

Lernen struktureller Modelle für die Bildanalyse

Bei der Fakultät für
Bauingenieur- und Vermessungswesen
der Universität Stuttgart
zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.)
eingereichte Dissertation

Vorgelegt von
Dipl.-Ing. Monika Sester
aus Reichenbach bei Gengenbach

Stuttgart 1995

Deutsche Geodätische Kommission

Prüfungskommission:

Vorsitzender: Prof. Dr. Sc. Wolfgang Keller

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Dieter Fritsch

2. Prof. Dr.-Ing. Wolfgang Förstner

Tag der Einreichung: 18. Januar 1995

Tag der mündlichen Prüfung: 24. Februar 1995 ¹

¹Erschienen: Reihe C, Nr. 441, Deutsche Geodätische Kommission, München

Inhaltsverzeichnis

Summary	6
I Einleitung und Zielsetzung der Arbeit	7
1 Einführung	7
1.1 Menschliches Sehen	7
1.2 Maschinelles Sehen	8
1.3 Übersicht über die Arbeit	10
1.4 Problemstellung und Lösungsansatz	11
1.4.1 Ablauf der Bildinterpretation	11
1.4.2 Automatische Ableitung strukturierter Modelle	12
1.5 Einordnung der Arbeit	13
II Grundlagen	15
2 Wahrnehmung	15
2.1 Grundlegendes zur Wahrnehmung	15
2.2 Wahrnehmung und Erkenntnis	16
2.3 Theorien zur Wahrnehmung	17
2.4 Wahrnehmung und Wissen	18
2.5 Implikationen für das Maschinelle Sehen	20
3 Wissen, Modelle und Repräsentation	22
3.1 Vorüberlegungen und Definitionen	22
3.2 Klassifizierung von Objektmodellen	23
3.2.1 Spezifische Modelle	24
3.2.2 Generische oder allgemeine Modelle	25
3.2.3 Übergeordnete Modelle	25
3.3 Objektmerkmale	26
3.4 Prozedurale Modelle	26
3.5 Formalismen zur Wissensdarstellung	27
3.5.1 Nicht-Objektbezogene Repräsentationsansätze	27
3.5.2 Objektbezogene Repräsentationsansätze	30
3.5.3 Objektorientierte Darstellung	32
3.5.4 Verarbeitung von Wissen	32
3.5.5 Diskussion der Repräsentationsformen	33

4	Modellierung	34
4.1	Allgemeine Vorüberlegungen	34
4.2	Anforderungen an die Modellierung	35
4.3	Möglichkeiten der Modellgenerierung	36
5	Maschinelles Lernen	38
5.1	Grundprinzip Maschinelles Lernverfahren	38
5.2	Klassifizierung der Verfahren	40
5.2.1	Klassifizierung nach dem Grad der Unterstützung durch den Lehrer	41
5.2.2	Klassifizierung nach der Wissensrepräsentation	42
5.3	Anwendungsgebiete Maschinelles Lernverfahren	42
5.4	Strukturelle Lernverfahren	43
5.4.1	ID3	44
5.4.2	Relax	44
5.4.3	Winston's Arch	45
5.4.4	Conceptual Clustering	46
5.5	Neuronale bzw. konnektionistische Lernverfahren	48
5.5.1	Backpropagation Algorithmus	48
5.5.2	Assoziativspeicher	49
5.5.3	Beispiele für den Einsatz Neuronaler Netze in der Bildanalyse	50
5.6	Diskussion und allgemeine Kritik	51
6	Verwandte Arbeiten	53
6.1	Generische Modelle für die Objekterkennung	53
6.1.1	Generalisierung der geometrischen Beschreibungselemente	53
6.1.2	Nutzung nicht-geometrischer Merkmale	55
6.2	Beispiele für Lernverfahren im Bereich der Bildanalyse	55
III	Automatische Ableitung von Modellen aus Beispielen	57
7	Überblick über das realisierte Verfahren	57
8	Lernverfahren ID3 mit Erweiterungen	61
8.1	Strategie und Beispiel	61
8.2	Defizite und Erweiterungen	64
8.2.1	Erweiterung auf kontinuierliche Attribute	65
8.2.2	Bewertung der Verzweigungen des Baums	67
8.2.3	Erweiterung auf ungenaue Merkmale	68

9 Objektorientierte Darstellung und Verarbeitung	70
9.1 Basisobjekte, Objekteigenschaften und Methoden	71
9.1.1 Definition der Basisobjekte	71
9.1.2 Definition der Basisrelationen	72
9.1.3 Erweiterbarer Funktionen-Pool	73
9.1.4 Objektklassen	75
9.1.5 Wichtige Methoden flächenhafter Objekte	76
9.2 Ablauf des Interpretationszyklus	80
9.3 Lernverfahren	81
9.3.1 Erzeugung von Objektklassen, ihren diskriminierenden Objektfunktionen und Prototypen	81
9.3.2 Lernen von Relationen und Erzeugung neuer Methoden	83
9.4 Konfliktlösungen	84
9.5 Beispielhafter Programmablauf	84
9.6 Automatische Erzeugung (Synthese) von Programmen für die Objekterkennung	89
10 Beispiele für inkrementelle, interaktive Objektmodellierung	92
10.1 Feldmodelle zur Landnutzungsklassifizierung	92
10.2 Modelle für kartographische Objekte	95
10.3 Diskussion der Beispiele	102
11 Zusammenfassung und Diskussion	107
Literaturverzeichnis	112
Dank	119
Lebenslauf	120

Summary

Every intelligent system has to rely on knowledge in order to act (relatively) autonomous. The basic problem is to bring the knowledge into the system - also known as knowledge acquisition. This process is usually done manually, namely by gathering the relevant information through interviews with experts. In this manner the knowledge of a certain domain is structured and stored in a so-called knowledge base. A disadvantage of such knowledge based systems (or expert systems) is, that the system only acts basing on its stored knowledge - even slight deviations from the given facts lead to failures. Thus every possibility has to be thought of in advance and fixed into the system.

