

Kapitel 5

Ein integrierter Ansatz zur Klassifizierung und Segmentierung mit pseudo zweidimensionalen Hidden-Markov-Modellen

Dieses Kapitel beschreibt ein neuartiges Verfahren, mit dem zweidimensionale Muster in komplexen Umgebungen aufgefunden und klassifiziert werden können. Durch das Einbringen von an die Umgebung adaptierten Zuständen in die in Kapitel 4 vorgestellten pseudo zweidimensionalen Hidden-Markov-Modelle, kann die Segmentierung und Klassifizierung auf integrierte Weise erfolgen. Nach einem Training der Muster-HMMs und der Umgebungszustände kann durch die vom Viterbi-Algorithmus festgelegte Merkmal-Zustandszuordnung eine Segmentierung der Bildszene erfolgen. Gleichzeitig gibt der Viterbi-Algorithmus einen Schätzwert für die Produktionswahrscheinlichkeit des Musters aus, mit dem eine Klassenzugehörigkeit ermittelt werden kann. Bevor dieses Verfahren detailliert vorgestellt wird, gehen die folgenden beiden Unterkapitel zunächst auf die zweidimensionale Musterklassifikation bzw. die rotationsinvariante Modellierung von Objektformen mit P2DHMMs ein.

5.1 Klassifizierung von Bildern mit P2DHMMs

Die Klassifizierung von Mustern mit P2DHMMs soll hier kurz am Beispiel der Erkennung von Gesichtern dargestellt werden. Die Hidden-Markov-Modelle sind in das Gebiet der automatischen Gesichtsklassifikation von Samaria von der Cambridge University eingeführt worden (siehe [Sam94b]). Zunächst wurde die eindimensionale, aus der Spracherkennung bekannte Variante verwendet und später die pseudo zweidimensionale, welche eine signifikante Fehlerreduktion erreichte ([Sam94a]).

Die Klassifizierung von Gesichtern mit P2DHMMs erfolgt durch die folgenden beiden Schritte: Der erste Schritt, nämlich die Merkmalextraktion, basiert auf der Diskreten-Cosinus-Transformation (DCT). Ein gegebenes Bild wird mit einem Abtastfenster, entsprechend dem in Abb. 4.3 illustrierten Schema, von oben nach unten und links nach rechts abgetastet. Die Bildpunkte in dem Abtastfenster mit der festen Größe $N \times N$ werden durch die Anwendung der DCT transformiert. Die Transformationsvorschrift lautet (siehe auch [Gon92]):

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right) \quad (5.1)$$

für $u, v = 0, \dots, N-1$

Die Werte für $\alpha(u)$ ergeben sich aus

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{für } u = 0 \\ \sqrt{\frac{2}{N}} & \text{für } u = 1, \dots, N-1 \end{cases} \quad (5.2)$$

Es hat sich anhand von experimentellen Ergebnissen gezeigt, daß es vorteilhaft ist, nicht alle Koeffizienten zu verwenden, sondern die 15 Koeffizienten mit dem höchsten Betrag auszuwählen. Dies sind die Koeffizienten $C(u, v)$ für die die Bedingung $(u + v \leq 4)$ gilt. Ebenfalls als vorteilhaft hat sich die Verwendung einer Blocküberlappung von 75% in beiden Bilddimensionen erwiesen. Diese Überlappung benachbarter Abtastblöcke ermöglicht die Modellierung von Kontext und ist vergleichbar mit den sog. Delta-Merkmalen in der automatischen Spracherkennung. Die Experimente, die zu dieser Konfiguration geführt haben, sind in den Arbeiten [Eic99b] und [Eic00] dokumentiert.

Der zweite Schritt ist die statistische Klassifikation mit den pseudo zweidimensionalen Hidden-Markov-Modellen. Jeweils ein P2DHMM mit einer Struktur wie sie in Abb. 4.3 dargestellt ist, wird für jede Person in der Trainingsdatenbasis mit dem Baum-Welch-Algorithmus trainiert. Die Klassifikation erfolgt dann durch die Maximum-Likelihood-Entscheidung auf Basis der mit dem Viterbi-Algorithmus ermittelten Produktionswahrscheinlichkeiten. Das zu testende Bild wird als diejenige Person erkannt, deren zugeordnetes Modell die höchste Produktionswahrscheinlichkeit ausgibt (siehe auch Gleichung 2.12).

Experimente wurden mit diesem Verfahren auf der Olivetti Research Laboratory (ORL) Datenbasis durchgeführt und sind in [Eic00] und [Eic99b] beschrieben. Die ORL Datenbasis besteht aus je zehn Frontalansichten von 40 verschiedenen Personen, wobei die Aufnahmen sowohl Beleuchtungsvariationen als auch Variationen in der Mimik aufweisen. Die Abbildungen 4.3 und 5.1 zeigen zwei Beispielbilder aus dieser Datenbasis. Diese Datenbasis wurde in der Literatur intensiv eingesetzt (siehe z.B. [Sam94a, Sam94b] und [Che95]) und es wurden mit verschiedenen Klassifizierungsmethoden Erkennungsgenauigkeiten zwischen 80 und 99,5% erreicht. Die beschriebene Methode, die auf DCT-Koeffizienten und P2DHMMs basiert, erreichte eine Genauigkeit von 100% ([Eic99b]).

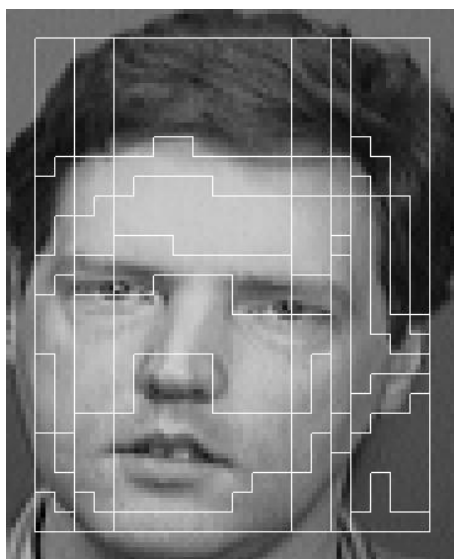


Abbildung 5.1: Zuordnung von Merkmalen und Modellzuständen durch den Viterbi-Algorithmus (aus [Eic00])

Obwohl diese optimale Erkennungsgenauigkeit erreicht wurde, so kann die Aufgabe der automatischen Gesichtserkennung für reale Anwendungen nicht als gelöst gelten. Bei der ORL-Datenbasis sind die zu klassifizierenden Bilder, wie auch die Testbilder, sowohl geeignet skaliert, als auch optimal ausgeschnitten. In Abb. 5.1 läßt sich erkennen, daß das Gesicht (inklusive der Haare) nahezu das gesamte Bild ausfüllt. Dies ist für alle Bilder dieser Datenbasis der Fall. Somit entfällt der Prozeß des Auffindens des Gesichts in dem vorgegebenen Bild. Durch diesen Auffindungsprozeß und die damit verbundene mögliche Fehlsegmentierung ergeben sich im Klassifikationsschritt zusätzliche Fehler.

Kapitel 5.3 beschreibt eine im Rahmen dieser Arbeit entwickelte Modellierungstechnik, die es ermöglicht, das Auffinden und die Klassifikation eines zweidimensionalen Musters in einem Verarbeitungsschritt mit den pseudo zweidimensionalen Hidden-Markov-Modellen durchzuführen. Diese Methode basiert auf der Fähigkeit des Viterbi-Algorithmus, die Merkmale den Modellzuständen automatisch zuzuordnen. Die Abb. 5.1 visualisiert, für ein in Merkmalblöcke der Größe 8×8 zerlegtes Bild, solch eine Merkmals-Zustandszuordnung unter Verwendung eines P2DHMM, das aus 6×7 Zuständen besteht. Diejenigen Bildbereiche, deren Merkmale einem einzigen Zustand zugeordnet wurden, sind in der Abbildung von einer weißen Linie umgeben. In der Abb. 5.1 ist gut zu erkennen, daß die Ausgabeverteilungen der Modellzustände jeweils auf bestimmte Bereiche des Bildes adaptiert sind. So wurde der Mund der abgebildeten Person beispielsweise einem einzigen Zustand zugeordnet. Ebenso gibt es Zustände, die ausschließlich dem Hintergrund oder den Haaren zugeordnet wurden. Sind solche Zustände entsprechend identifiziert, so könnte das Bild durch den Viterbi-Algorithmus in verschiedene Bereiche, wie z.B. die Haarregion, den Mundbereich oder den Hintergrund segmentiert werden. Dies ist eine zusätzliche Information, die sich

während des Klassifizierungsschrittes ergibt. Gelingt es, die Merkmale der Umgebung eines zu erkennenden Musters zu modellieren, so könnte durch den Viterbi-Algorithmus, in einem Schritt, ein Muster in seiner Umgebung aufgefunden werden und das Muster klassifiziert werden.

Bevor die gemeinsame Klassifizierung und Segmentierung mit P2DHMM und Umgebungsmodell ausführlich in Unterkapitel 5.3 vorgestellt wird, geht das folgende Unterkapitel auf die rotationsinvariante Modellierung von Objektformen mit P2DHMMs ein.

5.2 Rotationsinvariante Modellierung von Objektformen mit P2DHMMs

Die rotationsinvariante Modellierung mit P2DHMMs wurde im Rahmen dieser Arbeit entwickelt. Sie ermöglicht eine Modellierung von gedrehten Objekten, die eine hohe Toleranz gegenüber Deformationen, die z.B. von Überdeckungen mehrerer Objekte hervorgerufen wurden, aufweist. Insbesondere ist es mit diesem Ansatz möglich, polar abgetastete Muster sowohl in radialer, als auch in zirkularer Richtung verzerrungstolerant zu modellieren. Dies stellt bei stark deformierten Mustern, die etwa von anderen Objekten überdeckt waren, einen Vorteil dar (siehe auch Abb. 3.14). In Kapitel 3 wurde der hier vorgestellte Ansatz kurz erwähnt und in Tabelle 3.3 im Unterkapitel 3.5.5 bereits quantitative Ergebnisse präsentiert, die mit einer Datenbank, die aus deformierten Objekten besteht, erzielt wurden. Da sich Kapitel 3 mit der translations-, skalierungs- und rotationsinvarianten Modellierung von Objektformen bzw. handskizzierten Piktogrammen mit *eindimensionalen* HMMs befaßt und zudem in die Theorie der P2DHMMs erst in Kapitel 4 eingeführt wurde, erfolgt die detaillierte Darstellung des Ansatzes an dieser Stelle.

Abb. 5.2 zeigt die P2DHMM-Topologie, die für die rotationsinvariante Modellierung verwendet werden kann. Auf ähnliche Weise wie in Abb. 3.5 wird das Originalmodell zweimal dupliziert und von den Modellkopien umgeben. Zusätzlich werden die Wahrscheinlichkeiten für die Anfangszustände des ersten Modells sowie die Wahrscheinlichkeiten für die Endzustände des dritten P2DHMM entsprechend Abb. 5.2 verändert. Die Wahrscheinlichkeiten für die Anfangszustände des ersten Modells werden alle auf den Wert $1/N$ gesetzt, wobei N die Anzahl der Metazustände des Originalmodells bezeichnet. Dieser Schritt ist durch die Annahme motiviert, daß alle Drehungswinkel gleichwahrscheinlich sind. Bei dem Versuch, dies auf die Endzustände zu übertragen, treten hingegen Schwierigkeiten auf. Auf diesen Punkt wurde ausführlich in Unterkapitel 3.3.2 eingegangen.

