

# Erkennung von zusammengesetzten Objekten in Bildern unter Echtzeit-Anforderungen

Markus Ulrich, Carsten Steger, Albert Baumgartner und Heinrich Ebner

## Zusammenfassung

In der Industrie wächst der Bedarf an intelligenten und effizienten Verfahren zur Automatisierung der Fertigungsprozesse. Auch die moderne Photogrammetrie kann auf diesem Gebiet mit Methoden des Bildverstehens (Machine Vision) wichtige Beiträge leisten. Häufig müssen Objekte in Bildern automatisch erkannt werden. Durch den industriellen Hintergrund werden sehr hohe Ansprüche gestellt: Um einen möglichst hohen Produktionsdurchsatz zu gewährleisten, ist es nötig, die Erkennung in Echtzeit durchzuführen. Außerdem treten bei vielen Anwendungen Objekte auf, die sich aus beweglichen Einzelteilen zusammensetzen, was eine Erkennung wesentlich erschwert. Dieser Artikel stellt ein Verfahren vor, das es erlaubt, solche zusammengesetzten Objekte in Echtzeit zu erkennen, ohne dabei auf Spezial-Hardware angewiesen zu sein. Das Verfahren basiert auf einer hierarchischen Modellierung der zu erkennenden Objekte, was entscheidende Vorteile in der Leistungsfähigkeit mit sich bringt. Es zeichnet sich außerdem durch einen hohen Automationsgrad aus, wodurch es den Benutzer entlastet und einen großen Anwenderkreis anspricht.

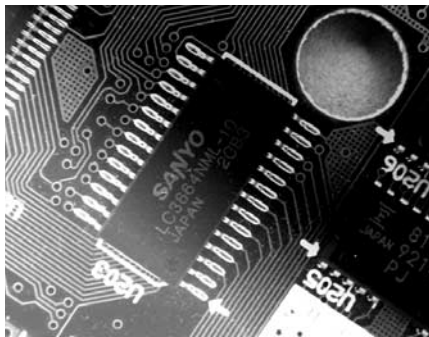
## Summary

*In industry, there is an increasing demand for intelligent and efficient approaches in order to automate the production process. Modern photogrammetry can make a substantial contribution in this field by using methods of machine vision. Often, objects must be automatically recognized in images. The requirements that are induced by the industrial background are very high: To facilitate a fast production process, it is necessary to recognize objects in real-time. Furthermore, objects that consist of several moving parts occur in many applications. These compound objects complicate the recognition considerably. This paper proposes a novel approach for recognizing compound objects in real-time while only using off-the-shelf hardware. The approach is based on a hierarchical model of the compound object. This leads to substantial improvements of the achieved performance. A further feature of the approach is its high degree of automation in order to achieve an easy use, and hence to facilitate a broad applicability.*

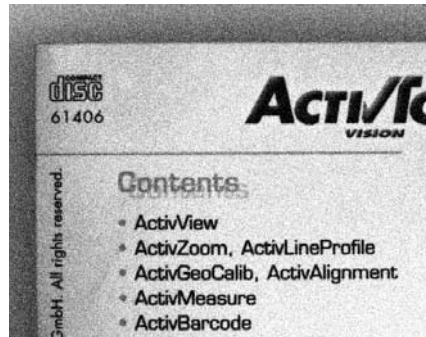
## 1 Einleitung

Die hohe Relevanz des fortschreitenden Automatisierungsprozesses auf dem Gebiet der industriellen Produktion ist unumstritten. Der bereits existierende hohe Automationsgrad kann u. a. dem Fortschritt auf dem Gebiet der Bildverarbeitung und des Bildverstehens (Machine Vision) zugeschrieben werden.

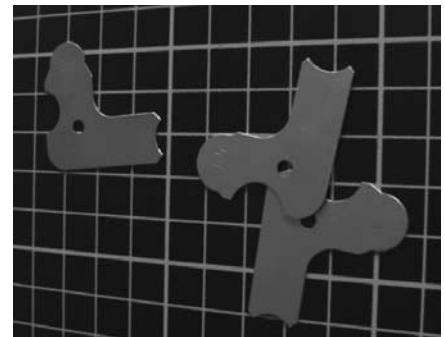
In der Industrie werden oft bildgebende Sensoren für die automatische Qualitätskontrolle oder Robotersteuerung (Pick and Place) eingesetzt. Um Aufgaben dieser Art zu bewerkstelligen, ist es notwendig, das entsprechende Objekt im Bild zu erkennen und zu lokalisieren, d. h. seine Lage (Position und Orientierung) zu bestimmen. Die vorliegende Arbeit beschränkt sich auf die Erkennung von 2D-Objekten, da eine aufwändigere Erkennung von 3D-Objekten für die meisten Anwendungen unnötig ist (Steger 2001). In der in Abb. 1(a) gezeigten Anwendung gilt es, die Abstände zwischen einzelnen Kontakten des Chips zu vermessen, um Kurzschlüssen vorzubeugen. Voraussetzung für die Vermessung ist, dass die Lage des Chips im Bild bekannt ist. Außerdem wird deutlich, dass die Erkennung auch unter schwierigen Lichtverhältnissen erfolgreich verlaufen muss. In Abb. 1(b) besteht die Aufgabe im Lesen der Ziffern unterhalb des »disc-Schriftzuges« mit Hilfe einer OCR (optical character recognition). Durch Lokalisieren des »disc-Schriftzuges« durch das Objekterkennungsverfahren kann die OCR auf den darunter liegenden Bildbereich eingeschränkt werden und kann sich nach einer entsprechenden Rotation des Bildbereiches auf das Lesen horizontal ausgerichteter Schriften beschränken. Wie aus diesem Beispiel hervorgeht, ist Robustheit gegenüber Bildrauschen eine weitere zu fordernde Eigenschaft. In Abb. 1(c) müssen Werkstücke von einem Roboter aufgegriffen werden. Mit Hilfe einer Kamerakalibrierung ist es möglich, die von der Objekterkennung erhaltenen Lagekoordinaten in Weltkoordinaten umzurechnen und dem Roboter zur Verfügung zu stellen. Eine weitere Anforderung, die sich aus diesem Beispiel ergibt, ist die Erkennung von projektiv verzerrten Objekten im Bild. Projektive Verzerrungen treten auf, wenn die Kamera konstruktionsbedingt nicht direkt über der Objektebene angebracht werden kann und damit die Blickrichtung nicht senkrecht zur Objektebene ist. Die Objektebene beschreibt dabei die Ebene, in der sich die Objekte bewegen können. Ebenso wichtig ist es, teilweise verdeckte Objekte oder mehrere auftretende Objektinstanzen im Bild zu finden. Die bei weitem am schwierigsten zu verwirklichende industrielle Anforderung ist allerdings, das Verfahren neben den oben genannten Anforderungen zusätzlich echtzeitfähig zu machen, um den Produktionsdurchsatz zu maximieren. Die Bedeutung des Begriffes »Echtzeit« lässt sich aus der maximalen Bildwiederholfrequenz von 33 ms handelsüblicher Kameras ableiten. Aufgrund der Tatsache, dass in den meisten Anwendungen z. B. nur jedes dritte oder vierte Bild neue Informationen enthält, ist es legitim, dem Verfahren eine Erkennungszeit von weniger als ca. 100 ms zuzugestehen (aus-



(a) Qualitätskontrolle



(b) OCR



(c) Pick &amp; Place

Abb. 1: Beispiele für den Einsatz der Objekterkennung in der Industrie

gehend von einer Standardbildgröße von  $640 \times 480$  Pixel und einem handelsüblichen Computer, z. B. 2 GHz Pentium 4). Die Photogrammetrie hat sich in den letzten Jahren stark auf dem Gebiet des Bildverstehens engagiert. Dabei lagen die Schwerpunkte bislang hauptsächlich auf der Automatisierung der Bildorientierungen, der Ableitung von digitalen Geländemodellen oder Orthofotos, der Erkennung von Objekten aus Luft- oder Satellitenbildern (z. B. Fahrzeuge, Straßen, Gebäude, Vegetation) oder auf der 3D-Rekonstruktion von Objekten aus terrestrisch aufgenommenen Bildern (z. B. Denkmäler, Kunstgegenstände). Aufgaben dieser Art stellen keine besonderen Ansprüche an die zulässige Rechenzeit und werden zumeist offline im »post processing« durchgeführt. Im Gegensatz dazu ist der Rechenzeitaspekt bei industriellen Anwendungen – wie oben erwähnt – wesentlich kritischer.

