unterzogen: Das Ziel ist ja die Bewegung eines Objektes nachzuvollziehen. Dazu muß auch die Bewegung probabilistisch erfaßt werden, was mit dem Modell $P(x_t|x_{t-1})$ in Schritt 2.1 geschieht. Zunächst wird jedes Element der neuen Samplemenge anhand eines vorhergesagten Richtungsvektors bewegt.

$$P(x_t) \leftarrow P(x_t|x_{t-1}) \quad (3)$$

Identische Elemente des neuen Sample-Sets werden identisch bewegt. Dies wird durch die Pfeile in Abbildung 4, die von der ersten Samplemenge ausgehen, angezeigt. Man erhält die neue Verteilung $P(x_t)$. Im nächsten Schritt (Schritt 2.2) wird Rauschen hinzugefügt, um identische Elemente zu trennen.

An dieser Stelle im Algorithmus ist die neue Samplemenge komplett, allerdings sind die Samples nicht gewichtet, was nun anhand der gelieferten Bildinformation geschieht. Ein Bildanalysealgorithmus trifft also Aussagen dazu, wie hoch die Wahrscheinlichkeit ist, an einer gegebenen Stelle x das Objekt auch wirklich in dem beschriebenen Zustand zu finden. Dies wird für alle N Samples durchgeführt.

$$\pi_t \leftarrow P(x_t) \leftarrow P(x_t|y_t) = k \cdot P(y_t|x_t) \cdot P(x_t) \quad (4)$$

Nach Normalisierung der Gewichte $\pi_t^{(n)}$ (es gilt also $\sum_n \pi_t^{(n)} = 1$) erhält man ein N -elementiges, gewichtetes Sample-Set des Zeitschrittes y_t . Dieses Sample-Set $\{x_t^{(n)}, \pi_t^{(n)}, n = 1, \dots, N\}$ repräsentiert die a-posteriori Wahrscheinlichkeit $P(x_t|y_t)$.

Am Ende dieses Schrittes befindet man sich wieder am oberen Ende der Abbildung 4, allerdings im nächsten Zeitschritt; der Algorithmus startet die nächste Iteration.

3.2.3 Auffinden des Objektes

Es kann zu jedem Zeitschritt, nachdem das aktuelle Sample-Set gebildet wurde, die wahrscheinlichste Position des Objektes gefunden werden: In der Nähe dieser Position werden die meisten Samples verteilt sein. Man kann nun anhand von verschiedenen Methoden, wobei sich eine globale Durchschnittsbildung oder „feiner“, eine Durchschnittsbildung über lokalen Zentren anbietet, aus den verschiedenen Samples, die wahrscheinlichste Objektposition schätzen mit

$$\varepsilon[f(x_t)] = \sum_{n=1}^N \pi_t^{(n)} \cdot f(x_t^{(n)}), \quad (5)$$

wobei man eine geeignete $f(x)$ verwendet, beispielsweise $f(x) = x$. Dieser Vorgang ist in Abbildung 5 anhand des Kopfes eines Kindes graphisch dargestellt.