Trotz der offensichtlichen Ähnlichkeiten der Modellstruktur in Abb. 5.2 und der eindimensionalen Struktur in Abb. 3.5 ist dennoch zu beachten, daß die hierarchische Struktur des P2DHMM-Ansatzes zu einer veränderten Modellierung der Merkmale führt. So werden entsprechend Abb. 3.2 und Gleichung 3.2 polar abgetastete Muster auf folgende Weise

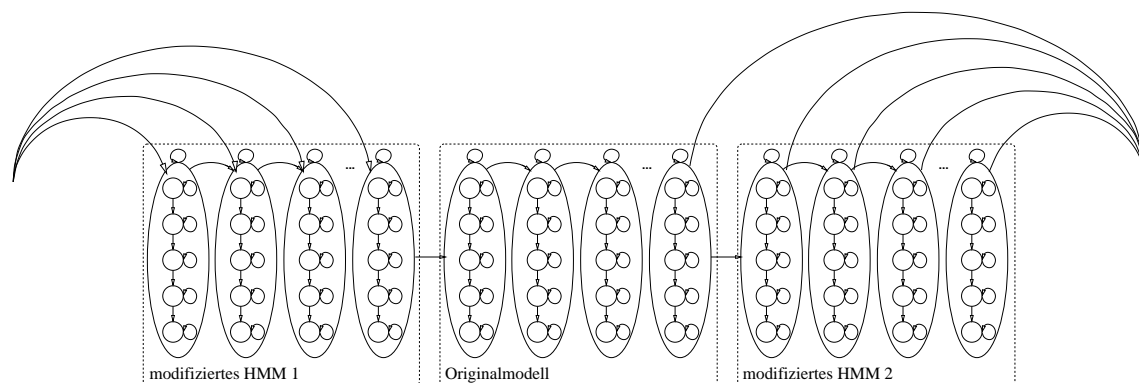


Abbildung 5.2: P2DHMM, das für die Modellierung von deformierten Objektformen verwendet werden kann

modelliert: Die für einen konstanten Winkel bestimmten Abtastwerte ($\varphi = \text{const.}$) werden einem Metazustand zugeordnet, was eine nichtlineare Musterverzerrung in zirkularer Richtung ermöglicht. Zusätzlich werden diese Abtastwerte selbst von eindimensionalen Hidden-Markov-Modellen modelliert. Dies ermöglicht eine nichtlineare Musterverzerrung in radialer Richtung.

Die mit diesem Ansatz erzielten Ergebnisse belegen die verformungstoleranten Modellierungseigenschaften der P2DHMMs. Die Experimente wurden mit einer Datenbasis, die aus künstlich deformierten Objekten besteht, durchgeführt (siehe auch Unterkapitel 3.5.5). Die künstlichen Deformationen sollen die Effekte von Teilüberdeckungen mit anderen Objekten widerspiegeln. Wie Tabelle 3.3 entnommen werden kann, ist die Retrieval-Effizienz gegenüber der Modellierung mit eindimensionalen HMMs von $\eta_T = 75,00\%$ auf $\eta_T = 79,17\%$ verbessert worden. Die im Vergleich mit den eindimensionalen Modellen gesteigerte Toleranz bei Musterverzerrungen in radialer Richtung hat sich hier positiv ausgewirkt. Es sei jedoch auch darauf hingewiesen, daß die Methode mit den eindimensionalen HMMs in Kombination mit den Merkmalstrom-Gewichten nochmals bessere Ergebnisse lieferte.

Nach dieser Beschreibung der rotationsinvarianten Modellierung mit P2DHMMs wird nun im folgenden Kapitel der schon zu Beginn des Kapitels erwähnte Ansatz zur integrierten Klassifizierung und Segmentierung mit pseudo zweidimensionalen HMMs vorgestellt.

5.3 Klassifizierung und Segmentierung mit P2DHMMs und Umgebungsmodell

Ein grundlegendes Problem der klassischen Bildverarbeitung, das bei Mustererkennungsaufgaben in der realen Welt auftritt, ist die große Abhängigkeit von guten Segmentierungsergebnissen. Die Bildsegmentierung wird für die darauf folgende Objektklassifizierung zu einem entscheidenden Schritt: Zunächst muß das Objekt vom Hintergrund isoliert werden,

um dann anschließend klassifiziert zu werden. Diese Trennung vom Hintergrund ist jedoch vielfach nur in Spezialfällen möglich, zum Beispiel falls das zu klassifizierende Objekt im Vergleich zum Hintergrund unterschiedliche Farb- oder Grauwerte aufweist. Bei vielen Mustererkennungsaufgaben sind solche idealen Bedingungen jedoch nicht vorhanden. So ist es kaum möglich, bei Aufnahmen von Straßenszenen, die Passanten, Fahrzeuge, Verkehrsschilder und Häuser zeigen, aufgrund von Farbinformationen Personen zu finden bzw. zu segmentieren. Eine solche Trennung in Objekt und Hintergrund ist ebenfalls nicht möglich, falls es keine Farbinformationen gibt, wie im Fall von Zeichnungen.

Die Abb. 1.1 in der Einleitung z.B. zeigte eine Schwarz-Weiss Skizze, bei der ein Piktogramm in eine komplexe Umgebung¹ eingebettet ist. Bemerkenswert ist, daß menschliche Betrachter ohne Mühe solche komplizierten Segmentierungsaufgaben bewältigen können, da sie in einem einzigen Schritt das Objekt vom Hintergrund trennen und gleichzeitig klassifizieren können. Dies führt zu der Annahme, daß es in vielen Anwendungen nur möglich ist, ein Objekt zu segmentieren, falls es im gleichen Verarbeitungsschritt erkannt und segmentiert wird. Genau diese Anforderung wird von den Hidden-Markov-Modellen erfüllt. Falls beispielsweise der Hintergrund in einem Bild von einem HMM modelliert wird und ein Objekt in dem Bild von einem weiteren Markov-Modell modelliert wird und sind ferner beide auf geeignete Weise miteinander verbunden, so kann mit dem Viterbi-Algorithmus eine Zuordnung der Merkmale von Hintergrund und Objekt zu den entsprechenden Modellen bestimmt werden. Das Ergebnis dieser Zuordnung ist also eine automatische Segmentierung des Bildes in einen Objektanteil und in einen Bildhintergrund. Zusätzlich zu dieser Segmentierung liefert der Viterbi-Algorithmus ein implizites Erkennungsergebnis, da der Viterbi-Algorithmus einen Näherungswert für die Wahrscheinlichkeit ausgibt, daß das Objekt von dem Hidden-Markov-Modell produziert wird.

5.3.1 Objekt-HMM mit Umgebungszuständen

Bei Betrachtung von Abb. 1.1 wird es offensichtlich, daß zweidimensionale Modellierungstechniken verwendet werden müssen, um die Skizze bearbeiten zu können. Daher bietet sich die Modellierung mit pseudo zweidimensionalen Hidden-Markov-Modellen an. Bei der integrierten Segmentierung und Klassifizierung mit P2DHMMs wird auf folgende Weise vorgefahren:

Schritt 1: Training von Objekt-HMMs für jede einzelne Klasse. Dabei können die in Unterkapitel 5.1 beschriebenen Schritte verwendet werden.

Schritt 2: Bestimmung der Wahrscheinlichkeitsverteilung der Merkmale des Hintergrunds.

¹Die Begriffe *Hintergrund* und *Umgebung* werden im folgenden auf gleiche Weise verwendet. Der Grund hierfür ist die gemeinsame Betrachtung von künstlich erzeugten und natürlichen Bildern.

Schritt 3: Umgeben der bereits trainierten Objekt-Modelle mit Zuständen, die die Wahrscheinlichkeitsverteilung der Hintergrundmerkmale als Ausgabewahrscheinlichkeiten verwenden. Diese Zustände werden im folgenden *Umgebungszustände* genannt.

Schritt 4: Anwendung des Viterbi-Algorithmus und anschließende Analyse der Merkmal-Modellzustandszuordnung, sowie Maximum-Likelihood-Entscheidung

Zu dieser Verfahrensweise ist folgendes anzumerken: Das Training von Objekt-Modellen geschieht auf analoge Weise zu dem Training, wie es in Kapitel 5.1 beschrieben wurde. Für jede a-priori festgelegte Klasse wird ein entsprechendes P2DHMM trainiert.

Die Bestimmung der Wahrscheinlichkeitsverteilung der Umgebungsmerkmale stellt hingegen ein noch zu lösendes Problem dar. Dabei sind grundsätzlich zwei verschiedene Situationen zu unterscheiden: Ein mögliches Szenario besteht darin, daß a-priori Wissen über den Hintergrund vorliegt. In diesem Fall können die Wahrscheinlichkeitsverteilungen entsprechend adaptiert werden. Liegt das Wissen über den Hintergrund beispielsweise als Beispiel- oder Trainingsmuster vor, so kann mit diesem Beispielmuster direkt ein Training erfolgen.

Der sehr viel schwierigere Fall besteht darin, daß kein a-priori Wissen über den Hintergrund vorliegt. In diesem Fall kann keine allgemeingültige Methode angegeben werden, um die Verteilung der Merkmale des Hintergrunds zu schätzen. Beispiele, wie für diese schwierige Aufgabe dennoch praxisrelevante Lösungen gefunden werden können, sind in den folgenden Unterkapiteln beschrieben. Das Bestimmen dieser Wahrscheinlichkeitsverteilungen kann auch als das Training eines Markov-Modells, welches nur aus einem einzigen Zustand besteht, angesehen werden. Die in der Literatur häufig zu findende Bestimmung von sog. Gaußschen-Mischverteilungen (GMM) ist identisch zu dem Training eines Hidden-Markov-Modells mit einem Zustand. Der dritte Schritt, der in obiger Aufzählung angegeben wurde, ist das Zusammenfügen der vorher bestimmten Modelle. Dabei ist das Objekt-P2DHMM vollständig mit Umgebungszuständen zu umgeben. Der letzte Schritt besteht in der Anwendung des Viterbi-Algorithmus und der schon beschriebenen Analyse der Merkmal-Modellzustandszuordnung und der mit der Wahrscheinlichkeitsschätzung verbundenen Klassifikation. Anwendungen und Evaluierungen der hier vorgestellten Methode werden in den folgenden Kapiteln vorgestellt.

5.3.2 Erkennen von handskizzierten Piktogrammen in komplexen Szenen

Die Aufgabe, handskizzierte Piktogramme zu erkennen stellt eine große Herausforderung dar (siehe auch Kapitel 3.3). Der Grund hierfür liegt in den großen Variationen innerhalb der Klassen, selbst bei einem einzigen Schreiber. Sind diese Piktogramme zusätzlich in verschiedene Umgebungen eingebettet, so wird die Aufgabe noch erheblich erschwert. In diesem Fall muß zusätzlich zu der Diskriminierung der einzelnen Piktogrammklassen auch das Pik-

togramm vom Hintergrund unterschieden, bzw. lokalisiert werden. In diesem Kapitel besteht der Hintergrund aus denselben Konstruktionselementen wie die Piktogramme selbst, nämlich aus handschriftlich angefertigten Linien und Schraffuren. Die Piktogramme, die in den künstlich erstellten Szenen eingebettet sind, sind dieselben, die in Kapitel 3.3 bzw. Abb. 3.1 vorgestellt wurden. Abbildung 5.3 zeigt sechs Bilder aus der verwendeten Datenbasis. Es



Abbildung 5.3: Handschriftlich skizzierte Piktogramme derselben Klasse 16 eingebettet in drei verschiedene Umgebungen.

wurden Piktogramme derselben Klasse, nämlich Klasse 16 aus Abb. 3.1 in drei verschiedene Umgebungen eingebettet. Die Aufgabe besteht nun darin, die Piktogrammklasse 16 von neunzehn konkurrierenden Klassen zu unterscheiden und das bei einer großen Variation in Größe, Form und Position. Zusätzlich sind die Linien, die den Hintergrund bilden von dem Piktogramm zu unterscheiden. Es kann dabei bei einigen Elementen der erstellten Datenbasis durchaus vorkommen, daß die Hintergrundschraffur in das zu bestimmende Piktogramm hineinragt, oder es berührt (siehe z.B. das untere linke Bild in Abb. 5.3). Insbesondere in diesen Fällen ist es mit konventionellen Methoden nahezu unmöglich, diese Szenen vorzusegmentieren.