Bei den in Abb. 1 vorgestellten Beispielen handelt es sich um starre Objekte. Zusammengesetzte Objekte hingegen bestehen wiederum aus mehreren starren Objektteilen, die sich relativ zueinander bewegen können, z. B. durch gegenseitiges Verschieben oder Verdrehen. Abb. 2 zeigt ein typisches Beispiel. Das Etikett, als ein zusammengesetztes Objekt, besteht aus den folgenden fünf starren Objektteilen: Rahmen, Schriftzug »BEST BEFORE END:« und drei Ziffernpaare des Mindesthaltbarkeits-

datums. Die beiden Schrägstriche sind dabei starr mit dem von ihnen jeweils rechts stehenden Ziffernpaar verbunden und stellen somit keine separaten Objektteile dar. Deutlich sind die relativen Lageänderungen der Objektteile in den Bildern zu erkennen. Diese sind darauf zurückzuführen, dass beim Druckverfahren unterschiedliche Stempel verwendet wurden, die nicht exakt ausgerichtet werden können. Ein ähnliches Problem tritt auch beim häufig verwendeten Mehrfarbendruck auf, bei dem für jede Farbe ein eigener Stempel verwendet wird. Ziel einer Qualitätskontrolle könnte es zum Beispiel sein, zu gewährleisten, dass nur einwandfrei bedruckte Produkte ausgeliefert werden. Dazu müssen die einzelnen Objektteile im Bild lokalisiert und auf ihre korrekte relative Lage kontrolliert werden.

Herkömmliche Verfahren sind in der Lage, starre Objekte zu erkennen. Der Einsatz dieser Verfahren ist bei zusammengesetzten Objekten jedoch nicht zufriedenstellend: Wird das Etikett als ein starres Objekt betrachtet, versagt eine Erkennung, da das Erscheinungsbild des Etikettes aufgrund der relativen Lageänderungen der Objektteile variiert. Eine einfache Methode, dieses Problem zu umgehen, ist, die einzelnen Objektteile separat zu suchen. Allerdings kann dabei vorhandenes Vorwissen über die ungefähre relative Lage der Objektteile nicht genutzt

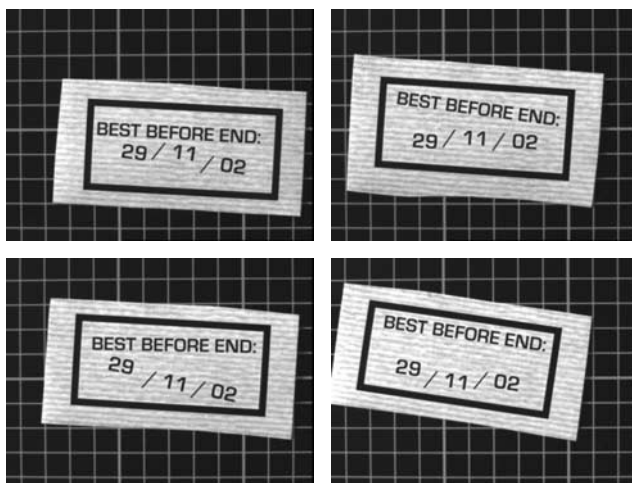


Abb. 2: Ein Etikett als typisches Beispiel für ein zusammengesetztes Objekt

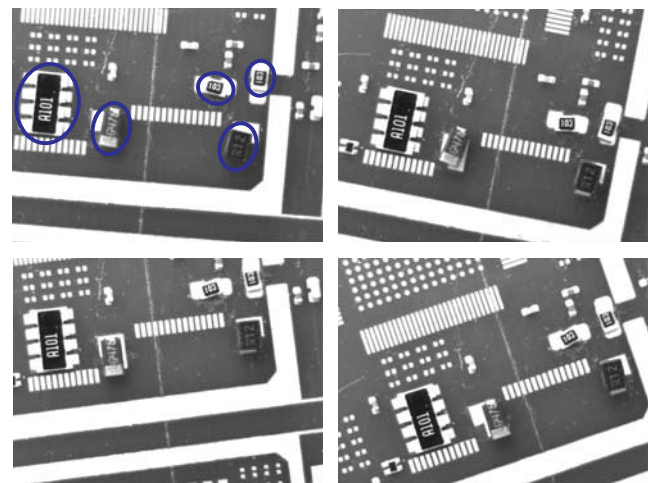


Abb. 3: Die Module auf der Platine bilden ein zusammengesetztes Objekt.

werden (z. B. Datum steht unterhalb des Schriftzuges und beide befinden sich innerhalb des Rahmens). Diese mangelnde Berücksichtigung des Vorwissens führt unweigerlich zu einem Effizienzverlust und verhindert unter Umständen die Echtzeitfähigkeit des Verfahrens.

Abb. 3 zeigt ein weiteres Beispiel für ein zusammengesetztes Objekt. Hier besteht das zusammengesetzte Objekt aus fünf Modulen auf einer Platine. Bei genauerer Betrachtung sind die relativen Lageverschiebungen zu erkennen, die auf die Ungenauigkeit der Bestückungsmaschine zurückzuführen sind. Auch hier ist der Einsatz eines Verfahrens zur Erkennung starrer Objekte entweder überhaupt nicht möglich oder zumindest problematisch, da die Echtzeitanforderung kaum erfüllt werden kann.

## 2 Bestehende Verfahren

Verfahren, die sich mit der Erkennung zusammengesetzter Objekte befassen, sind bislang relativ selten zu finden. In Grimson (1989), Lowe (1991), Li und Lee (2002) werden Verfahren zur Erkennung sog. »articulated objects« vorgestellt. Diese setzen voraus, dass die Einzelteile über Gelenke miteinander verbunden sind, was für die oben genannten Anwendungen zu restriktiv ist. In den meisten Verfahren werden die Objektteile in einem ersten Schritt unabhängig voneinander erkannt. Erst in einem zweiten Schritt wird überprüft, ob die Beziehungen zwischen den erkannten Objektteilen mit denen des Modells übereinstimmen (Grimson 1987, Grimson 1989, Kratchounova 1996). Diese Verfahren bestechen zwar durch ihre Einfachheit, sind aber ineffizient in der Hinsicht, dass die Information über die Beziehungen zwischen den Objektteilen nicht optimal genutzt wird. Diese Information wird in Hel-Or und Werman (1994a und 1994b) verwendet, um auftretende Mehrdeutigkeiten zu lösen. Dazu wird in einem rekursiven Prozess die wahrscheinlichste Lage eines Objektteils aus den bereits eindeutig erkannten Objektteilen ermittelt und mit dem Resultat der Suche verglichen. Voraussetzung ist, dass für jedes Objektteil bereits alle möglichen Lagen im Bild gefunden wurden und somit auch hier die Suche nicht von der Vorinformation profitiert.

Eine weitere Gruppe von Verfahren repräsentiert zusammengesetzte Objekte in einem Graphen. Dabei stellen die Knoten im Graphen die Objektteile und die Kanten die Beziehungen zwischen den Objektteilen dar. Durch Graph-Matching kann das zusammengesetzte Objekt erkannt werden (Li und Lee 2002, Suganthan 2002). Allerdings ist der Prozess des Graph-Matchings äußerst rechenintensiv und daher für Echtzeit Objekterkennung nicht geeignet. Ein ähnliches Verfahren wird in Felzenszwalb und Huttenlocher (2000) genutzt, bei dem das Objekt als deformierbare Anordnung von Einzelteilen beschrieben wird. Die Deformationen werden durch Federkräfte modelliert. Die wahrscheinlichste Position des

Objektes in einem Bild wird durch das Minimum an aufzubringenden Deformationskräften beschrieben. Dieses Verfahren zeigt allerdings keinen Vorteil bezüglich der Rechenzeit im Vergleich zur separaten Suche aller Objektteile.

Eine andere Kategorie von Verfahren befasst sich mit der Erkennung von elastischen, deformierbaren oder flexiblen Objekten (Jain et al. 1996, Pilu und Fischer 1997, Lee und Street 2000, Sclaroff und Liu 2001, Belongie et al. 2002, Gavrilu und Philomin 1999). Diese Verfahren können u. a. dazu genutzt werden, Abbildungen menschlicher Organe zu registrieren (z. B. Kernspintomographie, Computertomographie, Ultraschall). Für die Erkennung zusammengesetzter Objekte eignen sich diese allerdings nur bedingt, da zusammengesetzte Objekte keine tatsächlichen Deformationen zeigen, sondern aus starren Objektteilen bestehen.

## 3 Überblick über das Verfahren

In diesem Abschnitt wird das neue Verfahren zur Erkennung zusammengesetzter Objekte in Echtzeit vorgestellt. In den darauf folgenden Abschnitten werden die einzelnen Schritte des Verfahrens näher erläutert.