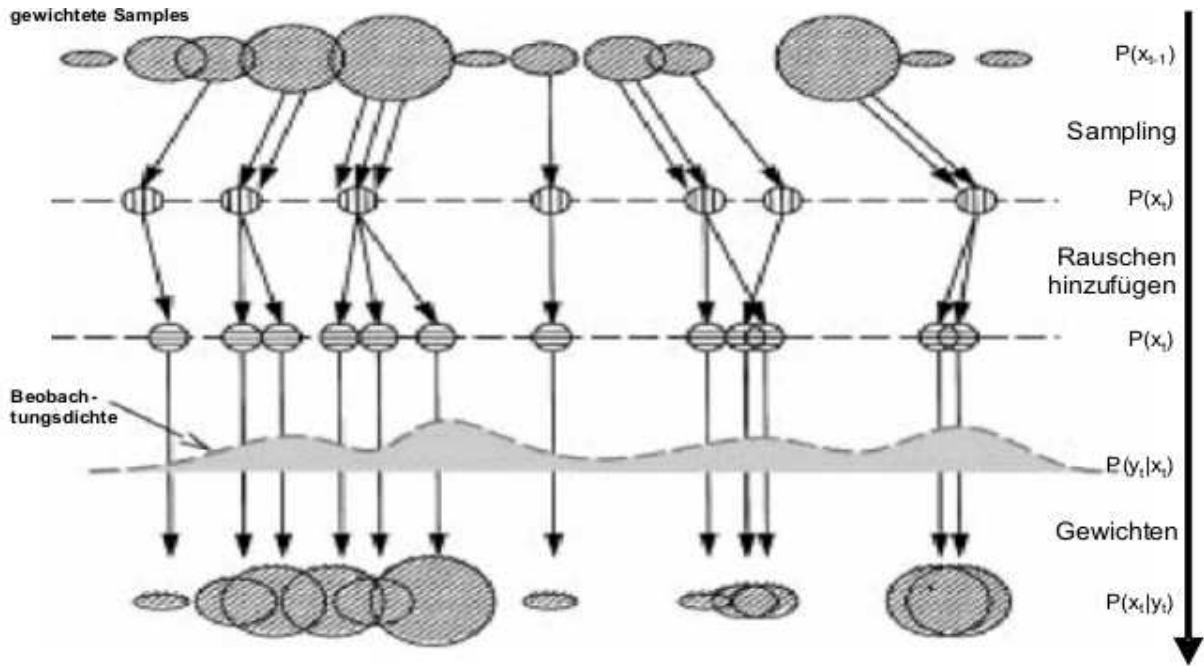


Abbildung 4: Bildhafter Ablauf des Condensation Algorithmus. Jeder der drei Schritte *Sampling* - *Rauschen hinzufügen* - *Gewichten* findet sich in der Iteration des Algorithmus wieder.

3.3 Modellierung der Objektdynamik

In diesem Abschnitt wird kurz erläutert, wie die Objektdynamik modelliert wird. Die Entwickler des Condensation Algorithmus führen folgende lineare Differentialgleichung zweiter Ordnung an:

$$x_t - \bar{x} = A \cdot (x_{t-1} - \bar{x}) + B \cdot w_t. \quad (6)$$

\bar{x} ist der Durchschnittswert des Zustands, A und B sind Matrizen, die die deterministischen und stochastischen Komponenten des dynamischen Modells repräsentieren und w_t ist eine Variable, die das Rauschen ausdrückt. Gleichung 7 beschreibt folgende Wahrscheinlichkeit $P(x_t|x_{t-1})$:

$$P(x_t|x_{t-1}) \propto \exp -\frac{1}{2} \| B^{-1} \cdot ((x_t - \bar{x}) - A \cdot (x_{t-1} - \bar{x})) \|^2 \quad (7)$$

Hier sind $A \cdot (x_{t-1} - \bar{x}_t)$ als die Vorhersage des neuen Zustands und $x_t - \bar{x}$ als der neue Zustand zu verstehen, wobei jeweils die Differenz mit dem Mittelwert \bar{x} gebildet wird. Ist die Differenz der beiden Werte Null, also $(x_t - \bar{x}) - A \cdot (x_{t-1} - \bar{x}_t) = 0$, stimmt die Vorhersage genau. B^{-1} gewichtet den Term in Bezug auf das Rauschen. Je größer nun der Abstand zwischen Vorhersage und neuem Zustand ist, desto kleiner ist die Wahrscheinlichkeit $P(x_t|x_{t-1})$. Es empfiehlt sich, die Parameter A, \bar{x} und B des dynamischen Modells je nach Anwendung anzupassen, beispielsweise während das Objekt typische Bewegungen in einer Trainingsphase für den Algorithmus ausführt.

4 Exkurs: Der Kalman Filter

Die Entwickler des Condensation Algorithmus vergleichen eine Anwendung ihres Algorithmus mit dem sehr populären und etablierten Kalman Filter, der 1960 von Rudolph E. Kalman entwickelt wurde. Sie wollen die Robustheit des Condensation Algorithmus bezüglich komplexer Bildhintergründe zeigen. Der erste Teil (4.1) dieses Abschnitts stellt den Kalman Filter kurz vor, während der zweite Teil (4.2) des Abschnitts näher auf die Versuchsparameter, die Versuchsdurchführung und die Ergebnisse eingeht.