Diese Aufgabe kann mittels pseudo zweidimensionaler Hidden-Markov-Modelle gelöst werden. Zu diesem Zweck wird das in Abb. 5.4 dargestellte Modell entsprechend den Ausführungen in Kapitel 5.3.1 verwendet. Die weiß dargestellten Zustände repräsentieren die mit Piktogrammerkmalen trainierten Zustände, wohingegen die grau dargestellten Zustände für die mit Hintergrundmerkmalen trainierten Zustände stehen, also Umgebungszustände sind. Die gepunkteten Modellübergänge in Abb. 5.4 wurden im Gegensatz zu den durchgezogenen Übergängen nicht auf Daten trainiert, sondern willkürlich ausgewählt. Der Grund hierfür ist, daß keine Trainingsdaten zur Verfügung stehen. Durch das Festlegen der in Abb. 5.4 gepunktet dargestellten Übergangswahrscheinlichkeiten wird das a-priori erwartete Flächenverhältnis zwischen Piktogramm und Bildhintergrund einer zu analysierenden Szene modelliert. Da dieses Verhältnis für die zu testenden Bilder nicht als bekannt vorausgesetzt wird, ste-

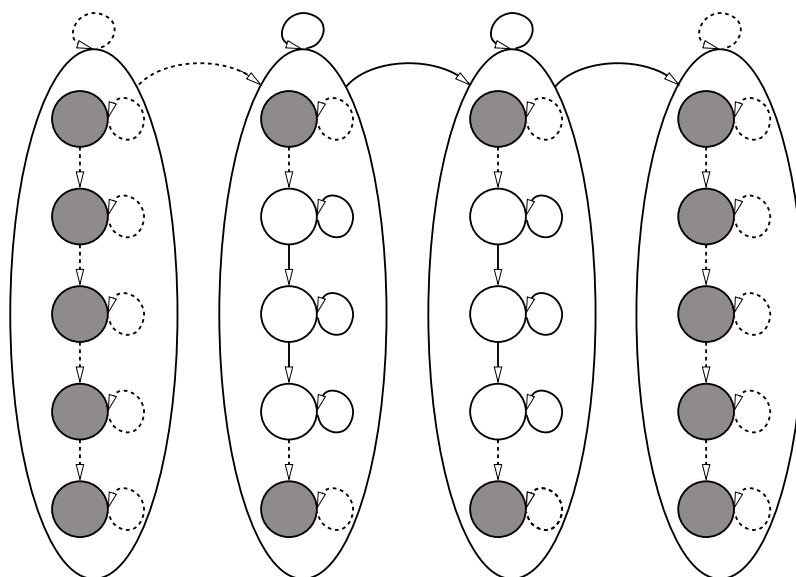


Abbildung 5.4: Pseudo zweidimensionales Hidden-Markov-Modell mit Umgebungszuständen

hen somit keine Trainingsdaten zur Verfügung. Die grau dargestellten Modellzustände teilen alle dieselbe Ausgabewahrscheinlichkeitsdichte. Dies wird im allgemeinen als Parameter-Verknüpfung (engl. *tying*) bezeichnet (siehe [You92]).

5.3.3 Bestimmung der Parameter der Umgebungszustände unter Verwendung von Vorwissen

Um die Ausgabewahrscheinlichkeiten der Umgebungszustände zu schätzen, wurden im Rahmen dieser Arbeit zwei verschiedene Strategien entwickelt. Die erste bezieht Vorwissen über die zu analysierende Szene in die Bestimmung der Ausgabeverteilung ein, während die zweite Strategie ohne dieses Vorwissen auskommt. Die erste entwickelte Methode verwendet eine große Anzahl von Bildern, die Piktogramme in derselben Umgebung, die auch für das Testbild erwartet wird, enthalten. Idealerweise würden Bilder, die ausschließlich Hintergründe enthalten, verwendet. Da solche Bilder jedoch erst hätten erstellt werden müssen, wurden stattdessen die bereits existierenden Bilder der Datenbasis verwendet. Da in diesen Bildern die eingebetteten Piktogramme aus verschiedenen Klassen stammen und sich ferner stets an anderen Positionen im Bild befinden, werden in die Umgebungszustände hauptsächlich die Eigenschaften des Hintergrundes eintrainiert. Der erste Schritt bei beiden Verfahren ist entsprechend der Darstellung in Kapitel 5.3.1 die Erstellung von klassenrepräsentierenden pseudo zweidimensionalen Hidden-Markov-Modellen. Dieser Schritt ist in Abb. 5.5 illustriert. Die Abbildung zeigt, wie mit Beispielsbildern der *Klasse 9* das P2DHMM λ_9 trainiert wird. Die dabei verwendete Merkmalsextraktion ist eine einfache Unterabtastung der Beispielsbilder. Es wurde beispielhaft in Abb. 5.5 von einem Modell der Größe 2×3 ausgegangen.

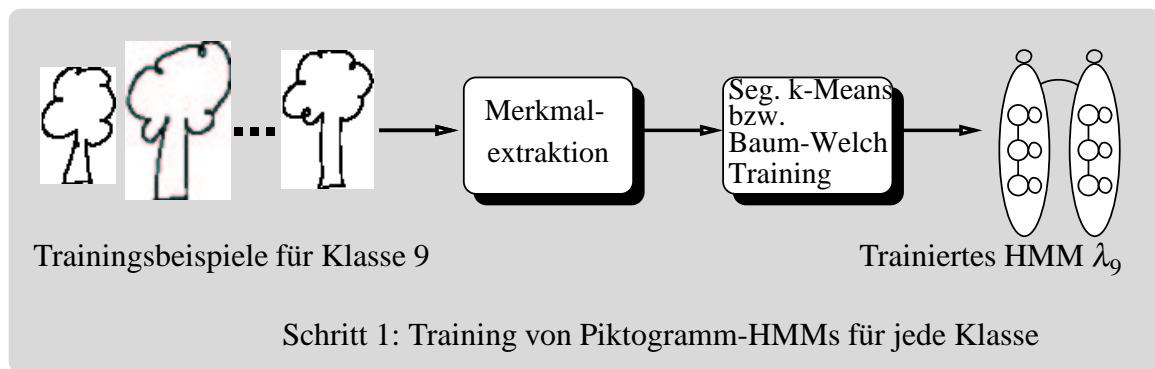


Abbildung 5.5: Training des P2DHMMs für Klasse 9

Die Abb. 5.6 zeigt schematisch die weiteren Bearbeitungsschritte, nämlich die Schätzung der Parameter der Umgebungszustände und den integrierten Segmentierungs- und Klassifizierungsschritt. Nachdem ein zu testendes Bild dem System präsentiert wurde, wird für jedes erweiterte Markov-Modell der Viterbi-Algorithmus durchgeführt. Die vergrößerten Modelle haben entsprechend Abb. 5.6, unter Bezugnahme auf die in Abb. 5.5 gezeigte Größe, eine Größe von 4×5 Zuständen. Die vom Viterbi-Algorithmus ausgegebenen Schätzwerte für die Produktionswahrscheinlichkeiten werden verwendet, um mittels der Maximum-Likelihood (ML)-Entscheidung das Eingangsmuster zu klassifizieren. Dabei wurden die Schätzwerte der Produktionswahrscheinlichkeiten sowohl auf den Hintergrundmerkmalen, als auch auf den Piktogrammerkmalen berechnet, die durch den Viterbi-Algorithmus den Umgebungs- bzw. Piktogrammzuständen zugeordnet werden. Eine mögliche Nachbearbeitung, wie etwa das Herausrechnen von Wahrscheinlichkeiten, die auf den Hintergrundmerkmalen berechnet wurden, ist nicht notwendig, da der Hauptanteil der Produktionswahrscheinlichkeit auf den Piktogrammerkmalen berechnet wird. Ein weiterer Grund ist, daß die Parameter der in Abb. 5.6 dargestellten Umgebungszustände auch über die verschiedenen Modelle hinweg miteinander verknüpft sind. Sie teilen sich alle dieselbe Ausgabeverteilung. Ist also die Anzahl der Merkmale, die den Umgebungszuständen zugeordnet werden für die verschiedenen Modelle in etwa gleich, so ergeben sich aufgrund der gleichen Ausgabeverteilungen auch gleiche Produktionswahrscheinlichkeiten.

5.3.4 Adaptive Bestimmung der Parameter der Umgebungszustände

Die Erfahrungen, die durch die Experimente gesammelt wurden, die mit der im vorherigen Unterkapitel beschriebenen Methode durchgeführt wurden, konnten genutzt werden, um eine verbesserte Methode zu entwerfen. Dieses neue Verfahren verwendet eine Parameterschätzung für den Umgebungszustand, welches kein Vorwissen über das zu klassifizierende Bild erfordert. Dieses adaptive Verfahren ist in Abb. 5.7 schematisch dargestellt. Die Bestimmung der Parameter des Umgebungszustands wird hierbei direkt auf dem unsegmentierten Testbild

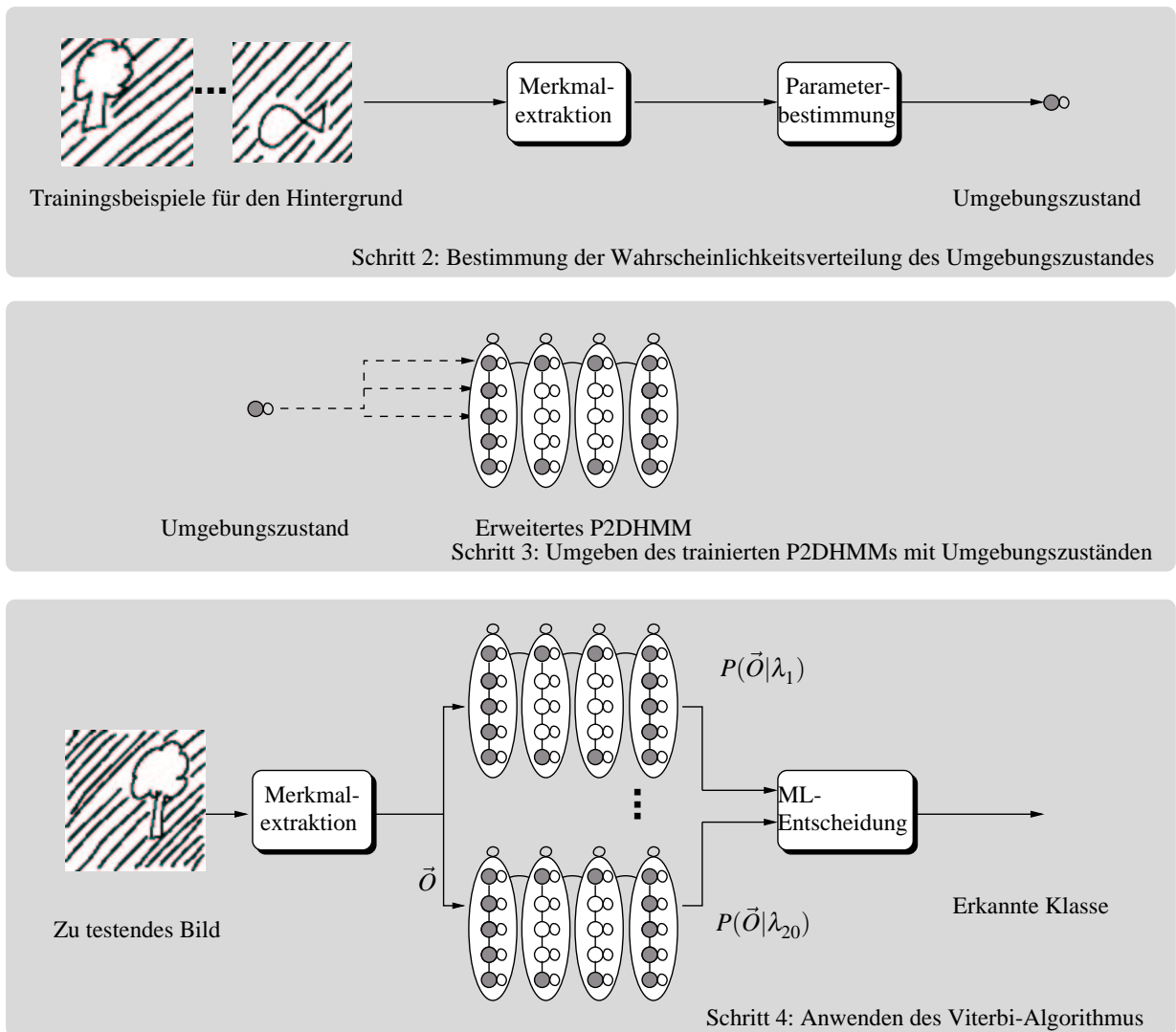


Abbildung 5.6: Schematische Darstellung der auf a-priori Wissen basierenden Parameter-schätzung für den Umgebungszustand sowie die integrierte Segmentierung und Klassifizierung.

durchgeführt. Anders formuliert wird die vollständige Sequenz der Merkmalvektoren, die dem Eingangsbild entnommen wird, zum Training des Umgebungszustandes verwendet. Da bei der verwendeten Datenbasis der Hintergrund die größte Fläche ausfüllt und mithin auf diesem Hintergrund die meisten Merkmalvektoren berechnet werden, wird der Umgebungszustand überwiegend auf die Eigenschaften der Hintergrundmerkmale adaptiert.