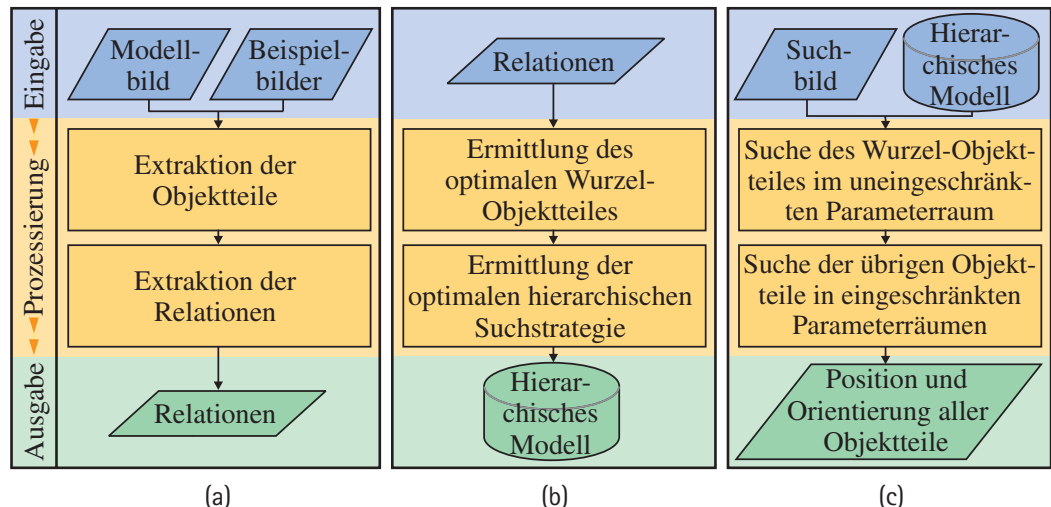
Die zu Grunde liegende Strategie kann in eine Vorstufe und drei Hauptstufen gegliedert werden. Die vom Rest des Verfahrens unabhängige Vorstufe dient der Eliminierung projektiver Verzerrungen des Bildes. Der Vorteil dieses Vorgehens ist, dass die Algorithmen der darauf folgenden Stufen von projektiv entzerrten Bildern ausgehen können und somit wesentlich einfacher und effizienter konzipiert werden können. Die wesentliche Neuheit dieser Entzerrung ist, dass die Eliminierung der projektiven Verzerrungen in Echtzeit erfolgt und somit auch für eine Vielzahl anderer zeitkritischer Verfahren einsetzbar ist. Die Entzerrung erfolgt auf der Grundlage einer zuvor durchgeführten Kamerakalibrierung, und ermöglicht es somit, auch radiale Verzeichnungen des Kameraobjektives effizient zu eliminieren.

Der Hauptteil des Verfahrens basiert auf einer hierarchischen Modellierung des zusammengesetzten Objektes (Ulrich et al. 2002). Abb. 4 zeigt einen groben Überblick über die drei Hauptstufen des Verfahrens.

Die erste Stufe umfasst das Trainieren des hierarchischen Modells auf der Grundlage eines Modellbildes und einiger Beispielbilder (siehe Abb. 4(a)). Im Modellbild gibt der Benutzer die Bildregion (region of interest (ROI)) an, die das komplette zusammengesetzte Objekt beinhaltet. Innerhalb der Beispiel-Bilder sollten die gegenseitigen Bewegungen der Objektteile zum Ausdruck kommen, wie z. B. in den Abbildungen 2 oder 3. Aufgrund dieser Daten werden dann die starren Objektteile und deren gegenseitige Beziehungen automatisch extrahiert. Das Ergebnis des Trainierens ist somit nicht das komplette hierarchische Modell, sondern beinhaltet lediglich die Relationen



Abb. 4: Flussdiagramme der drei Stufen zur Erkennung zusammengesetzter Objekte. Die Offline-Phase beinhaltet das Trainieren (a) und Aufbauen (b) des hierarchischen Modells. Die Objekterkennung (c) repräsentiert die Online-Phase.



zwischen den Objektteilen. Erst in der zweiten Stufe wird das hierarchische Modell auf Grundlage der trainierten Relationen aufgebaut (siehe Abb. 4(b)). Dazu wird nach bestimmten Kriterien eines der Objektteile ausgewählt. Dieses wird im Folgenden als Wurzel-Objektteil bezeichnet. Des Weiteren wird die optimale hierarchische Suchstrategie ermittelt, an deren Spitze das Wurzel-Objektteil steht. Das fertige hierarchische Modell besteht somit aus den Objektteilen, den Relationen und der hierarchischen Suchstrategie. In der dritten Stufe wird das hierarchische Modell genutzt, um das zusammengesetzte Objekt in einem Suchbild zu erkennen (siehe Abb. 4(c)): Um eine Erkennung in Echtzeit zu gewährleisten, wird lediglich das Wurzel-Objektteil im gesamten Parameterraum gesucht. Für die übrigen Objektteile hingegen kann der Suchraum stark eingeschränkt werden. Dies wird durch ein rekursives Vorgehen ermöglicht, in dem die Objektteile relativ zueinander gesucht werden. Die Reihenfolge der Suche ist durch die Suchstrategie festgelegt, die eingeschränkten Suchräume ergeben sich aus den trainierten Relationen.

Im Folgenden werden die ersten beiden Stufen auch als Offline-Phase bezeichnet, da es genügt, diese Stufen für ein bestimmtes Objekt lediglich einmal durchzuführen. Die dritte Stufe repräsentiert die zeitkritische Online-Phase, da diese für jedes Suchbild erneut ausgeführt werden muss und daher der Echtzeitanforderung gerecht werden muss.

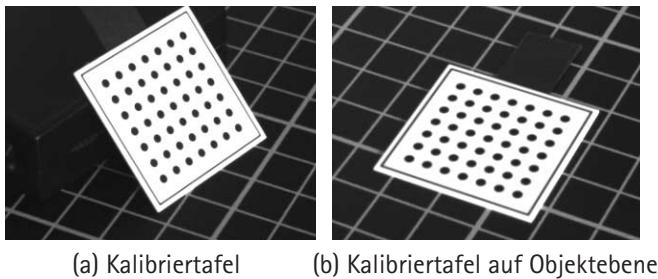
#### 4 Kamera-Kalibrierung und Entzerrung

Um projektive Verzerrungen im Bild zu eliminieren, ist es notwendig, die exakte Lage der Kamera relativ zur Objektebene zu kennen. Die Lage der Kamera kann mit Hilfe einer Kalibrierung ermittelt werden. In dieser Arbeit wird das in Lenz (1987) vorgestellte Kameramodell verwendet, das von einer Lochkamera mit radialer Verzerrung ausgeht und die Transformation zwischen Weltkoordinaten und Pixelkoordinaten beschreibt. Die Ka-

librierung umfasst die Bestimmung sowohl der äußeren Kameraparameter  $K_a$  (3D-Position, 3D-Drehung) als auch der inneren Kameraparameter  $K_i$  (Bildweite, Bildhauptpunkt, Pixelabstand auf dem Sensor, radiale Verzerrung). Das Berücksichtigen der radialen Verzerrung ist für exaktes Messen im Bild unverzichtbar. Die Durchführung der Kalibrierung orientiert sich an dem in Lanser et al. (1995) beschriebenen Verfahren. Dazu wird eine planare Kalibriertafel (siehe Abb. 5) in unterschiedliche Lagen gebracht und von der räumlich fixierten Kamera aufgenommen. Die so aufgenommenen Kalibrierbilder sollten möglichst alle Freiheitsgrade der Kalibriertafel abdecken. Die Positionen der kreisförmigen Marken auf der Kalibriertafel sind exakt vermessen und können als fehlerfreie Soll-Koordinaten angenommen werden. Nach einer automatischen Extraktion der Marken in den Kalibrierbildern können die Bildkoordinaten der Markenn Mittelpunkte über die unbekannten inneren und äußeren Kameraparameter mit den gegebenen Soll-Koordinaten in Beziehung gesetzt werden. Die Kameraparameter können letztendlich durch Minimierung der quadratischen Differenzen mit Hilfe des Levenberg-Marquardt-Algorithmus geschätzt werden (Marquardt 1963). Dabei erhält man die für alle Kalibrierbilder als konstant angenommenen inneren Kameraparameter sowie für jedes aufgenommene Kalibrierbild eine äußere Orientierung der Kamera. Um eine Entzerrung der Objektebene zu ermöglichen, muss die Kalibriertafel in mindestens einem Kalibrierbild planar auf der Objektebene liegen (siehe Abb. 5(b)). Die für dieses Bild erhaltene äußere Orientierung (korrigiert um die Dicke  $d$  der Kalibriertafel) beschreibt die Lage der Kamera relativ zur Objektebene. Damit ist die Abbildung  $\pi$  eines in Weltkoordinaten gegebenen 3D-Punktes in der Objektebene  $\vec{p}_w = (x_w, y_w, z_w = d)$  ins Pixelkoordinatensystem  $\vec{p} = (r, c)$  vollständig bekannt:

$$\vec{p} = \pi(\vec{p}_w, K_a, K_i) . \quad (1)$$

Die Entzerrung eines Bildes kann als Projektion des Bildes in die Objektebene interpretiert werden. Nach der indirekten Methode können die Pixel  $\vec{p}_w$  des entzerrten Bil-



(a) Kalibriertafel

(b) Kalibriertafel auf Objektebene

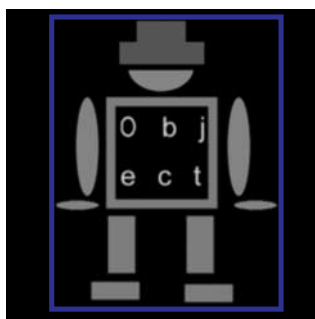
Abb. 5: Zwei Beispiele für Kalibrierbilder. Die in der Objektebene liegende Kalibriertafel (b) wird dazu genutzt, die Abbildungsvorschrift für die Entzerrung zu berechnen.

des mit (1) ins ursprüngliche (verzerrte) Bild zurücktransformiert werden. Um die Entzerrung zu beschleunigen, wird diese rechenintensive Transformation lediglich einmal durchgeführt und in einer *Look-up-Table* gespeichert. Dadurch wird eine Entzerrung von Bildern in Echtzeit ermöglicht.