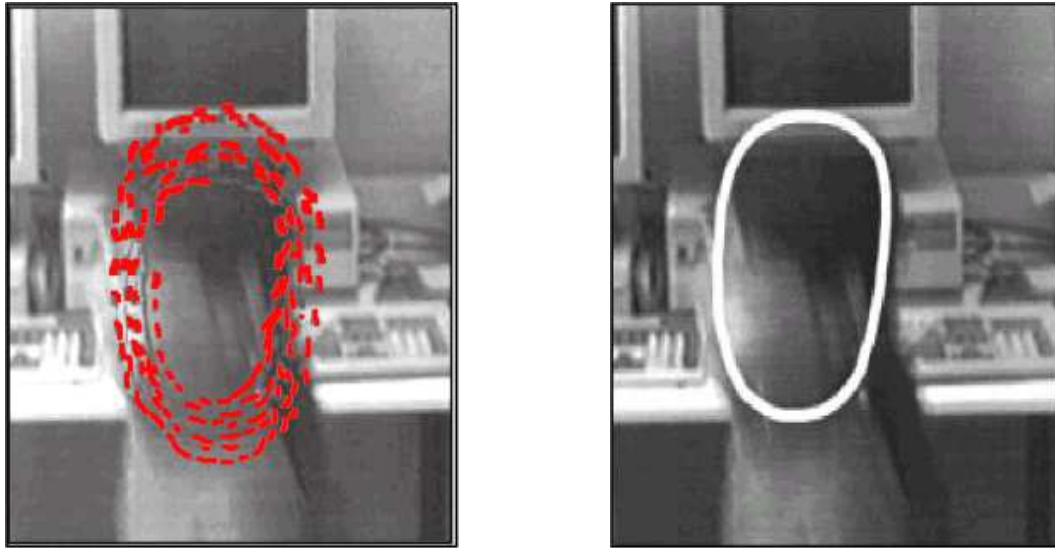


Abbildung 5: Eine Menge an Hypothesen (Sample-Set, links). Wahrscheinlichste Position des Objektes (rechts).

4.1 Der Kalman Filter

Der Kalman Filter löst das allgemeine Problem der Schätzung und Vorhersage des zukünftigen Verhaltens eines Prozesses, der als lineares, dynamisches System modelliert ist [5],[6],[7]. Die Untersuchung und Vorhersage des Prozesses erfolgt diskret über die Zeit und rekursiv. Bei jedem Iterationsschritt wird der Zustand des Prozesses neu berechnet, wobei die Matrizen, die die Historie des Prozesses, die Varianzen und die Kovarianzen der Prozeßvariablen beschreiben, aktualisiert werden. Man nimmt eine dem Prozeßrauschen zugrundeliegende Gaußverteilung an. Der Kalman Filter bietet eine geschlossene Lösung. Die folgenden beiden Gleichungen werden als Grundlage zur Beschreibung eines Prozesses angenommen:

$$x_t = A_t \cdot x_{t-1} + w_t \quad (8)$$

$$z_t = H_t \cdot x_t + v_t \quad (9)$$

x_t ist der Zustandsvektor, also beispielsweise Position oder Geschwindigkeit des Objektes. Er setzt sich zusammen aus der Prozeßmatrix A , die den vorherigen Zustandsvektor x_{t-1} gewichtet und dem Rauschen des Prozesses w_t . Der Meßvektor z_t besteht aus der Meßmatrix H , die den Meßvektor gewichtet und dem Meßrauschen v_t . Die erste Gleichung bestimmt die Entwicklung des Zustands über die Zeit, die zweite Gleichung bringt Zustand und Abmessung in Bezug zueinander.

Um einen so modellierten Prozeß nun mathematisch beschreiben bzw. Aussagen über den Prozeß oder sein zukünftiges Verhalten machen zu können, muß man den Zustand x_t des Systems bestimmen können, was aber nicht direkt möglich ist, da die (unbekannten) Rauschen w_t und v_t vorhanden sind. Diese werden als Zufallsprozesse angenommen, die gaußverteilt sind.