5.3.5 Experimentelle Ergebnisse

Die beiden vorgestellten Methoden wurden auf einer Piktogramm-Datenbasis evaluiert, die in der Arbeit [Mul99a] vorgestellt wurde und folgendermaßen aufgebaut ist: Es sind 200 isolierte Piktogramme vorhanden, jeweils 10 verschiedene für jede der Klassen in Abb. 3.1. Weiterhin sind 300 Bilder mit Piktogrammen in drei unterschiedlichen Umgebungen (vgl. Abb. 5.3) vorhanden. Es handelt sich dabei um fünf Beispiele pro Klasse und Hintergrund. Die drei verschiedenen Hintergründe sind sehr gut durch die Beispiele in Abb. 5.3 repräsentiert. Wie der Abbildung ebenfalls entnommen werden kann, variiert neben der Gesamtgröße der Bilder auch die Position der Piktogramme innerhalb des Bildes und zudem auch das Flächenverhältnis zwischen dem Piktogramm selbst und dem Hintergrund. Dieses Flächenverhältnis variiert in der verwendeten Datenbasis zwischen 11 und 41%. Die Bilder wurden auf einem Graphiktableau vom Autor gezeichnet. Es wurde für die Experimente die folgende Merkmalextraktion, die eine Unterabtastung des Bildes darstellt, verwendet: Jedes Bild wurde in Blöcke der Größe 30×30 Bildpunkte, unter Verwendung einer Blocküberlappung von 75%, eingeteilt. Danach wurde jeder Bildblock in neun weitere Teilblöcke geteilt und auf diesen dann der mittlere Grauwert bestimmt. Nach der Merkmalextraktion wurde analog zu Abb. 5.5 jeweils ein P2DHMM für jede Piktogramm-Klasse auf zehn isolierten Beispielbildern trainiert. Diese pseudo zweidimensionalen Modelle hatten eine Größe von fünf Metazuständen mit jeweils fünf Modellzuständen. Nachdem die Ausgabeverteilungen der Umgebungszustände mit Hilfe einer der beiden vorgestellten Verfahren bestimmt waren, wurden die trainierten P2DHMM mit den Umgebungszuständen umgeben und somit auf die Größe 7×7 erweitert. Der Viterbi-Algorithmus wird dann verwendet, um die wahrscheinlichste Zustandssequenz zu bestimmen, die das aktuelle Bild generieren kann. Das Ergebnis des Viterbi-Algorithmus ist die Zuordnung der Merkmale zu den Piktogramm- und Umgebungszuständen. Eine solche Zuordnung ist in Abb. 5.8 für eines der in Abb. 5.3 vorgestellten Beispiele, angegeben. In der Abb. 5.8 steht die graue Schattierung für Merkmalblöcke, die den in Abb. 5.4 ebenfalls grau dargestellten Umgebungszuständen zugeordnet wurden und die weiße Fläche für Merkmale, die dem Piktogrammmodell zugeordnet wurden.

Die Segmentierung isoliert betrachtet stellt kein optimales Ergebnis dar. Die helle Fläche in Abb. 5.8, die dem Piktogramm zugeordnet ist, scheint aus vertikal gegeneinander verschobenen Streifen unterschiedlicher Länge zu bestehen. Somit kann gefolgert werden, daß eine Modell-Topologie verwendet wurde, bei der die Bildspalten von Metazuständen model-

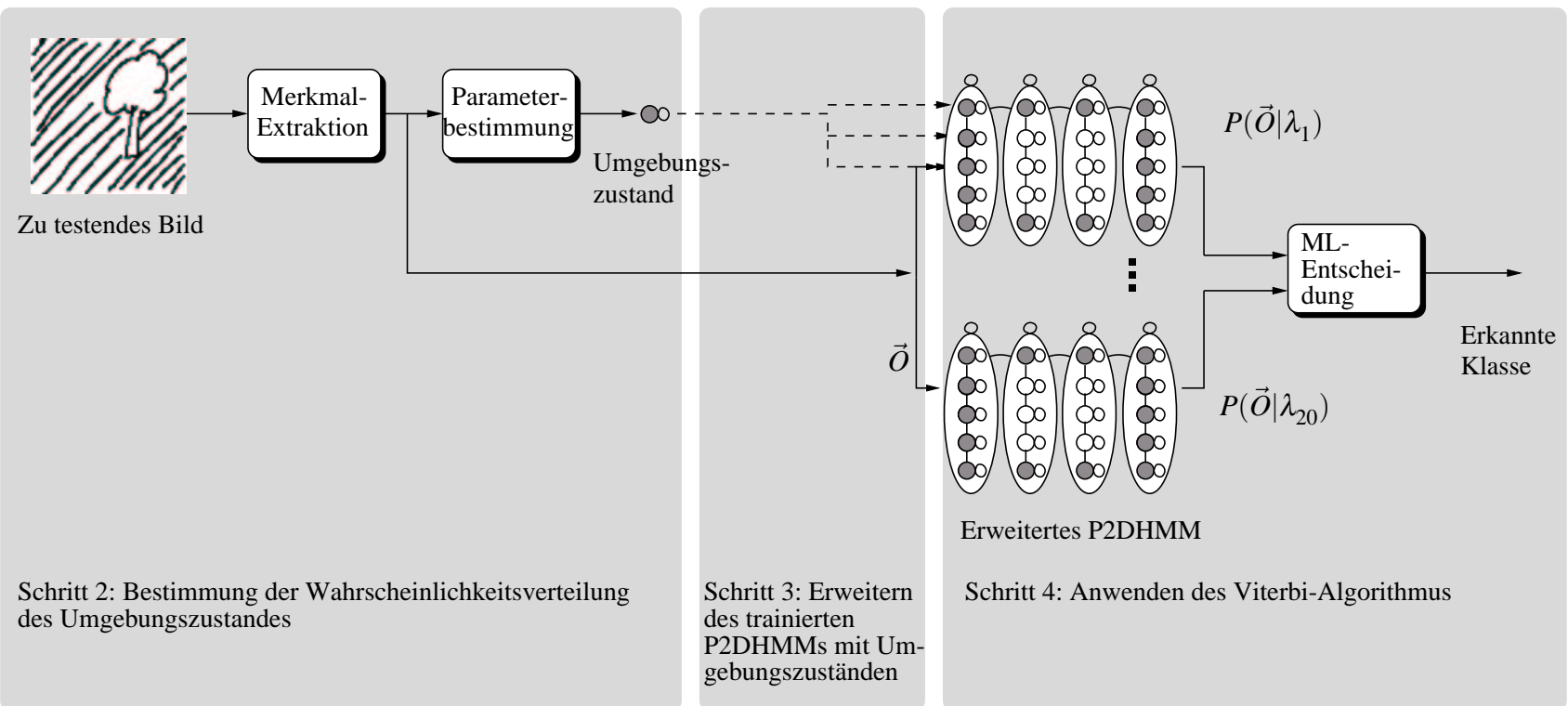


Abbildung 5.7: Adaptive Parameterschätzung

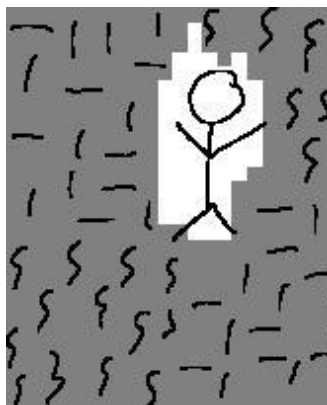


Abbildung 5.8: Segmentierungsergebnis nach Anwendung des Viterbi-Algorithmus



Richtung	Hintergrund 1	Hintergrund 2	Hintergrund 3	Adaptives Verfahren
	91,0%	90,0%	90,0%	90,3%
	91,0%	89,0%	91,0%	90,0%

Tabelle 5.1: Erkennungsgenauigkeiten, die in den Experimenten erzielt wurden

liert werden. Eine Segmentierung bei Verwendung der alternativen Topologie, nämlich der Modellierung von Bildzeilen mit den Metazuständen, würde eine dem Piktogramm zugeordnete Fläche entstehen lassen, die aus gegeneinander verschobenen Zeilen aufgebaut ist. Dieser Aufbau aus vertikalen oder horizontalen Streifen ist eine unmittelbare Folge aus der Vernachlässigung von Abhängigkeiten bei den pseudo zweidimensionalen Modellen.

Wie bereits erwähnt wurde, liefert der Viterbi-Algorithmus neben der wahrscheinlichsten Merkmal-Zustandszuordnung auch einen Schätzwert für die Produktionswahrscheinlichkeit. Aufgrund dieser Wahrscheinlichkeiten, daß das Gesamtbild von dem Hidden-Markov-Modell generiert wurde, erfolgt die Klassifikation. Diese Klassifikationsergebnisse sind in Tabelle 5.1 zusammengefaßt. Die beiden Zeilen der Tabelle geben die Ergebnisse jeweils für unterschiedliche Orientierungen der P2DHMMs an. Die obere Zeile entspricht dabei der Modellierung von Bildspalten mit den Metazuständen und in der unteren Zeile modellieren die Metazustände die Bildzeilen. Dies ist in Tab. 5.1 durch die Verwendung entsprechend orientierter Modelle illustriert. Die ersten drei Spalten in der Tabelle zeigen getrennt nach dem verwendeten Hintergrund die Ergebnisse für die erste vorgestellte Methode (siehe auch Abb. 5.6). Die Erkennungsergebnisse für die adaptive Methode sind in der vierten Spalte angegeben. Bei dieser Methode wurden alle drei Hintergründe gemeinsam präsentiert.

5.3.6 Retrieval von Formen in technischen Zeichnungen

In Kapitel 3 konnten Methoden, die mit Hilfe von Piktogrammdatenbasen evaluiert wurden, erfolgreich auf praxisrelevantere Aufgaben, wie etwa der inhaltsbasierten Abfrage von Bilddatenbasen (Kapitel 3.5), übertragen werden. Auf ähnliche Weise können die Algorithmen der Kapitel 5.3.1 bis 5.3.4 auf natürliche Bilder übertragen werden. Anwendungsbeispiele, bei denen natürliche Bilder verarbeitet werden, sind z.B. das Erkennen von Personen in komplexen (Straßen-)Szenen, sowie das Auffinden von Objektformen in technischen Zeichnungen. Die letztgenannte Aufgabe wird in den folgenden Abschnitten behandelt.