Ein Beispiel für die Entzerrung ist in Abb. 6 zu sehen. Die Transformation des  $640 \times 482$  Pixel großen Bildes in ein  $515 \times 527$  Pixel großes entzerrtes Bild benötigt auf einem 2 GHz Pentium 4 lediglich 8.3 ms.

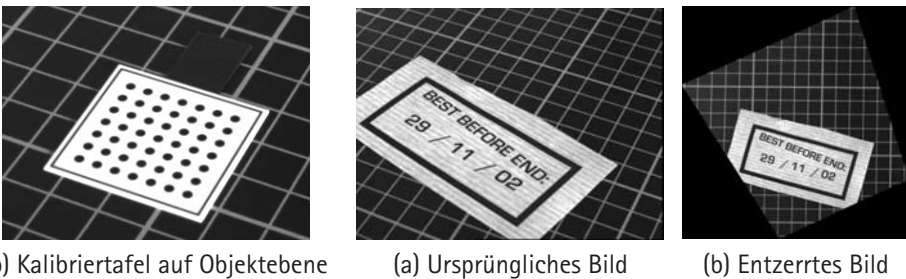
## 5 Trainieren des Hierarchischen Modells

Das Prinzip des vorgestellten Verfahrens soll anhand eines künstlichen Beispiels erläutert werden. Abb. 7 zeigt die wichtigsten Eingabedaten. Eine vom Benutzer festgelegte rechteckige ROI (blau) definiert das zusammengesetzte Objekt im Modellbild (Abb. 7(a)). Des Weiteren werden sechs Beispielbilder, die die relativen Bewegun-



(a) Modellbild

Abb. 7: Eingabedaten zum Trainieren des hierarchischen Modells



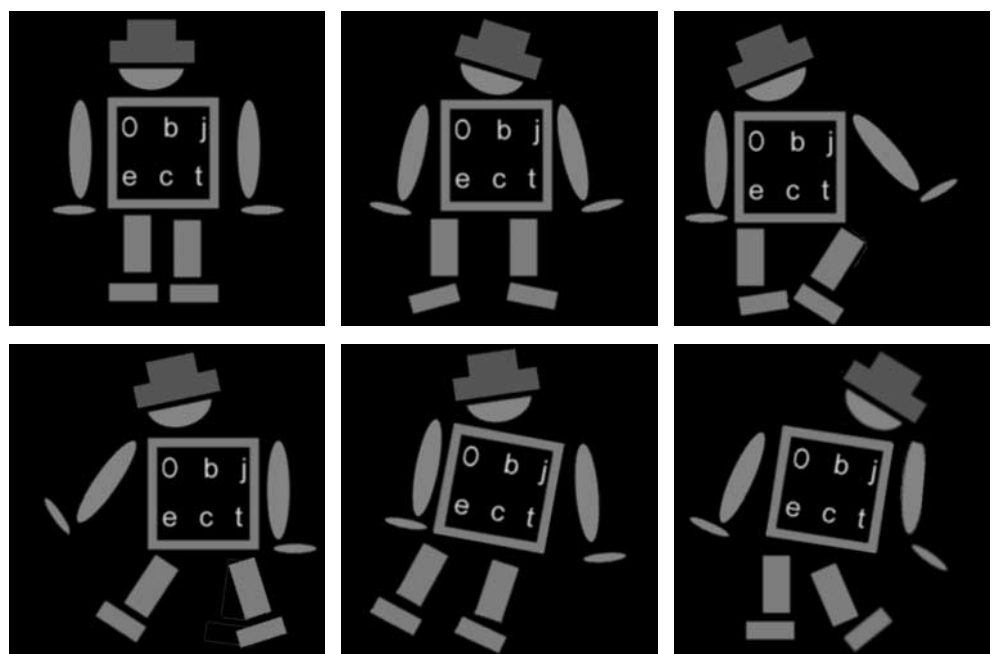
(a) Ursprüngliches Bild

(b) Entzerrtes Bild

Abb. 6: Beispiel zur Entzerrung

gen der entsprechenden Objektteile zeigen, zur Verfügung gestellt (Abb. 7(b)). Bei näherer Betrachtung der Beispielbilder können zwei Beobachtungen gemacht werden: Zum einen besteht das zusammengesetzte Objekt in diesem Beispiel offensichtlich aus mehreren starren Objektteilen (Kopf, Oberkörper, zwei Arme, zwei Hände, zwei Beine und zwei Füße). Zum anderen bewegen sich die Objektteile relativ zueinander in unterschiedlichem Ausmaß. Ziel des Trainierens ist es nun, den Prozess dieser menschlichen visuellen Wahrnehmung nachzubilden. Dieser Prozess umfasst die Extraktion der starren Objektteile sowie deren Relationen.

Für die folgenden Diskussionen ist es wichtig, zwischen Komponenten und (starken) Objektteilen zu unterscheiden. Ein starrer Objektteil besteht aus einer oder mehreren Komponenten. Die Komponenten, die zum selben Objektteil gehören, zeigen keine gegenseitigen Bewegungen, während sich Objektteile zueinander bewegen. Blenden wir bewusst Vorwissen über das Aussehen einer menschlichen Person aus und konzentrieren uns auf die Analyse des Modellbildes, gelingt es uns nicht, starre Objektteile zu erkennen. Es ist lediglich möglich, das Objekt in kleine Komponenten zu zerlegen: Hut, Gesicht, zwei



(b) Beispielbilder

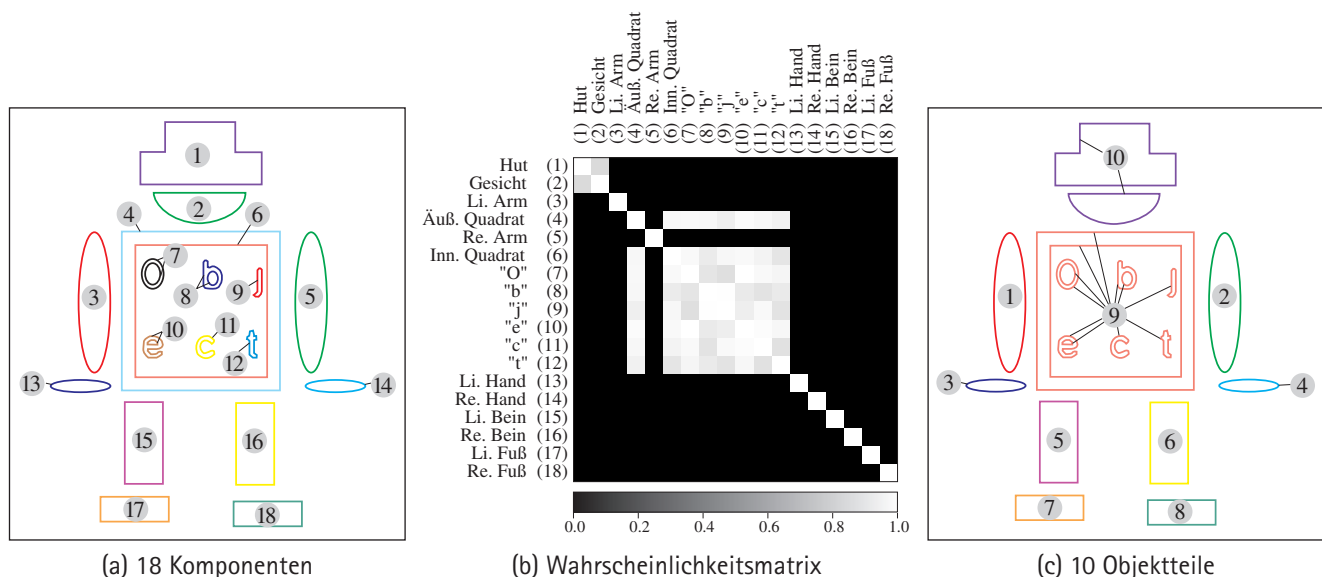


Abb. 8: Die extrahierten Komponenten (a) werden mit Hilfe der Wahrscheinlichkeitsmatrix (b) zu starren Objektteilen (c) geclustert.

Arme, zwei Hände, zwei Beine, zwei Füße, quadratischer Umriss des Oberkörpers und eine Komponente für jeden Buchstaben auf dem Oberkörper. Die Zerlegung des Objektes in Komponenten erfolgt in diesem Beispiel offensichtlich auf der Basis homogener Grauwerte. Weiterhin wird die Betrachtung auf die Beispielbilder aus, so versuchen wir unbewusst, die Komponenten aus dem Modellbild in den Beispielbildern wiederzuerkennen (Ullman 1979). Letztendlich verschmelzen diejenigen Komponenten, die in keinem der Bilder relative Bewegungen zeigen, automatisch zu starren Objektteilen.