Der Algorithmus besteht aus zwei Schritten, dem *time update* Schritt mit den Gleichungen

$$x := A \cdot x \quad (10)$$

$$P := A \cdot P \cdot A^T + Q \quad (11)$$

und dem *measurement update* Schritt mit den Gleichungen

$$K_1 := H \cdot P \cdot H^T + R \quad (12)$$

$$K := P \cdot H^T \cdot K_1^{-1} \quad (13)$$

$$x := x + K \cdot (z - H \cdot x) \quad (14)$$

$$P := (I - K \cdot H) \cdot P. \quad (15)$$

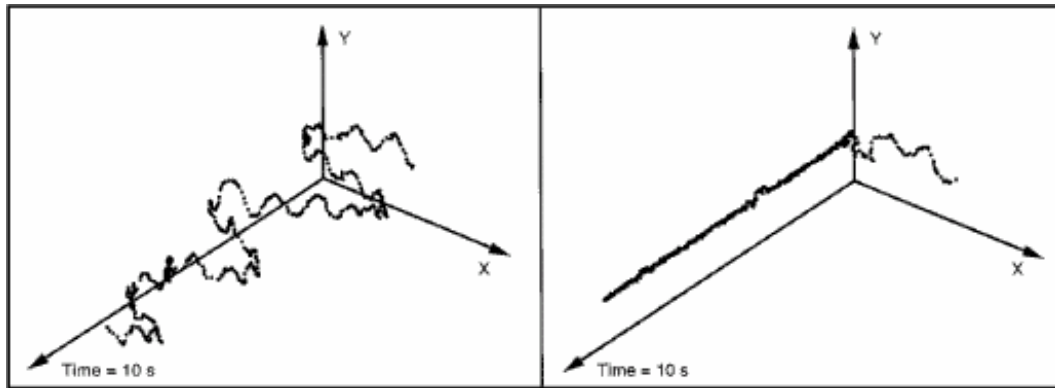


Abbildung 6: Darstellung der Verfolgung eines Objektes als Koordinatensystemplot (Position des Objektes (x- und y-Achse) im Bild zu einem bestimmten Zeitschritt (z-Achse)). Der Condensation Algorithmus (links). Der Kalman Filter (rechts).

Im *time update* Schritt werden der a priori Schätzwert x (Gleichung 10) für den Zustand des nächsten Zeitschrittes und die Fehlerkovarianzmatrix P (Gleichung 11) berechnet, im *measurement update* Schritt wird der neue Meßwert x in die Berechnung einbezogen, die Kalman-Gain berechnet (Gleichungen 12 und 13), der a posteriori Schätzwert (Gleichung 14) und die Fehlerkovarianzmatrix P (Gleichung 15) bestimmt. Der endgültige Algorithmus beschreibt dann die geschlossene Lösung des Prozesses.

4.2 Vergleich des Condensation Algorithmus mit dem Kalman Filter

Die Entwickler vergleichen eine Anwendung des Condensation Algorithmus mit dem Kalman Filter. Sie machen dies anhand eines zehn Sekunden dauernden Videobeispiels (500 Einzelbilder), in dem ein Kind unruhig vor einem komplexen Hintergrund hin- und heränzelt. Die Aufgabe ist die Verfolgung des Kopfes des Kindes, der durch einen *Shape-Space* beschrieben wird, der auf einer handgemalten Umrandung des Kopfes basiert.

Dem Bewegungsmodell zugrunde liegen zwei aufeinander aufbauende Trainingssequenzen, in denen sich das Kind vor einem einfarbigen Hintergrund bewegt. Dieses Modell wird durch einen Kalman Filter erstellt, der den Kopf des Kindes in der ersten Trainingssequenz zunächst 2,8 Sekunden (140 Einzelbilder) lang korrekt verfolgt. Anhand der aus dieser ersten Sequenz gewonnenen Informationen gelingt es dem Kalman Filter, in einer weiteren Trainingssequenz, den Kopf des Kindes 16 Sekunden (800 Einzelbilder) lang korrekt zu verfolgen.

Der Vergleichstest findet nun mit dem tanzenden Kind vor dem schon erwähnten komplexen Hintergrund statt, wobei sowohl dem Condensation Algorithmus als auch dem Kalman Filter das aus der 16 Sekunden dauernden Trainingssequenz gewonnene Bewegungsmodell zugrunde liegt.

Hier zeigt sich - so die Autoren - die Robustheit des Condensation Algorithmus bezüglich komplexer Bilder. Obwohl der Algorithmus nur auf einer Sampledichte von $N = 100$ operiert, gelingt eine korrekte Verfolgung des Kopfes über die gesamten zehn Sekunden (Abbildung 5). Der Kalman Filter verliert den Kopf des Kindes nach 0,8 Sekunden (40 Bildern) und ist bis zum Ende der Videosequenz nicht zu einer Wiedererkennung des gesuchten Objekts im Stande.