Es wurde in Kapitel 3.5 bereits erwähnt, daß inhaltsbasierte Abfragen von Bilddatenbasen eine erhebliche Reduktion des Indexierungsaufwandes ermöglichen und ferner dem Benutzer eines solchen Systems eine intuitive Benutzerschnittstelle zur Verfügung stellen. In diesem Kapitel wird ein experimentelles Datenbanksystem vorgestellt, das es ermöglicht, technische Zeichnungen inhaltsbasiert abfragen zu können. Durch den Einsatz der pseudo zweidimensionalen Markov-Modelle mit Umgebungsmodell können mittels einer Skizze spezifizierte Details in den technischen Zeichnungen gefunden und lokalisiert werden. Eine solche skizzenbasierte Anfrage könnte etwa der folgenden textuellen Beschreibung entsprechen: *Zeige alle technischen Zeichnungen, die eine oder mehrere Schrauben enthalten*. Bei der textuellen Beschreibung fehlt jedoch noch eine Angabe über die spezielle Form der gesuchten Schraube. Dies ist mit Hilfe einer Skizze sehr viel intuitiver zu spezifizieren. Abb. 5.9 zeigt eine Anfrageskizze und die dazu passenden technischen Zeichnungen aus der verwen-

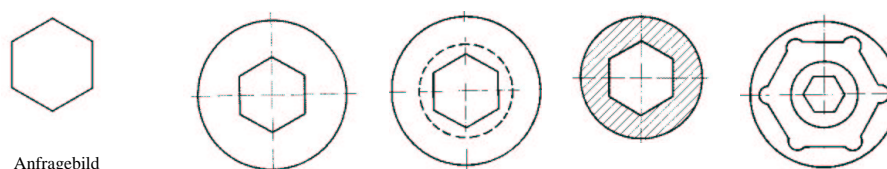


Abbildung 5.9: Anfrageskizze und vier Elemente aus der verwendeten Datenbasis. Die technischen Zeichnungen wurden [Bod88] entnommen.

deten Datenbasis. Ein solches Abfrageergebnis ist sehr schwer zu erzielen, da das gesuchte Objekt (eine hexagonale Form) mit anderen Objekten in der technischen Zeichnung verbunden ist und auch in seiner Größe stark variiert. Da diese Aufgabe sehr ähnlich zu der in Kapitel 5.3.2 vorgestellten Aufgabe ist, nämlich der Klassifikation von Piktogrammen in komplexen Umgebungen, ist es offensichtlich, daß die vorgestellte Modellierung mit zweidimensionalen Markov-Modellen hier einen Lösungsansatz darstellt. Diese Methode unterscheidet sich von konventionellen mehrstufigen Ansätzen, die zunächst Bildelemente wie beispielsweise Linien, Kreisbögen und Schnittpunkte bestimmen und anschließend eine darauf aufbauende Bildinterpretation durchführen (siehe z.B. [Abl97] und [Kas90]). Solch ein mehrstufiger Ansatz ist jedoch immer dann schwer anzuwenden, wenn es nicht möglich ist, die Zeichnung in einfache Elemente zu unterteilen. Genau dieses Problem tritt jedoch bei den

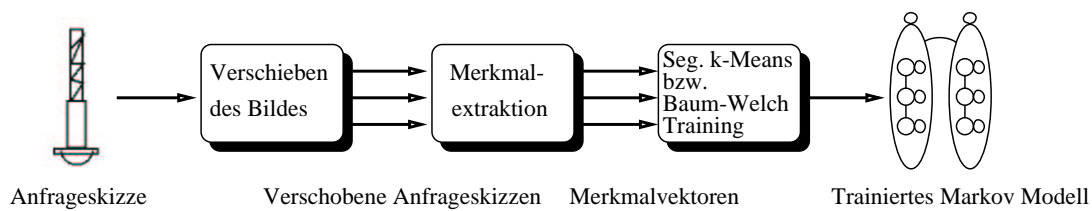


Abbildung 5.10: Training eines pseudo zweidimensionalen Modells auf dem Anfragebild

technischen Zeichnungen auf, da hier Bildelemente oft durch Schraffuren (vgl. Abb. 5.9) verbunden sind und somit beispielsweise eine Zerlegung in nichtverbundene Einzelkomponenten (engl. Connected Component Analysis, siehe auch [Gon92]) scheitert. Aus diesem Grund wird in diesem Kapitel der integrierte Segmentierungs- und Klassifikationsansatz verwendet. Grundsätzlich kommt die gleiche Modellierung, wie sie in Unterkapitel 5.3.1 beschrieben wurde, zum Einsatz.

Da das P2DHMM in Abb. 5.4 das gesuchte Objekt in einem unbekanntem Kontext modelliert, muß mit dem Anfragebild ein entsprechendes Objekt-Modell trainiert werden. Hierfür steht lediglich ein einziges Bild zur Verfügung. Aus diesem Grund wird das Anfragebild um einige Bildpunkte in jede Richtung verschoben, um so künstlich die Anzahl an Trainingsbildern zu erhöhen und eine robustere Modellierung zu erreichen. Dies ist in Abb. 5.10 illustriert. Die Umgebung, in die das gesuchte Objekt eingebettet ist, wird mit Umgebungszuständen modelliert, deren Ausgabeverteilungen auf den einzelnen Datenbankelementen ermittelt wurden. Das mit den Umgebungszuständen erweiterte Modell ist, wie in Abb. 5.4 dargestellt, aufgebaut. Abbildung 5.11 zeigt die Bestimmung der Parameter der Wahrscheinlichkeitsdichten der Umgebungszustände. Jedes Bild der Datenbank wird durch eine Merkmalsequenz und deren modellierte Verteilung repräsentiert. Sobald ein Anfragebild vorliegt, wird ein Objektmodell trainiert (siehe Abb. 5.10) und dieses Modell mit Umgebungszuständen erweitert. Anschließend werden sukzessive für jedes Datenbankelement die Produktionswahrscheinlichkeiten mit dem Viterbi-Algorithmus bestimmt. Dabei werden die jeweiligen Merkmalsequenzen und Wahrscheinlichkeitsverteilungen für die Umgebungszustände (siehe Abb. 5.10) verwendet.

Die wesentlichen Schritte bei dem Verfahren zum Auffinden des Anfrageobjekts in Bild-datenbanken mit pseudo zweidimensionalen Modellen werden im folgenden zusammengefaßt:

- Schritt 1: Merkmalsextraktion und Parameterbestimmung der Umgebungszustände für jedes Bild der Datenbank (Abb. 5.11)
- Schritt 2: Präsentation eines Anfragebildes und Training eines Hidden-Markov-Modells für dieses Bild (Abb. 5.10)
- Schritt 3: Erweiterung des in Schritt 2 bestimmten P2DHMM mit den Umgebungszuständen (vgl. Abb. 5.4)

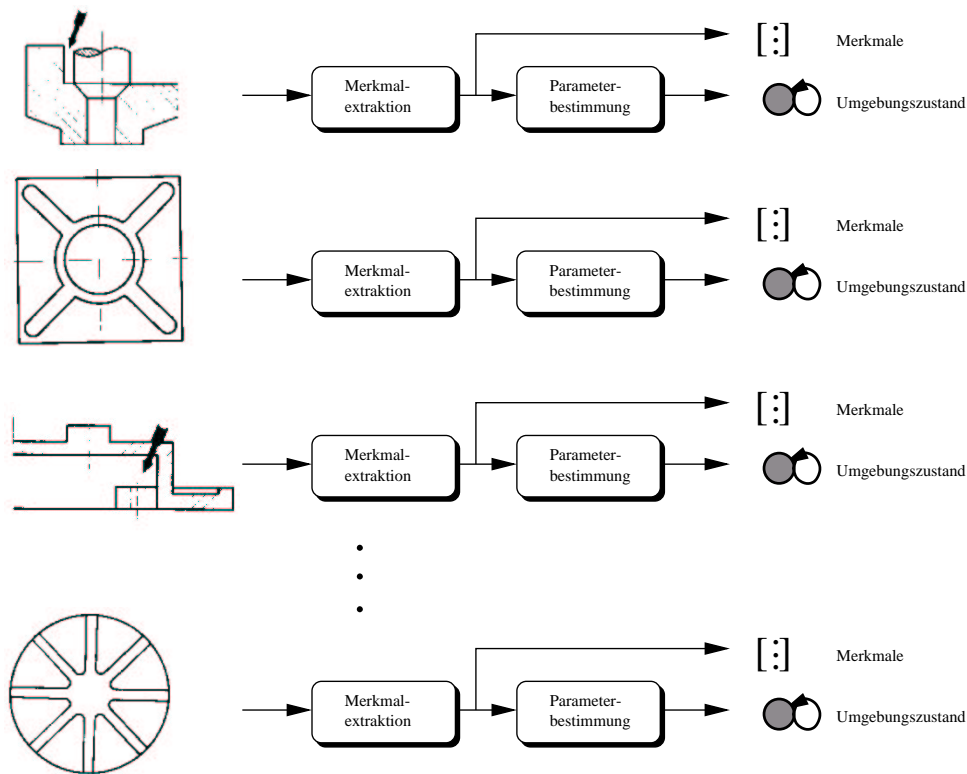


Abbildung 5.11: Bearbeitung der Datenbankelemente

Schritt 4: Für jedes Datenbankelement wird mit dem Viterbi-Algorithmus die Merkmals-Modellzustandszuordnung und die Produktionswahrscheinlichkeit berechnet. Dabei werden sowohl die in Schritt 1 berechneten Merkmale als auch deren ebenfalls schon berechneten Wahrscheinlichkeitsverteilungen verwendet. Die grundlegende Modellstruktur nach Abb. 5.4 bleibt jedoch bestehen.

Schritt 5: Unter Verwendung der in Schritt 4 bestimmten Segmentierungen werden die Produktionswahrscheinlichkeiten neu berechnet. Diese Neuberechnung erfolgt ausschließlich auf den Merkmalen, die den Objektzuständen zugeordnet wurden.

Schritt 6: Ordnen der Datenbankelemente nach den Neuberechneten Produktionswahrscheinlichkeiten und Anzeigen der Bilder auf den höchsten Rängen.

Schritt 7: Zurück zu Schritt 2 und präsentieren der nächsten Anfrage.

Zu Schritt 2 ist anzumerken, daß im Gegensatz zu dem in Kapitel 3.5 dargestellten Retrievalverfahren das Hidden-Markov-Modell mit dem Anfragebild trainiert wird. Bei dem Verfahren in Kapitel 3.5 wird eine Merkmalsequenz auf dem Anfragebild berechnet und die Datenbankelemente durch Markov-Modelle repräsentiert. Eine weitere Anmerkung betrifft Schritt 1 in obiger Aufzählung: Dieser Schritt muß lediglich einmal ausgeführt werden, solange die Datenbank nicht verändert wird. Falls eine technische Zeichnung der Datenbank

zugefügt wird, muß die Merkmalextraktion und die Bestimmung der Wahrscheinlichkeitsverteilung nur für dieses neue Element durchgeführt werden. Die verwendete Datenbank besteht aus 56 technischen Zeichnungen, die mit Hilfe eines Scanners digitalisiert wurden und alle dem Konstruktionsatlas von Bode [Bod88] entstammen. Die Bilder wurden mit 300dpi eingescannt und variieren in der Größe zwischen 180×134 und 256×240 Bildpunkten. Die Bilder werden direkt, d.h. ohne Verwendung von typischen Vorverarbeitungsschritten wie z.B. dem Entfernen isolierter Bildpunkte (Despeckle) oder einer Liniendickennormalisierung (Linethinning) verarbeitet.