Mit Hilfe dieses Wissens ist es möglich, den Prozess der menschlichen Wahrnehmung zu simulieren. Zunächst werden im Bildbereich, der durch die ROI im Modellbild definiert ist, kleine Komponenten extrahiert. Dies kann, z. B., auf der Basis von Zusammenhangskomponenten extrahierter Kantenpixel (Jähne 2002) geschehen (siehe Abb. 8(a)). Mit Hilfe eines geeigneten Verfahrens zur Erkennung starrer Objekte, wird versucht, die Komponenten in den Beispielbildern wiederzufinden. Werden die Komponenten durch Kantenpixel beschrieben, so sollte das Objekterkennungsverfahren ebenfalls Kanten als Merkmal verwenden. Sehr gut geeignete Verfahren sind z. B. das formbasierte Matching (Steger 2002) und die modifizierte generalisierte Hough-Transformation (Ulrich et al. 2001) aufgrund ihrer hohen Robustheit und Echtzeitfähigkeit. Für die Implementierung des hier vorgestellten Verfahrens wurde das formbasierte Matching gewählt, da es eine höhere Robustheit gegenüber extremen Beleuchtungsunterschieden aufweist. Aufgrund von Rotationssymmetrien der Komponenten, Ähnlichkeiten der Komponenten zu anderen Komponenten oder zu anderen Bildstrukturen kann es vorkommen, dass eine Komponente in einem Beispielbild mehrmals gefunden wird. So wird z. B. der linke Fuß in jedem Beispielbild in vier verschiedenen Lagen gefunden: an der Stelle des linken Fußes in zwei um 180° verschiedenen Orientierungen

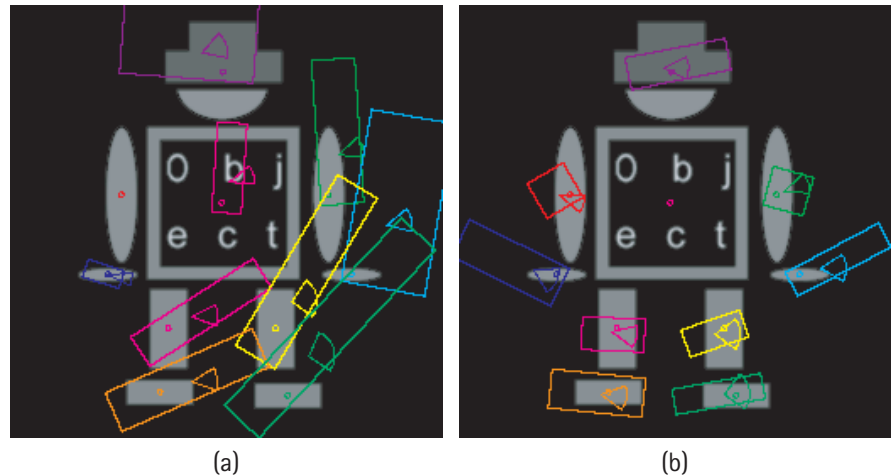
und an der Stelle des rechten Fußes ebenfalls in zwei Orientierungen. Diese Mehrdeutigkeiten müssen vor einer weiterführenden Analyse gelöst werden. Dazu werden in jedem Beispielbild alle gefundenen Instanzen (Matches) einer Komponente bewertet. Die (eindeutige) relative Lage der Komponenten im Modellbild dient dabei als Referenz-Konfiguration. Der Match  $k$  einer Komponente  $i$  im Beispielbild wird mit der Lage der Komponente im Referenzbild verglichen. Die Matches, die zu geringeren Veränderungen der Gesamtkonfiguration bezüglich der Referenz-Konfiguration führen, erhalten einen geringeren Kostenwert  $\Psi_{i,k}$ . Die Auflösung der Mehrdeutigkeiten ist äquivalent mit dem Problem, für jede Komponente  $i$  den Match  $k$  zu finden, so dass

$$\sum_i \Psi_{i,k} \rightarrow \min \quad (2)$$

unter der Nebenbedingung, dass Mehrfachzuordnungen vermieden werden, d. h. dass verschiedene Komponenten nicht derselben Struktur im Bild zugeordnet werden dürfen. Eine effiziente Lösung dieser Optimierungsaufgabe wird durch die Darstellung des Problems mit Hilfe eines bipartiten Graphen ermöglicht. Ein bipartiter Graph besteht aus zwei Knotenmengen, wobei nur Knoten in unterschiedlichen Knotenmengen durch Kanten verbunden sind. Die erste Knotenmenge repräsentiert die Komponenten, die zweite die Strukturen im Beispielbild, denen Komponenten zugeordnet wurden. Die Kanten im Graphen stellen die Matches dar und sind mit dem jeweiligen Kostenwert  $\Psi_{i,k}$  gewichtet. Durch bipartites Graph-Matching (z. B. mit dem Simplex-Algorithmus (Press et al. 1992)) kann die in (2) formulierte Optimierungsaufgabe äußerst effizient gelöst werden. Damit wird jeder Komponente in jedem Beispielbild ein eindeutiger Match zugewiesen.

Durch paarweise Analyse der Komponenten ist es nun möglich, zu entscheiden, ob zwei Komponenten zum sel-

Abb. 9: Relationen des linken Armes (a) und des Oberkörpers (b) ins Modellbild projiziert. Die Objektteile sind durch ihre Referenzpunkte symbolisiert. Die Relationen sind als Rechtecke (Positions-Relation) und Kreissektoren (Orientierungs-Relation) dargestellt.



ben Objektteil gehören. Dazu wird die Hypothese aufgestellt, dass sich beide Komponenten in allen Beispielbildern gleich bewegen und damit einem starren Objektteil zugehören. Mit Hilfe eines Hypothesentests (Koch 1987) kann die Wahrscheinlichkeit dafür berechnet werden, dass zwei Komponenten diese Bedingung erfüllen. In der in Abb. 8(b) gezeigten symmetrischen Matrix sind die entsprechenden Wahrscheinlichkeiten als Grauwerte visualisiert. Durch sukzessives paarweises Clustern der Matrix (Jain et al. 1999) unter Verwendung einer Mindestwahrscheinlichkeit können die Komponenten zu starren Objektteilen zusammengeführt werden (siehe Abb. 8(c)). Nun können die Relationen zwischen den Objektteilen bestimmt werden. Dazu wird für jedes Objektteil-Paar  $(i, j)$  die Lage des zweiten Objektteils  $j$  relativ zur Lage des ersten Objektteils  $i$  ausgedrückt. Über die Sequenz an Beispielbildern ergibt sich somit eine scheinbare Bewegung des zweiten Objektteils relativ zum ersten. Die relative Positionsänderung wird dabei auf den Referenzpunkt (z.B. der Schwerpunkt der Kantenpixel) der Objektteile bezogen. Das kleinste umschließende Rechteck um die Positionen des Referenzpunktes beschreibt die Positions-Relation. Analog dazu wird die Orientierungs-Relation durch das kleinste umschließende Winkelintervall der Orientierungsänderungen des zweiten Objektteils über die Sequenz beschrieben. Im Folgenden werden die Seitenlängen des Rechtecks mit  $a_{i,j}$  und  $b_{i,j}$  und das Winkelintervall mit  $\Delta\alpha_{i,j}$  bezeichnet. Abb. 9 zeigt die Relationen des linken Armes und des Oberkörpers zu den übrigen Objektteilen. Es wird z.B. deutlich, dass die Bewegung der linken Hand relativ zum linken Arm geringer ist als die relativen Bewegungen der übrigen Objektteile. Außerdem ist zu erkennen, dass die Bewegungen relativ zum Oberkörper durchschnittlich kleiner sind als die Bewegungen relativ zum linken Arm.

## 6 Aufbau des Hierarchischen Modells

In der zweiten Stufe der Offline-Phase werden die in der ersten Stufe extrahierten Relationen dazu genutzt, um

das hierarchische Modell aufzubauen. Zunächst wird für jedes Objektteil unter Verwendung des formbasierten Matchings (Steger 2002) ein starres Modell erzeugt. Die Relationen zwischen den Objektteilen können zur Berechnung des in der Online-Phase aufzubringenden Suchaufwandes herangezogen werden. Dabei wird davon ausgegangen, dass das Ausmaß der relativen Bewegungen während der Objekterkennung in der Online-Phase dem Ausmaß der relativen Bewegungen in den Beispielbildern während des Trainierens in der Offline-Phase entspricht. Sollte diese Bedingung nicht erfüllt sein, können an die automatisch trainierten Relationen manuell Toleranzwerte angebracht werden. Der Suchaufwand  $\Omega_{i,j}$ , um Objektteil  $j$  relativ zu Objektteil  $i$  zu suchen, ist somit proportional zu den Relationen zwischen beiden Objektteilen:

$$\Omega_{i,j} \propto a_{i,j} \cdot b_{i,j} \cdot \Delta\alpha_{i,j} . \quad (3)$$