Offenbar ist in diesem Fall die Eigenschaft des Condensation Algorithmus, verschiedene Hypothesen gleichzeitig verfolgen zu können, dem Kalman Filter (der ja auf der Annahme einer Gauß-Verteilung operiert und eine geschlossene Lösung bietet) überlegen.

5 Zusammenfassung

Der Condensation Algorithmus zielt auf die Verfolgung eines Objektes in einer Videosequenz ab. Dazu wird ein probabilistischer Ansatz benutzt, der sich des Theorems von Bayes bedient und auf einer diskreten Menge an Hypothesen operiert. Samplestellen innerhalb eines Bildes repräsentieren eine Wahrscheinlichkeitsverteilung $P(x)$. Durch den Vergleich mit den aktuellen Bildinformationen wird diese Verteilung gewichtet und fortwährend angepaßt. Das Theorem von Bayes dient dazu, die schwierige Aufgabe, die

direkte Bestimmung der Objektposition im Bild, durch die einfachere Aufgabe, ob sich das Objekt in einem vermuteten Zustand befindet, auszudrücken. Je nach Anwendung können sehr verschiedene Ansätze verwendet werden, um den Zustand des gesuchten Objekts zu beschreiben. Isard und Blake bedienen sich in ihrer Arbeit der Shape-Spaces, aber auch andere Ansätze sind denkbar.

Man muß bei der Implementierung des Algorithmus einen Kompromiß zwischen Echtzeitanalyse und Genauigkeit eingehen: Je genauer die Wahrscheinlichkeitsverteilung ist (je mehr Samples man im Bild verteilen läßt), um so rechenintensiver ist die Analyse. Die Autoren führen in einigen Videobeispielen auf, daß oft schon eine Zahl von 100 Samples genügt, um ein für Echtzeitanalysen hinreichend genaues Ergebnis zu erzielen. Leider läßt sich dies nicht immer ohne weiteres an den Videobeispielen erkennen. Die Vermutung, daß die Autoren aufsehenerregende Videosequenzen (beispielsweise die Bewegung eines Strauchblattes im Wind) so lange mit unterschiedlichen Parametern getestet haben, bis sie eine für diese Anwendung genügende Genauigkeit erreicht haben, läßt sich oft nicht vollständig von der Hand weisen, da eine unabhängige Testquelle nicht genannt wird.

Da der Condensation Algorithmus aber in seinen Komponenten leicht verändert (beispielsweise das zugrunde liegende Bewegungsmodell) und so den Erfordernissen bestimmter Anwendungen angepaßt werden kann, scheint er ein guter Ansatz für die verschiedenen Problemstellungen der Objektverfolgung zu sein.

Literatur

- [1] Isard M, Blake A: Condensation - Conditional Density Propagation for Visual Tracking. International Journal of Computer Vision, 1998; 29(1):5-28.
- [2] Isard M: The Condensation Algorithm Home Page. University of Edinburgh, Division of Informatics, Sept. 1998; http://www.dai.ed.ac.uk/CVonline/LOCAL_COPIES/ISARD1/condensation.html
- [3] Fisher RB: CVonline - Compendium of Computer Vision. Division of Informatics University of Edinburgh, Juli 2002; <http://www.dai.ed.ac.uk/CVonline>
- [4] Denzler J: Probabilistische Folgenmodellierung. Universität Erlangen - Nürnberg, Juli 2001; <http://www5.informatik.uni-erlangen.de/lectures/SS01/LectureDescription/profo/archive.htm>
- [5] Guézic A: Tracking Pitches for Broadcast Television; Computer; 35(3):38-43, März 2002
- [6] Welch G, Bishop G: An Introduction to the Kalman Filter. University of North Carolina - Chapel Hill, Department of Computer Science, 1998; <http://www.informatik.uni-bonn.de/III/lehre/seminare/Mustererkennung/WS99/kalman.pdf>
- [7] Welch G: The Kalman Filter Home Page. University of North Carolina - Chapel Hill, Department of Computer Science, 2002; <http://www.cs.unc.edu/~welch/kalman/>