Testdurchläufe mit dem experimentellen System wurden mit zehn verschiedenen Anfrageskizzen, die mit dem verbreiteten UNIX-Programm *xfig* erstellt wurden, durchgeführt. Drei dieser Anfragen sind zusammen mit den vom System zurückgelieferten technischen Zeichnungen in den Abb. 5.12 bis 5.14 gezeigt. Die Anfrageskizzen sind in den Abbildungen jeweils im linken oberen Teil dargestellt. Es sind die ähnlichsten fünf Elemente der Datenbank, geordnet nach ihrem Ähnlichkeitsmaß von links nach rechts und oben nach unten, dargestellt. Bemerkenswert sind die guten Abfrageergebnisse z.B. in Abb. 5.12, wo mit der groben Skizze einer Schraube zwei technische Zeichnungen gefunden werden konnten, die eine solche Schraube enthalten und dies obwohl beide Objekte in Schraffuren eingebettet sind. Weitere qualitative Ergebnisse, die mit dem in diesem Kapitel dargestellten System erzielt wurden, sind in den Arbeiten [Mul99e] und [Mul00b] zu finden. Mit einem algorithmisch identischen Verfahren, jedoch in dem Anwendungskontext der Kunstarchive wurde ein System zum Auffinden von Wasserzeichen in [Mul99g] vorgestellt.

In den Abb. 5.12 bis 5.14 sind die Rangfolgen, geordnet nach den berechneten Produktionswahrscheinlichkeiten angegeben. Es wurden während der Anwendung des Viterbi-Algorithmus jedoch auch die gesuchten Objekte in den Zeichnungen lokalisiert. Eine solche Lokalisierung, die durch die Zuordnung der Merkmale zu den Modellzuständen ermittelt wird, ist exemplarisch in Abb. 5.15, für die skizzierte Schraube und das Datenbankelement mit dem höchsten Ähnlichkeitsmaß (vgl. Abb. 5.12), angegeben. Das Raster, das in Abb. 5.15 über die technische Zeichnung gelegt wurde, visualisiert die einzelnen Blöcke, auf denen die Merkmalvektoren berechnet werden. Die Merkmale, die den grau schattierten Flächen entnommen wurden, sind durch den Viterbi-Algorithmus den Umgebungszuständen zugeordnet worden (vgl. Abb. 5.4), wohingegen die weiß dargestellten Flächen zu Merkmalen gehören, die dem Objektmodell (Anfragemodell) zugeordnet wurden. Obwohl die dargestellte Segmentierung kein optimales Ergebnis darstellt, wird sich dennoch durch die Zuordnung der weiß dargestellten Merkmale zu den Objektzuständen eine hohe Produktionswahrscheinlichkeit ergeben.

Die im Rahmen dieser Arbeit entwickelte Methode, nämlich die integrierte Segmentierung und Klassifikation mit P2DHMMs zeigt gute Ergebnisse auf dem Gebiet des Retrievals von technischen Zeichnungen. Um die Leistungsfähigkeit dieses Ansatzes weiter testen zu können, wurde nach einer anspruchsvollen und gleichzeitig praxisrelevanten Anwendung

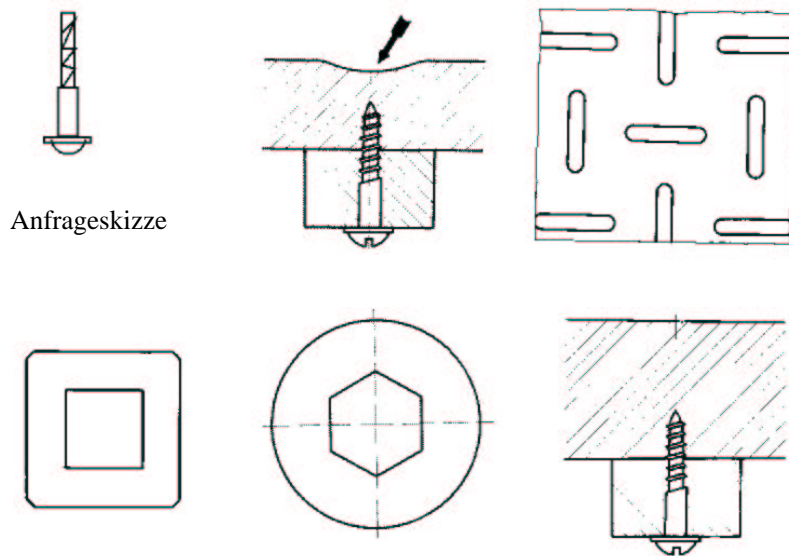


Abbildung 5.12: Anfrageskizze und fünf zurückgelieferte technische Zeichnungen (1)

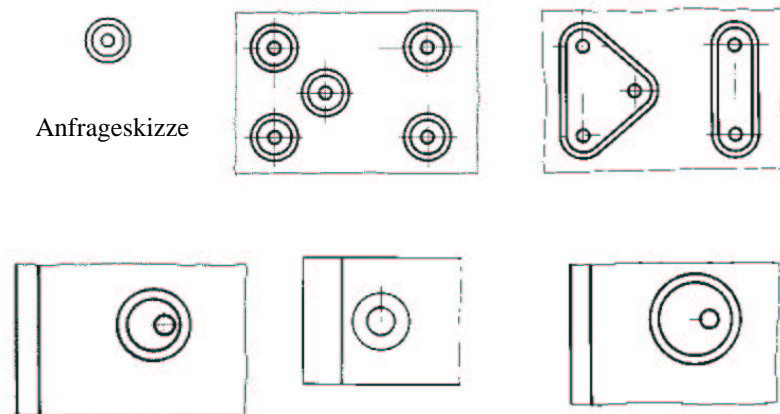


Abbildung 5.13: Anfrageskizze und fünf zurückgelieferte technische Zeichnungen (2)

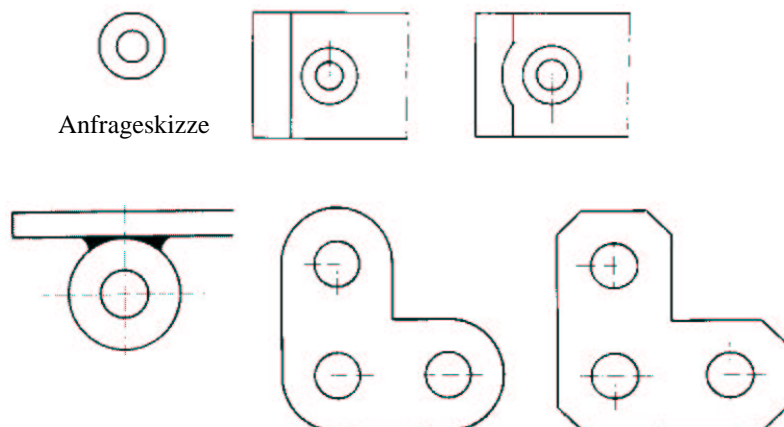


Abbildung 5.14: Anfrageskizze und fünf zurückgelieferte technische Zeichnungen (3)

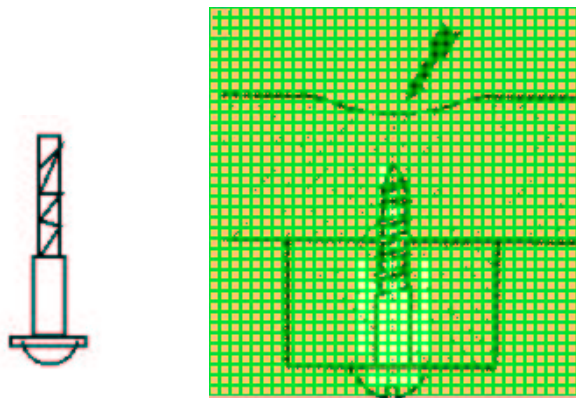


Abbildung 5.15: Anfrageskizze und zugeordnete Segmentierung des Datenbankelements

gesucht, die nicht ausschließlich binäre Bilder, wie technische Zeichnungen verwendet. Eine solche Aufgabe wurde in Form des im folgenden Kapitel beschriebenen Personen-Trackings gefunden.

5.4 Tracking von Personen

Das Tracking von Personen stellt ein sehr interessantes Anwendungsszenario für die P2DHMMs mit Umgebungsmodell dar. So kann untersucht werden, ob die Methode auch auf Grauwertbilder bzw. Farbbilder übertragbar ist. Zudem kann überprüft werden, ob die Erkennung von Personen in Einzelbildern in geeigneter Weise mit einem Algorithmus kombiniert werden kann, mit dem ganze Bildsequenzen bearbeiten werden können. Bevor das Personen-Tracking mit P2DHMM ausführlich beschrieben wird, wird zunächst kurz in dieses Anwendungsszenario eingeführt.

Das Tracking von Personen kann als eine Basistechnik angesehen werden, die eine Vielzahl von Applikationen, insbesondere in der Mensch-Maschine-Kommunikation und im Multimediabereich ermöglicht. Bei der Mensch-Maschine-Kommunikation ist die Information von großer Relevanz, ob sich eine oder mehrere Personen in einer definierten Umgebung befinden. Zusätzlich muß oft herausgefunden werden, wer die Personen sind, welche Handlungen diese ausführen und wo die genauen Positionen der Personen sind. Das Personen-Tracking löst dabei zunächst die folgenden beiden Unteraufgaben: Die Anwesenheit von Personen wird festgestellt und deren Positionen werden über eine Sequenz von Bildern verfolgt. Die Kenntnis der Position der Personen kann genutzt werden, um in einem weiteren Schritt die Personen zu identifizieren. Dies kann durch das Ausschneiden des Gesichts und durch die Verwendung der in Unterkapitel 5.1 vorgestellten Klassifikationsmethoden erfolgen. Weiterhin gibt die Position, bzw. die Trajektorie einer Person einen wichtigen Hinweis auf ausgeführte Aktionen oder Gesten.

Die Analyse und Erkennung von menschlichen Handlungen in Bildsequenzen wird auch bei einer kommerziell sehr bedeutenden Anwendung, nämlich der automatischen Überwachung benötigt. Beispiele für Überwachungsszenarien schließen Tankstellen, Kaufhäuser, Geldautomaten oder Verkehrskreuzungen ein.

Im Bereich der Multimediaanwendungen kann das Personen-Tracking einen wichtigen Beitrag zum automatischen Indexieren von Videoaufnahmen liefern. Insbesondere bei Sportaufnahmen stellt die Position der Spieler eine wichtige Information dar. Lläuft beispielsweise ein Tennisspieler in die Nähe des Netzes, so erfolgt sehr wahrscheinlich ein Angriff und es wird im folgenden mit hoher Wahrscheinlichkeit ein Volley geschlagen. Eine ähnlich wichtige Information stellt die Position der Spieler beim Fußball dar, da auch hier aus der Verteilung der Spieler auf dem Feld die jeweilige Spielsituation ermittelt werden kann. Ein umfangreicher und aktueller Übersichtsartikel zu dem Thema der menschlichen Aktionsanalyse, die das Personen-Tracking einschließt, ist von Gavrilin in der Arbeit [Gav99] zusammengestellt worden. In [Gav99] findet sich eine Vielzahl weiterer Einsatzmöglichkeiten des Personen-Trackings und eine Vorstellung der am häufigsten eingesetzten Verfahren.

Wie schon erwähnt wurde, teilt sich das Personen-Tracking in die beiden Teilaufgaben, nämlich die Anwesenheit von Personen zu detektieren und die Dynamik der Bewegung zu erfassen. Das folgende Unterkapitel befaßt sich mit der ersten genannten Teilaufgabe, nämlich dem Auffinden von Personen in natürlichen Bildern. Die hierbei eingesetzten Methoden basieren auf der in Kapitel 5.3 eingeführten Modellierung mit pseudo zweidimensionalen Hidden-Markov-Modellen und Umgebungszuständen.