Um einen absoluten Wert für den Suchaufwand zu erhalten, müssen zusätzlich die Eigenschaften des verwendeten Verfahrens zur Erkennung starrer Objekte (hier formbasiertes Matching) mit einbezogen werden. Die Aufgabe, die optimale Suchstrategie für ein vorgegebenes Wurzel-Objektteil  $r$  zu finden, ist äquivalent zur Minimierung des Gesamt-Suchaufwandes  $\Omega$ :

$$\Omega = \sum_{j=1, j \neq r}^n \Omega_{i(j),j} \rightarrow \min . \quad (4)$$

Dabei ist  $n$  die Anzahl an Objektteilen und  $\Omega_{i(j),j}$  der Aufwand, Objektteil  $j$  relativ zu seinem Vorgänger  $i(j)$  zu suchen. Die Suchstrategie lässt sich somit in Form eines Suchbaumes darstellen, an dessen Spitze das Wurzel-Objektteil  $r$  steht. Anschaulich formuliert, besteht die Lösung von (4) darin, für jedes Objektteil  $i \neq r$  einen Vorgänger  $i(j)$  zu finden, so dass  $\Omega$  minimiert wird. Dieses Optimierungsproblem kann in Form eines Graphen dargestellt werden, in dem die Knoten die Objektteile repräsentieren und die Kanten zwischen zwei Knoten mit dem relativen Suchaufwand gewichtet sind (siehe Abb. 10(a)). Da der Suchaufwand nicht symmetrisch ist,



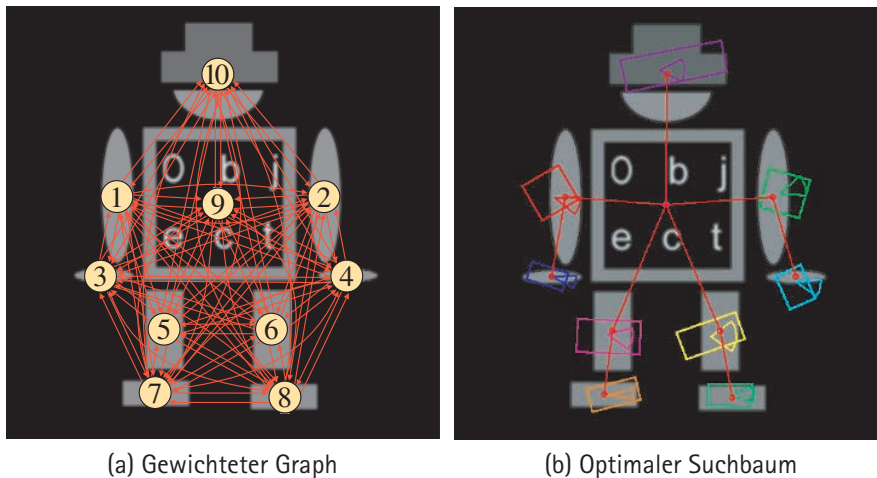


Abb. 10: Der relative Suchaufwand kann als gewichteter Graph dargestellt werden (a). Der optimale Suchbaum beschreibt die Suchreihenfolge, unter der der Gesamt-Suchaufwand minimal wird (b).

d. h.  $\Omega_{i,j} \neq \Omega_{j,i}$  handelt es sich dabei um einen sog. gerichteten Graphen. Der optimale Suchbaum kann dann durch die Berechnung der sog. *minimum spanning arborescence* (entspricht dem *minimum spanning tree* (Graham und Hell 1985) im ungerichteten Graphen) gefunden werden. Ein effizienter Algorithmus ist in Fischetti und Toth (1993) zu finden. Abb. 10(b) zeigt das Ergebnis für den Fall, dass der Oberkörper als Wurzel-Objektteil ausgewählt wurde. Das Ergebnis führt zu dem Schluss, dass es am effektivsten ist, den Kopf, die Arme und die Beine relativ zur Lage des Oberkörpers zu suchen und anschließend die Erkennung der Hände und Füße relativ zur Lage der Arme bzw. Beine vorzunehmen. Demzufolge genügt es, in der späteren Online-Phase die Referenz-Punkte der einzelnen Objektteile innerhalb der dargestellten rechteckigen Positions-Relationen und innerhalb der durch die Kreissektoren visualisierten Orientierungs-Relationen zu suchen.

Die Wahl des Wurzel-Objektteils kann automatisch erfolgen. Dabei wird für jedes Objektteil der zugehörige optimale Suchbaum berechnet. Der mit dem optimalen Suchbaum verbundene Suchaufwand zusammen mit dem Aufwand, das Wurzel-Objektteil zu suchen, dient dabei als Auswahlkriterium.

Letztendlich besteht das hierarchische Modell aus den starren Modellen für die Objektteile, den Relationen und der optimalen Suchstrategie.

## 7 Objekterkennung

Die wichtigsten Eingabedaten in die Online-Phase sind das Suchbild und das hierarchische Modell. Zunächst wird das starre Modell des Wurzel-Objektteils ausgewählt und im gesamten vom Benutzer vorgegebenen Parameterraum gesucht (z. B. im gesamten Bild und in beliebiger Orientierung). Der Suchraum der verbleibenden  $n-1$  Objektteile kann dann stark eingeschränkt werden. Der Suchraum für ein Objektteil wird dabei aus der Lage des Vorgänger-Objektteils im Suchbaum und den Relationen zwischen diesen beiden Objektteilen bestimmt. Dieser

Vorgang wird rekursiv durchgeführt bis alle Objektteile gesucht wurden.

Um einen ersten Eindruck von der hohen Effizienz des Verfahrens zu erhalten, wird die Erkennung für ein Suchbild sowohl unter Verwendung des hierarchischen Modells als auch mit Hilfe eines Brute-Force-Ansatzes durchgeführt. Beim Brute-Force-Ansatz werden die Objektteile unabhängig voneinander im gesamten Parameterraum gesucht. Auf einem 2 GHz Pentium 4 benötigt die Erkennung unter Verwendung des hierarchischen Modells 20 ms, während der Brute-Force-Ansatz 310 ms benötigt. Außerdem wird ein zweiter Vorteil des hierarchischen Modells deutlich: Dadurch, dass die Suchräume stark eingeschränkt werden, können Probleme, die durch auftretende Mehrdeutigkeiten hervorgerufen werden würden, vermieden werden. So kann beim Brute-Force Ansatz zum Beispiel nicht zwischen den Matches des linken und des rechten Armes unterschieden werden. Ein ähnliches Problem tritt auf, wenn sich mehrere Instanzen des Objektes im Bild befinden.

Die Praxistauglichkeit des beschriebenen Prinzips wurde durch verschiedene Erweiterungen gesteigert, von denen die beiden wichtigsten genannt werden sollen: Das hierarchische Modell kann bei nicht-gefundenen Objektteilen dynamisch angepasst werden. Sollte während der Suche ein Objektteil (z. B. der rechte Arm) nicht gefunden werden, so ist es zunächst nicht möglich, die Suche für die im Suchbaum nachfolgenden Objektteile (z. B. rechte Hand) zu starten. Folglich wäre die Robustheit des Verfahrens gegenüber Verdeckungen relativ gering. Durch eine geeignete automatische Anpassung des Suchbaumes während der Objekterkennung (z. B. rechte Hand wird relativ zur Lage des Oberkörpers gesucht, wenn rechter Arm nicht gefunden wurde) lässt sich die Robustheit gegenüber Verdeckungen deutlich steigern. Wenn darüber hinaus damit gerechnet werden muss, dass das Wurzel-Objektteil nicht gefunden wird, greift eine zweite Erweiterung: Um auch in einem solchen Fall hohe Robustheit zu gewährleisten, werden sukzessive unterschiedliche Objektteile als Wurzel-Objektteil gewählt. Ausgehend von der Lage aller gefundenen Wurzel-Objektteile wird dann die relative Suche durchgeführt.



## 8 Beispiele

Die hohe Leistungsfähigkeit des Verfahrens soll anhand zweier Beispiele illustriert werden. Alle Berechnungen erfolgten auf einem 2 GHz Pentium 4.

Im ersten Beispiel soll das in Abb. 2 gezeigte Etikett unter perspektivischen Verzerrungen erkannt werden. Nach erfolgter Kamerakalibrierung wurde die Look-up-Table zur Bildentzerrung berechnet. Die Größe der entzerrten Bilder wurde dabei auf  $515 \times 527$  Pixel festgelegt. Das Trainieren des hierarchischen Modells wurde basierend auf dem entzerrten Modellbild und den entzerrten Beispielbildern durchgeführt. Das zusammengesetzte Objekt wurde zuvor durch eine rechteckige ROI, die den gesamten Aufdruck umschließt, im Modellbild definiert. In Abb. 11(a) sind das entzerrte Modellbild und die automatisch extrahierten Komponenten dargestellt. Nach Analyse der relativen Bewegungen in 18 Beispielbildern konnten die Komponenten zu fünf Objektteilen zusammengefasst werden (vgl. Abb. 11(b)). Der optimale Suchbaum und die entsprechenden Relationen sind in Abb. 11(c) zu sehen: Als Wurzel-Objektteil wurde der rechteckige Rahmen gewählt, relativ zu dem der Schriftzug und der mittlere Teil des Datums gesucht werden. Letztendlich werden ausgehend von der Lage des mittleren Teiles die beiden übrigen Teile des Datums gesucht. Abbildungen 11(d)–(g) zeigen einige Suchbilder, in denen das Etikett wieder gefunden werden soll. Die Online-Phase besteht somit aus der Entzerrung der Suchbilder und der anschließenden

Objekterkennung mit Hilfe des hierarchischen Modells. Um das Ergebnis der Objekterkennung zu validieren, wurden die extrahierten Kanten der Objektteile in der jeweils gefundenen Lage in das originale Suchbild projiziert. Es ist deutlich zu erkennen, dass trotz der starken projektiven Verzerrung und der relativen Bewegungen der Objektteile alle Instanzen korrekt gefunden werden konnten. Die Erkennung benötigte eine durchschnittliche Rechenzeit von lediglich 51 ms. Davon sind ca. 8 ms für die Entzerrung, 33 ms für die Suche nach dem Wurzel-Objektteil und 10 ms für die Suche der übrigen Teile zu veranschlagen. Ohne die Verwendung des hierarchischen Modells würde die Suche 512 ms benötigen. Demnach ist in diesem Beispiel eine Geschwindigkeitssteigerung um den Faktor 12 durch die Nutzung des hierarchischen Modells im Vergleich zum Brute-Force-Ansatz zu verbuchen.

Im zweiten Beispiel sollen die in Abb. 3 gezeigten Module auf der Platine erkannt werden. Aufgrund der Senkrechtaufnahme erübrigt sich eine Entzerrung der Bilder. Da sich zwischen den Objektteilen stark texturierter Hintergrund befindet, wurde in diesem Fall eine manuelle Auswahl der Objektteile durch den Benutzer einer automatischen vorgezogen. Dadurch wurde vermieden, dass die Platine unnötigerweise als zusätzliches Objektteil extrahiert wurde. Die fünf vom Benutzer definierten ROIs sind in Abb. 12(a), die entsprechenden Kanten der Objektteile in Abb. 12(b) zu sehen. Die Relationen wurden mit Hilfe von 12 Beispielbildern trainiert. Das Ergebnis

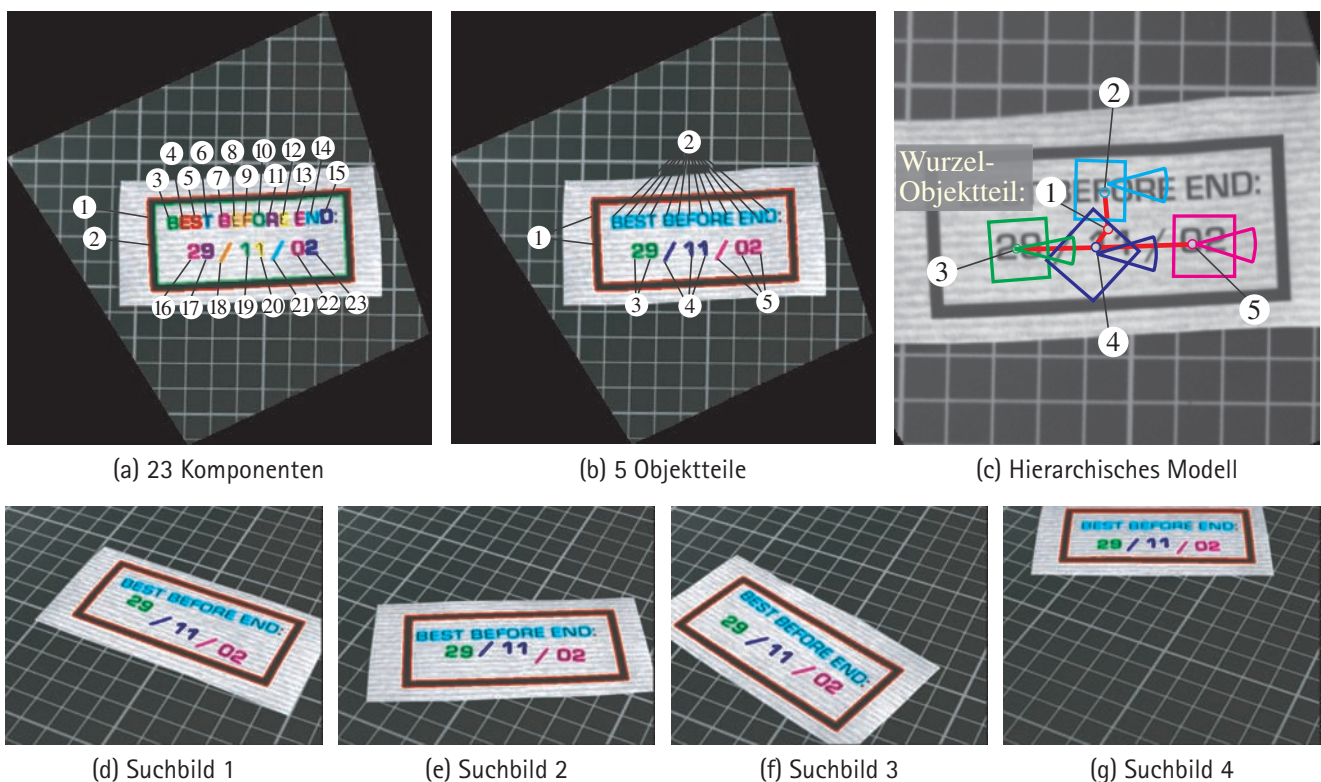


Abb. 11: Erkennung des Aufdrucks auf einem Etikett unter projektiven Verzerrungen. Der rechteckige Rahmen, der Schriftzug und die drei Ziffernpaare des Datums bewegen sich relativ zueinander.

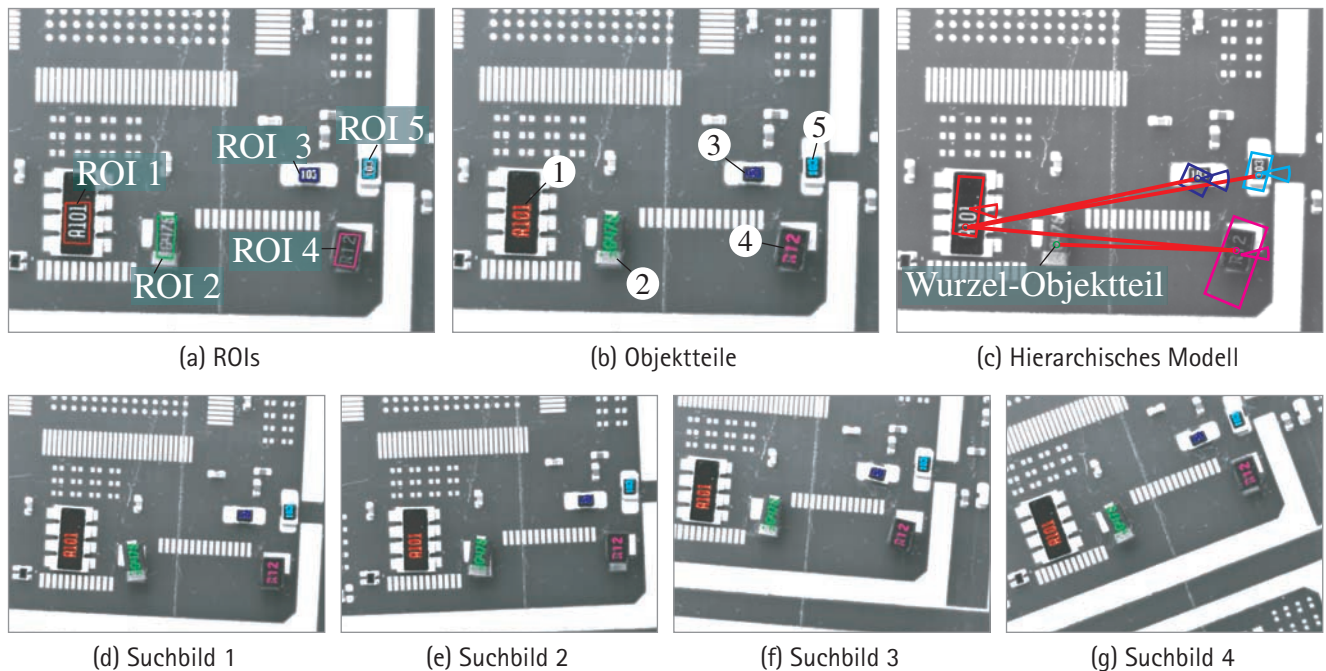


Abb. 12: Erkennung elektronischer Module auf einer Platine. Die relativen Positionen und Orientierungen der Module variieren leicht.

der Modellgenerierung ist in Abb. 12(c) zu sehen. Objektteil 2 wurde als Wurzel-Objektteil vorgeschlagen, von dem aus Objektteil 4 gesucht wird. Die Lage von Objektteil 4 wird dann genutzt, um den Suchraum von Objektteil 1 einzuschränken, von dem wiederum Objektteile 3 und 5 gesucht werden. Das Verfahren wurde an mehreren Suchbildern getestet (vgl. Abb. 12(d)–(g)). In der Online-Phase benötigt das Verfahren durchschnittlich 20 ms, um das Objekt zu finden. Davon können 14 ms dem Erkennen des Wurzel-Objektteiles und lediglich 6 ms dem Erkennen der übrigen Objektteile zugeschrieben werden. Im Vergleich dazu benötigt eine Erkennung ohne Verwendung des hierarchischen Modells 240 ms, was ebenfalls auf eine Beschleunigung um den Faktor 12 schließen lässt.