5.4.1 Auffinden von Personen in natürlichen Bildern mit pseudo zweidimensionalen Hidden-Markov-Modellen

Das Auffinden von Personen in Bildern stellt aufgrund der großen Formvariation eine anspruchsvolle Aufgabe dar. Dies kann anhand von Abb. 5.16 erläutert werden. In der Abbildung sind zwei Umrisse derselben sich bewegenden Person gezeigt. Die beiden Umrisse haben sehr unterschiedliche Formen und bei einer weiteren Analyse der Bewegung dieser Person werden zusätzliche Variationen in der Form auftreten. Das Auffinden von Personen



Abbildung 5.16: Umrisse einer sich bewegenden Person



Abbildung 5.17: Beispiele für das Auffinden von Personen in komplexen Umgebungen

ist somit erheblich anspruchsvoller, als das Auffinden von starren Objekten, wie beispielsweise von Fahrzeugen oder das Auffinden von Gesichtern. Rowley beschreibt in [Row96] eine auf künstlichen neuronalen Netzen basierende Methode, mit der frontale Ansichten von Gesichtern sehr zuverlässig in Bildern gefunden werden können. Aufgrund der großen Formvariationen bietet sich diese Methode jedoch nicht zum Auffinden von Personen an. Stattdessen kann die schon beschriebene Methode, die auf P2DHMMs in Kombination mit Umgebungszuständen basiert, verwendet werden, da sie die folgenden Vorteile bietet:

- Die elastischen Modellierungseigenschaften der Hidden-Markov-Modelle passen sehr gut zu den stark variierenden Personenabbildungen.
- Die P2DHMMs ermöglichen eine elastische Modellierung sowohl in vertikaler, als auch in horizontaler Richtung.
- Durch den Viterbi-Algorithmus und die Merkmal-Zustandszuordnung kann eine integrierte Segmentierung und Klassifikation erfolgen. Dies ermöglicht es, nicht nur die Personen aufzufinden, sondern diese im gleichen Schritt auch zu erkennen.
- Da einem Modellzustand unterschiedlich viele Merkmale zugeordnet werden können, ist es möglich, verschieden große zweidimensionale Muster mit einem P2DHMM zu verarbeiten. Dies ist ein Vorteil bei unterschiedlich großen Abbildungen von Personen, die sich aus der Aufnahmesituation oder durch verschieden große Personen ergeben können. Ein auf KNNs basierendes System, wie etwa das in [Row96] beschriebene, ist stets auf die Analyse einer festen Mustergröße beschränkt.
- Es stehen effiziente Algorithmen für das Training und die Klassifikation zur Verfügung.

Die Abb. 5.17 zeigt Ergebnisse, die mit dem im Rahmen dieser Arbeit entwickelten Verfah-

ren erzielt wurden. Die auf der linken Seite in Abb. 5.17 dargestellte Abbildung einer Person wurde zusammen mit anderen, ähnlichen Abbildungen zum Training eines P2DHMMs verwendet. Dabei wurde eine blockweise Merkmalsextraktion zusammen mit einer diskreten Cosinus-Transformation verwendet (siehe Gleichung 5.1 und Abb. 4.3). Anschließend wurde das in Kapitel 5.3.2 vorgestellte, adaptive Parameterschätzungsverfahren für die Umgebungszustände angewendet. Für die beiden Testbilder in Abb. 5.17 wurden jeweils unter Verwendung aller Merkmale Verteilungen bestimmt, die als Ausgabeverteilungen der Umgebungszustände verwendet werden (siehe auch Abb. 5.7). Anschließend wird mit dem Viterbi-Algorithmus die Merkmal-Zustandszuordnung bestimmt und somit die Aufteilung der Merkmale in Umgebung und Person ermittelt. In der Abb. 5.17 ist für die beiden Testbilder die so ermittelte Grenze zwischen der Person und der Umgebung eingezeichnet. Der Abbildung kann entnommen werden, daß bei Verwendung der P2DHMMs mit Umgebungszuständen die Person in komplexen Szenen lokalisiert werden kann.

Neben der adaptiven Methode zur Bestimmung der Verteilung der Umgebungszustände kann auch die nicht-adaptive Methode bei der Personendetektion verwendet werden (siehe Abb. 5.6). Diese bietet Vorteile bei fest installierten Kameras und somit z.B. bei der Überwachung von Tankstellen, da bedingt durch den statischen Hintergrund die Verteilungen der Umgebungszustände sehr zuverlässig geschätzt werden können. Zudem steht oft eine große Anzahl von Trainingsbildern zur Verfügung, bei denen sich keine Person im Bild befindet und somit *ausschließlich* der Hintergrund eintrainiert wird. Es sei jedoch an dieser Stelle angemerkt, daß konventionelle Verfahren, die auf der Subtraktion des Bildhintergrundes basieren, bei fest installierten Kameras geeigneter sind.

Die vorgestellte Personendetektion arbeitet auf einzelnen Bildern und berücksichtigt somit ausschließlich statische Informationen. Um die Dynamik der Bewegung einer Person modellieren zu können, ist somit ein weiterer Verarbeitungsschritt, der Bildsequenzen verwendet, erforderlich. Dies erfolgt mit dem im folgenden betrachteten sog. Kalman-Filter.

5.4.2 Kalman-Filter

Der Kalman-Filter-Algorithmus wurde 1960 von R.E. Kalman in der Arbeit [Kal60] veröffentlicht und stellt eine rekursive Lösung der folgenden Aufgabe dar. Gegeben sei ein diskreter, zeitabhängiger Prozeß, der durch die folgende lineare Differenzgleichung beschrieben wird:

$$\vec{x}_{k+1} = A \cdot \vec{x}_k + \vec{w} \quad (5.3)$$

Dabei ist \vec{x}_k ein Zustandsvektor zum Zeitpunkt t_k , A eine Übergangsmatrix und \vec{w} ein Vektor mit Zufallsvariablen, der das Prozeßrauschen charakterisiert. Es wird angenommen, daß \vec{w} durch eine multivariate Gaußverteilung mit dem Mittelwert 0 und der Kovarianzmatrix Q beschrieben werden kann. Ferner ist die folgende sog. Meßgleichung gegeben, die die nicht

meßbaren Systemzustände \vec{x}_k mit dem beobachtbaren Meßvektor \vec{z}_k verknüpft:

$$\vec{z}_k = H_k \cdot \vec{x}_k + \vec{v} \quad (5.4)$$

Die Matrix H gibt die Beziehung zwischen dem Zustandsvektor \vec{x}_k und dem Vektor der meßbaren Größen \vec{z}_k an. \vec{v} ist ein Vektor aus Zufallsvariablen, der das Meßrauschen charakterisiert. Der Vektor \vec{v} wird wiederum als eine, durch eine multivariate Gaußverteilung mit dem Mittelwert 0 charakterisierte Größe, angesehen. Die Kovarianzmatrix der Gaußverteilung sei mit R bezeichnet. Der Kalman-Filter-Algorithmus löst nun bei einem durch die Gleichungen 5.3 und 5.4 beschriebenen System die Aufgabe, die Zustandsvektoren \vec{x}_k , bei Kenntnis der Meßwerte \vec{z}_k , zu schätzen. Im Gegensatz zu der Größe \vec{x}_k , die den tatsächlichen Zustand des Systems zum Zeitpunkt t_k darstellt, bezeichnet $\hat{\vec{x}}_k$ den Schätzwert für diesen Zustand bei bekanntem Meßwert \vec{z}_k . $\hat{\vec{x}}_k$ wird, da der Meßwert bekannt ist, als *a posteriori* Schätzwert bezeichnet, während $\hat{\vec{x}}_k^-$ den *a priori* Schätzwert, also den Schätzwert für \vec{x}_k , der ohne Berücksichtigung der Messung \vec{z}_k ermittelt wird, bezeichnet.

Der Kalman-Filter-Algorithmus stellt eine rekursive Lösung dar, der aus den folgenden drei Verarbeitungsstufen besteht: der Initialisierung, der Vorhersage der Parameter und dem Abgleich mit der Messung (siehe auch [Wel95]). Die beiden letztgenannten Schritte werden aufeinanderfolgend für jeden Zeitschritt wiederholt. Bei der Initialisierung wird der Anfangszustand $\hat{\vec{x}}_0^-$, sowie die Kovarianzmatrix des Fehlers bei der Bestimmung der Zustände festgelegt. Diese Kovarianzmatrix wird mit P_0^- bezeichnet und berechnet sich zu:

$$P_k^- = E\{e_k^- e_k^{-T}\} \quad \text{mit} \quad e_k^- = \vec{x}_k - \hat{\vec{x}}_k^- \quad (5.5)$$

Die Kovarianzmatrix des Fehlers unter Berücksichtigung der Messung z_k ergibt sich zu:

$$P_k = E\{e_k e_k^T\} \quad \text{mit} \quad e_k = \vec{x}_k - \hat{\vec{x}}_k \quad (5.6)$$

Der Schritt der *Vorhersage der Parameter* ermöglicht das zeitliche Vorwärtsprojizieren des aktuellen Zeitpunktes und der aktuellen Kovarianzmatrizen des Fehlers. Das Ergebnis dieses Schritts sind a-priori-Schätzwerte für den folgenden Zeitschritt. Das Kalman-Filter verwendet hierbei die folgenden Gleichungen:

$$\hat{\vec{x}}_{k+1}^- = A \cdot \hat{\vec{x}}_k \quad (5.7)$$

$$P_{k+1}^- = A \cdot P_k \cdot A^T + Q \quad (5.8)$$

Der Schritt der mit *Abgleich mit der Messung* bezeichnet wurde, verwendet eine neue Messung, um zusammen mit der a-priori-Schätzung eine verbesserte a-posteriori-Schätzung zu erhalten. Dabei kommen die folgenden Gleichungen zum Einsatz:

$$K_k = \frac{P_k^- H^T}{H P_k^- - H^T + R} \quad (5.9)$$

$$\hat{\vec{x}}_k = \hat{\vec{x}}_k^- + K_k \cdot [\vec{z}_k - H \cdot \hat{\vec{x}}_k^-] \quad (5.10)$$

$$P_k^+ = [I - K_k \cdot H] \cdot P_k^- \quad (5.11)$$

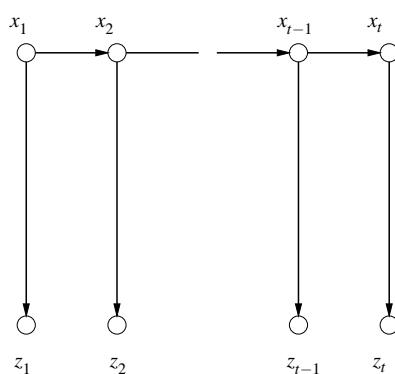


Abbildung 5.18: Darstellung des dem Kalman-Filter zugrundeliegenden Modells als dynamisches Bayes-Netz (aus [Mur00])

In Gleichung 5.11 bezeichnet I die Einheitsmatrix und die Größe K_k die sog. *Kalman-Verstärkung*. Nach der Durchführung der beiden durch die Gleichungen 5.7 und 5.8 bzw. 5.9 bis 5.11 beschriebenen Schritte werden diese für den folgenden Zeitschritt wiederholt. Dabei werden die a-posteriori-Schätzwerte des aktuellen Zustandes für die Vorhersage der a-priori-Schätzwerte des folgenden Zeitschrittes verwendet. Diese rekursive Arbeitsweise des Kalman-Filters hat zu einer weiten Verbreitung des Algorithmus geführt, da sie den Echtzeiteinsatz ermöglicht. Die Gleichungen des Kalman-Filters wurden in diesem Unterkapitel ohne Herleitung angegeben. Diese Herleitungen finden sich z.B. in dem Buch [Gre93].