Beide Beispiele verdeutlichen sehr gut den hohen Geschwindigkeitsgewinn, der durch den Einsatz des vorgestellten Verfahrens zu erzielen ist.

## 9 Resümee

In diesem Artikel wurde ein neues Verfahren vorgestellt, mit dem zusammengesetzte 2D-Objekte in Bildern unter Echtzeit-Anforderungen erkannt werden können. Das dem Verfahren zu Grunde liegende Prinzip basiert auf der bestmöglichen Verringerung des Suchaufwandes und dient somit dem Ziel, die Berechnungszeit während der Erkennungsphase zu minimieren. Die Umsetzung dieses Zieles wird durch die Einschränkung der Suche entsprechend der relativen Bewegungen der Objektteile erreicht.

Dies führt zu der Verwendung eines hierarchischen Modells: Lediglich das Objektteil, das an der Spitze der Hierarchie steht, wird innerhalb des gesamten Suchraumes gesucht. Die verbleibenden Objektteile werden hingegen innerhalb eingeschränkter Suchräume relativ zueinander unter Verwendung eines rekursiven Verfahrens gesucht. Durch den Einsatz des hierarchischen Modells kann Vorwissen über die räumlichen Beziehungen, d. h. die relativen Bewegungen, zwischen den Objektteilen bereits in einer sehr frühen Phase der Erkennung genutzt werden. Dadurch wird die Rechenzeit entscheidend reduziert. Außerdem werden durch die eingeschränkten Suchräume für die einzelnen Objektteile Probleme, die durch auftretende Mehrdeutigkeiten hervorgerufen werden würden, vermieden. Eine komplizierte und rechenintensive Lösung der Mehrdeutigkeiten während der Erkennungsphase erübrigt sich somit. Das vorgestellte Verfahren ist nicht auf eine bestimmte Objektart beschränkt, sondern auf nahezu beliebige Objekte anwendbar. Außerdem zeichnet sich das Verfahren durch eine hohe Robustheit aus und ist in der Lage, selbst unter projektiven Verzerrungen Echtzeit zu gewährleisten, was einen Einsatz im industriellen Bereich ermöglicht. Darüber hinaus wird durch den hohen Automationsgrad ein großer Anwenderkreis angesprochen. Das Verfahren ist ein gutes Beispiel dafür, dass die moderne Photogrammetrie in Verbindung mit Methoden des Bildverstehens auch in von der Photogrammetrie bislang noch weitgehend unerschlossenen Bereichen wertvolle Beiträge liefert und sich dadurch neue Einsatzmöglichkeiten und lohnenswerte Zukunftsmärkte sichern kann.

## Literatur

- Belongie, S., Malik, J. und Puzicha, J.: Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4): S. 509–522, 2002.
- Felzenszwalb, P. F. und Huttenlocher, D. P.: Efficient matching of pictorial structures. *Computer Vision and Pattern Recognition*, Band 2: S. 66–73, 2000.
- Fischetti, M. und Toth, P.: An efficient algorithm for the min-sum arborescence problem on complete digraphs. *ORSA Journal of Computing*, 5(4): S. 426–434, 1993.
- Gavrila, D. M. und Philomin, V.: Real-time object detection for »smart« vehicles. 7th International Conference on Computer Vision, Band I: S. 87–93, 1999.
- Graham, R. L. und Hell, P.: On the history of the minimum spanning tree problem. *Annals of the History of Computing*, 7(1): S. 43–57, 1985.
- Grimson, W. E. L.: Recognition of object families using parameterized models. *International Conference on Computer Vision*: S. 93–101, 1987.
- Grimson, W. E. L.: On the recognition of parametrized 2-D objects. *International Journal of Computer Vision*, 2(4): S. 353–372, 1989.
- Hel-Or, Y. und Werman, M.: Constraint-fusion for interpretation of articulated objects. *Computer Vision and Pattern Recognition*: S. 39–45, 1994.
- Hel-Or, Y. und Werman, M.: Model based pose estimation of articulated and constraint objects. *Third European Conference on Computer Vision*: S. 199–204, 1994.
- Jähne, B.: *Digital Image Processing*. 5. Ausgabe, Springer-Verlag, 2002.
- Jain, A. K., Murty, M. N. und Flynn, P. J.: Data clustering: A review. *ACM Computing Surveys*, 31(3): S. 264–323, 1999.
- Jain, A. K., Zhong, Y. und Lakshmanan, S.: Object matching using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(3): S. 267–278, 1996.
- Koch, K. R.: *Parameterschätzung und Hypothesentests in linearen Modellen*. Dümmler, Bonn, 1987.
- Kratchounova, T., Krebs, B. und Korn, B.: Erkennung und Bestimmung der aktuellen Konstellation von Objekten mit Scharniergelenken. In: B. Jähne, P. Geißler, H. Haußbeck und F. Hering (eds.), *Mustererkennung, Informatik aktuell*, Springer-Verlag, Berlin: S. 502–509, 1996.
- Lanser, S., Zierl, C. und Beutlhauser, R.: Multibildkalibrierung einer CCD-Kamera. In: G. Sagerer, S. Posch und F. Kummert (eds.), *Mustererkennung, Informatik aktuell*, Springer-Verlag, Berlin: S. 481–491, 1995.
- Lee, K.-M. und Street, W. N.: Generalized Hough transforms with flexible templates. *International Conference on Artificial Intelligence*: S. 1133–1139, 2000.
- Lenz, R.: Linsenfehlerkorrigierte Eichung von Halbleiterkameras mit Standardobjektiven für hochgenaue 3D-Messungen in Echtzeit. In: E. Paulus (eds.), *Mustererkennung, Informatik Fachberichte 149*, Springer-Verlag, Berlin: S. 212–216, 1987.
- Li, W.-J. und Lee, T.: Object recognition and articulated object learning by accumulative Hopfield matching. *Pattern Recognition*, 35(9): S. 1933–1948, 2002.
- Lowe, D. G.: Fitting parametrized 3-D models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5): S. 441–450, 1991.
- Marquardt, D. W.: An algorithm for least-squares estimation of non-linear parameters. *SIAM Journal of Applied Mathematics*, 11(2): S. 431–441, 1963.
- Pilu, M. und Fisher, R. B.: Model-driven grouping and recognition of generic object parts from single images. *Journal of Robotics and Autonomous Systems*, 21: S. 107–122, 1997.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T. und Flannery, B. P.: *Numerical Recipes in C: The Art of Scientific Computing*. 2. Ausgabe, Cambridge University Press, Cambridge, 1992.
- Scaroff, S. und Liu, L.: Deformable shape detection and description via model-based region grouping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5): S. 475–489, 2001.
- Steger, C.: Similarity measures for occlusion, clutter, and illumination invariant object recognition. In: B. Radig und S. Florczyk (eds.), *Mustererkennung 2001*, Springer, München: S. 148–154, 2001.
- Steger, C.: Occlusion, clutter, and illumination invariant object recognition. *International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences*, Band XXXIV, 3A, Graz: S. 345–350, 2002.
- Suganthan, P. N.: Structural pattern recognition using genetic algorithms. *Pattern Recognition*, 35(9): S. 1883–1893, 2002.
- Ullman, S.: *The Interpretation of Visual Motion*. MIT Press, Cambridge, MA., 1979.
- Ulrich, M., Baumgartner, A. und Steger, C.: Automatic hierarchical object decomposition for object recognition. *International Archives of Photogrammetry and Remote Sensing*, Band XXXIV, 5: S. 99–104, 2002.
- Ulrich, M., Steger, C., Baumgartner, A. und Ebner, H.: Real-time object recognition in digital images for industrial applications. 5th Conference on Optical 3-D Measurement Techniques, Wien: S. 308–318, 2001.

## Anschrift der Autoren

Dr.-Ing. Markus Ulrich  
MVTec Software GmbH  
Neherstraße 1, 81675 München  
Tel.: 089 457695-0, Fax: 089 457695-55  
ulrich@mvtec.com

Dr. rer. nat. Carsten Steger  
MVTec Software GmbH  
Neherstraße 1, 81675 München  
Tel.: 089 457695-0, Fax: 089 457695-55  
steger@mvtec.com

Dr.-Ing. Albert Baumgartner  
Lehrstuhl für Photogrammetrie und Fernerkundung  
Technische Universität München  
Arcisstraße 21, 80333 München  
Tel.: 089 28922671, Fax: 089 2809573  
albert.baumgartner@bv.tum.de

Univ.-Prof. Dr.-Ing. Heinrich Ebner  
Lehrstuhl für Photogrammetrie und Fernerkundung  
Technische Universität München  
Arcisstraße 21, 80333 München  
Tel.: 089 28922671, Fax: 089 2809573  
heinrich.ebner@bv.tum.de