Das Kalman-Filter stellt nicht nur einen Algorithmus dar, der aus den rauschbehafteten Messungen die tatsächlichen System-Zustände schätzt, bzw. *filtert*, sondern es kann auch als ein Modell für Vorgänge und Prozesse angesehen werden. Dies wird vor allem anhand der Gleichungen 5.3 und 5.4 deutlich. Die Modellvorstellung, die das Kalman-Filter zugrundelegt, kann ähnlich wie bei Hidden-Markov-Modellen basierend auf dynamischen Bayes-Netzen weiter analysiert werden (siehe auch Unterkapitel 2.2.6). Abb. 5.18 stellt das Kalman-Filter als dynamisches Bayes-Netz dar. Bei dem Vergleich zwischen den Abb. 2.4 und 5.18 fällt die große Ähnlichkeit zwischen beiden Bayes-Netzen auf. Dies deutet auf ähnliche Abhängigkeitsbeziehungen hin, die sowohl beim Hidden-Markov-Modell, als auch beim Kalman-Filter vorausgesetzt werden. Diese gemeinsamen Abhängigkeitsbeziehungen können auch den Gleichungen 2.2 und 5.3 entnommen werden, die beide die jeweiligen Modelle als kausal und als beschränkt gedächtnisbehaftet charakterisieren. Ferner liegt bei beiden Modellen eine statistischen Abhängigkeit erster Ordnung der nicht beobachtbaren Variablen vor. Diese nicht beobachtbaren Zufallsvariablen sind beim Kalman-Filter durch die Systemzustände \vec{x}_k und beim Hidden-Markov-Modell durch die Modellzustände q_t gegeben. Der wichtigste Unterschied zwischen beiden Modellen besteht darin, daß alle Knoten des Bayes-Netzes, die den Zufallsvariablen des Modells entsprechen, im Fall des Kalman-Filters kontinuierliche Werte annehmen können. Dies ist beim HMM nicht der Fall, da die Zufallsvariable für die Einnahme eines Modellzustands q_t nur diskrete Wer-

te aus der Menge $\{S_1, \dots, S_N\}$ annehmen kann. Zusammenfassend läßt sich somit feststellen, daß das dem Kalman-Filter zugrundeliegende Modell ein Hidden-Markov-Modell mit nicht beobachtbaren Zufallsvariablen ist, die beliebige Werte annehmen können (siehe auch [Min96, Smy97, Min99, Mur00]).

5.4.3 Interaktion zwischen Kalman-Filter und P2DHMM

Mit dem P2DHMM-basierten Ansatz, der in diesem Kapitel 5 ausführlich beschrieben wurde, ist es möglich, Personen in komplexen Bildszenen aufzufinden (siehe auch Abb. 5.17). Dabei werden ausschließlich Einzelbilder betrachtet. Zusätzlich steht mit dem im vorherigen Unterkapitel vorgestellten Kalman-Filter ein Algorithmus zur Verfügung, der die dynamische Modellierung von Bildsequenzen ermöglicht. Die Kombination beider Methoden erlaubt es somit, Personen in einer Bildsequenz zu verfolgen. Um dies zu erreichen, wird das im folgenden beschriebene dynamische Modell für die Bewegung einer Person angenommen. Der Zustandsvektor \vec{x} sei folgendermaßen gegeben:

$$\vec{x} = (x_s, y_s, v_x, v_y, w, h)^T \quad (5.12)$$

In obiger Gleichung steht das Wertepaar (x_s, y_s) für die Koordinaten des Schwerpunkts einer Person, (v_x, v_y) bezeichnen die horizontale bzw. die vertikale Geschwindigkeit des Schwerpunktes und (w, h) bezeichnen die Breite und Höhe eines die Person umschreibenden Rechtecks. Die Übergangsmatrix A wird folgendermaßen gewählt und stellt zusammen mit Gleichung 5.3 ein einfaches dynamisches Modell dar:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.13)$$

Unter Verwendung des Segmentierungsergebnisses, welches durch die pseudo zweidimensionalen Hidden-Markov-Modelle erzeugt wird, wird der folgende Meßvektor bestimmt:

$$\vec{z} = (x_s, y_s, w, h)^T \quad (5.14)$$

Dabei steht das Wertepaar (x_s, y_s) für die aus dem Segmentierungsergebnis berechneten Koordinaten des Schwerpunktes der Person. Die Größen (w, h) in Gleichung 5.14 sind die Breite, bzw. die Höhe eines die segmentierte Person umschließenden Rechtecks. Sowohl der Schwerpunkt (x_s, y_s) als auch die Größen (w, h) können z.B. auf der eingerahmten Fläche in Abb. 5.17 berechnet werden. Durch diese Festlegung des Meßvektors ergibt sich die

folgende Matrix H :

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.15)$$

Unter Verwendung der P2DHMMs und des Kalman-Filters, sowie des bisher definierten Zustandsvektors, des Meßvektors und der beiden Matrizen A und H , können Personen in Bildsequenzen verfolgt werden. Es ist jedoch zweckmäßig, zusätzlich eine Interaktion zwischen dem Markov-Modell und dem Kalman-Filter einzuführen. Dieser Schritt erhöht die Robustheit des Tracking-Verfahrens und funktioniert auf folgende Weise: Bei der Bestimmung der Segmentierung mit dem P2DHMM wird die zu verfolgende Person nicht im gesamten Bild gesucht, sondern lediglich in dem durch das Kalman-Filter vorhergesagten Bereich. Dieser Bereich ist durch die Komponenten des a-priori Schätzwertes \vec{x}_{k+1}^- gegeben. Zusätzlich wird eine Erweiterung des Suchbereiches um den Faktor 1,5 vorgenommen, da somit sichergestellt wird, daß es genügend Merkmale gibt, die den Umgebungszuständen zugeordnet werden können. Unter Verwendung der aktuellen Meßwerte, die dem Segmentierungsergebnis entnommen werden, wird der a-posteriori-Schätzwert \vec{x}_k für die Systemzustände ermittelt. Die Komponenten x_s und y_s des Vektors \vec{x}_k , die den Schwerpunkt der Person angeben, werden verwendet, um die Position der Person in den einzelnen Bildern einer Sequenz zu bestimmen, bzw. zu markieren. Dies wird an einer Beispielsequenz im folgenden Unterkapitel gezeigt. Vorher werden jedoch die wesentlichen Schritte bei dem Trackingverfahren mit P2DHMM und Kalman-Filter im folgenden zusammenfassend dargestellt.

Das Tracking-Verfahren beginnt mit dem manuellen Ausschneiden einer rechteckigen Region aus dem ersten Bild einer Sequenz. Diese Bildregion soll die zu trackende Person möglichst vollständig enthalten. Das mit Beispielen einer großen Zahl von Personen vortrainierte P2DHMM wird an die zu trackende Person unter Verwendung eines Trainingsschrittes adaptiert. Anschließend wird mit dem Viterbi-Algorithmus die Merkmal-Zustandszuordnung bestimmt und somit die Modellzustände identifiziert, die die Merkmale der Person bzw. den Bildhintergrund modellieren. Schließlich werden sukzessive die folgenden Einzelschritte durchgeführt:

Schritt 1: Mit dem Viterbi-Algorithmus und unter Verwendung des adaptierten P2DHMMs wird ein Bild der Sequenz in eine Personenregion und in einen Bildhintergrund segmentiert (vgl. Abb. 5.17).

Schritt 2: Ausgehend von dieser Segmentierung werden die Komponenten des Meßvektors bestimmt. Die Koordinaten des Schwerpunktes der Person (x_s, y_s) werden als Flächenschwerpunkt der der Person zugeordneten Bildregion bestimmt. Anschließend wird ein die Personenregion umschreibendes Rechteck bestimmt, dessen Breite bzw. Höhe das Wertepaar (w, h) bestimmt.

Schritt 3: Der Meßvektor $\vec{z} = (x_s, y_s, w, h)^T$ wird dem Kalman-Filter zugeführt. Das Kalman-Filter führt den Schritt *Abgleich mit der Messung* aus. Hierzu werden die Gleichungen 5.9 bis 5.11 verwendet.

Schritt 4: Die a-priori-Schätzwerte für den folgenden Zeitschritt \vec{x}_{k+1}^- , die aus dem Schritt *Vorhersage der Parameter* stammen, werden verwendet, um den Suchbereich für das folgende Bild der Sequenz zu bestimmen (Gleichungen 5.7 und 5.8). Dieser Suchbereich wird um den Faktor 1,5 erweitert.

Schritt 5: Die Segmentierung des folgenden Bildes der Sequenz in einen Hintergrund und eine Personenregion wird auf diesen Suchbereich eingeschränkt.

Schritt 7: Zurück zu Schritt 1 und bearbeiten des folgenden Einzelbildes.

5.4.4 Experimentelle Ergebnisse

Abb. 5.19 zeigt einen Ausschnitt aus einer Sequenz, die eine sich bewegende Person zeigt. Zusätzlich sind die Ergebnisse des Personen-Trackings durch die blauen Rechtecke visualisiert. Diese Rechtecke repräsentieren die Komponenten x_s, y_s und w, h des a-posteriori Schätzwertes für den Zustandsvektor \vec{x}_k . Die Komponenten des dabei verwendeten Meßvektors \vec{z} werden aus dem Segmentierungsergebnis berechnet, das mit einem P2DHMM, das aus 6×6 Zuständen besteht, erzeugt wird. Das Training des Modells erfolgt mit handsegmentierten Abbildungen der gleichen Person, die aus einer anderen Sequenz stammen. Das gute Trackingergebnis wurde trotz der teilweisen Verdeckung der Person durch Stuhlreihen und Tische erzielt. Die starke Schattenbildung in Teilen der Sequenz stellt für konventionelle Verfahren ein großes Problem dar, da z.B. Differenzbildmethoden, die in [Jan97] verwendet wurden, auch in dem Schattenbereich die Anwesenheit einer Person signalisieren würden. Das hier verwendete Markov-Modell, das unter Verwendung von Trainingsdaten erstellt wurde, lokalisiert die Person jedoch auf robuste Weise. Weitere Ergebnisse, die mit dem vorgestellten Verfahren erzielt wurden, sind in [Rig99b, Rig99a] zu finden. In diesen Arbeiten sind auch Sequenzen zu finden, die mit einer bewegten Kamera aufgenommen wurden.



Abbildung 5.19: Ausschnitt aus einer Bildsequenz, die eine sich bewegende Person zeigt. Die blau eingezeichneten Rechtecke visualisieren das Trackingergebnis (aus [Win98]).

5.5 Kapitelzusammenfassung

Ein im Rahmen dieser Arbeit entwickelter Ansatz, der die integrierte Segmentierung und Klassifikation von in komplexen Umgebungen eingebetteten Mustern ermöglicht, wurde vorgestellt. Der Ansatz verwendet pseudo zweidimensionale Hidden-Markov-Modelle in Kombination mit an die Umgebung angepaßten Umgebungszuständen. Nach der Anwendung des Viterbi-Algorithmus liegt eine Merkmal-Zustandszuordnung, die als Segmentierung interpretiert werden kann, sowie ein Schätzwert für die Produktionswahrscheinlichkeit vor. Unter Verwendung dieses Schätzwertes erfolgt die Musterklassifikation.

Die Anpassung der Parameter der Umgebungszustände kann auf verschiedene Weisen erfolgen, je nachdem, ob Vorwissen über die zu analysierende Szene vorliegt oder nicht. Für den letztgenannten Fall wurde ein Verfahren entwickelt, bei dem die Parameter der Umgebungszustände auf allen Merkmalen des zu analysierenden Bildes bestimmt wurden.

Das Verfahren wurde zunächst mit Hilfe einer Piktogrammdatenbasis, die aus 20 Klassen besteht, evaluiert. Weitere Experimente sind beschrieben, die die Eignung des P2DHMM-Ansatzes für das Retrieval von Formen in technischen Zeichnungen belegen. Somit ist es möglich, z.B. eine mit einer Skizze spezifizierte Schraube in komplexen technischen Zeichnungen aufzufinden. Schließlich wurde der P2DHMM-Ansatz für das Personen-Tracking eingesetzt. Dabei konnte gezeigt werden, daß Muster bzw. Personen auch in Grauwertbildern bzw. Farbbildern mit dem vorgestellten Ansatz gefunden werden können. Es zeigte sich ebenfalls, daß der Ansatz gut kombinierbar ist mit einem sog. Kalman-Filter, das die Dynamik der Bewegung der Person modelliert.