Also in image understanding a lot of information has to be known before an automatic process can start. An important part of it is the knowledge about the objects in the image. Thus all possible appearing objects in a scene have to be known in advance and described in a way, that they can be used by automatic processes. A more flexible method of knowledge acquisition is to imitate the human learning process by using techniques from Machine Learning. In this thesis the method "Learning from Examples" is used to derive descriptions (models) of objects for image interpretation.

The implemented approach is organized in an object-oriented way and consists of an integrated process of learning and interpretation. The learning component itself is the program ID3, which derives concepts from given examples. Concepts which can be learned are both object descriptions and also relations between objects - which are usually hard both to define and to acquire. Thus the system allows to gain complex object models. Starting from an uninterpreted scene, in the course of the process object models are continuously acquired and refined, until characteristic and discriminating descriptions for all the object in the scene are gained. In this way the process results in a complete scene description but, more important, also in a collection of abstract object models of the scene. An additional important feature of the procedure is the possibility to determine also complex object classes in an iterative manner.

The system is semi-automatic in the sense that control is guided both by the objects and a so-called teacher, who shows examples and counter-examples for a concept to be learned. The system is verified with examples from automatic map interpretation. The novelty of this approach lies in an integrated, iterative process of learning and interpretation.

Teil I

Einleitung und Zielsetzung der Arbeit

1 Einführung

1.1 Menschliches Sehen

Der Blick aus einem Flugzeug auf eine Landschaft, aber auch die Betrachtung eines Luftbildes eröffnen gleichermaßen neue Perspektiven der üblichen Betrachtungsweise unserer Umgebung. Trotz des ungewohnten Blickwinkels ist es uns allerdings unmittelbar möglich, die verschiedensten Dinge auf der Erdoberfläche zu erkennen. So lassen sich Vegetation und bebautes Gebiet unterscheiden, man kann Reihenhaussiedlungen und Einzelgehöfte ausmachen, die Feldstrukturen sind sichtbar und Verkehrswege durchziehen das Land. Meist erkennen wir all diese Objekte unmittelbar. Liegen allerdings Verdeckungen vor, oder zeigt sich ein Objekt in bislang unbekannter Form, so wird deutlich, daß wir über ein Arsenal an Strategien verfügen, um Dinge dennoch aus dem Zusammenhang heraus zu erkennen: selbst wenn auf einer baumbestandenen Allee Bäume die Straße verdecken, ist es möglich, dieses stückweise unsichtbare Gebilde als zusammenhängend zu sehen, es als Straße zu bezeichnen, und seinem Verlauf zu folgen. Ebenso können die Umrisse eines Hauses entdeckt werden, selbst wenn das Haus einen Schatten wirft oder teilweise durch Bäume verdeckt ist. Unser Wissen um die Objekte der realen Welt und ihre Abbildung in der Vogelperspektive ist hierfür die wesentliche Voraussetzung. Ebenso wichtig sind die Beziehungen zwischen den Objekten: die Tatsache, daß Häuser Zufahrtswege besitzen müssen, wird unbewußt dazu ausgenutzt, diese zu erkennen. Das Wissen, daß sich auf Hausdächern meist Kamine befinden, trägt dazu bei, ein unbekanntes Objekt auf dem Dach als Kamin zu erkennen. Unser Wissen hilft uns auch, nach bestimmten Objekten zu suchen. Soll beispielsweise eine Landebahn erkannt werden, so wenden wir einen 'Filter' an, der gezielt nach länglichen, asphaltierten (daher grau-schwarzen) Flächen sucht, die bestimmte Markierungen aufweisen. Eine weitere grundlegende Fähigkeit kommt dann zum Tragen, wenn unbekannte Objekte auftauchen: ihre Bedeutung kann entweder aus dem Zusammenhang erschlossen oder durch Lernen ermittelt werden, etwa durch Hinweise oder Erklärungen bzw. durch Nachschlagen in einem Buch.

Das Forschungsgebiet "Bildverstehen" beschäftigt sich damit, dieses eben beschriebene menschliche Verhalten einer Maschine zu übertragen, was mit dem englischen Begriff "Computer Vision" sehr deutlich zum Ausdruck kommt. Die visuelle Wahrnehmung funktioniert bei Menschen normalerweise problemlos. Wir tun es ständig und nehmen es meist sogar überhaupt nicht als bewußten Vorgang wahr. Dies macht es auch so schwierig, es als Problem zu erkennen und erschwert somit die Suche nach den zugrundeliegenden Prozessen.

Das Szenario der Betrachtung der Landschaft vom Flugzeug aus zeigt die Möglichkeiten menschlicher Wahrnehmung und verdeutlicht, welche Fähigkeiten gefragt sind, wenn es um die visuelle Wahrnehmung und die Interpretation von Bildern oder generell unserer Umgebung geht. Im Hinblick auf eine Automation dieser Prozesse wurde obige Betrachtung bewußt für einen Fall durchgeführt, bei dem einige Informationsquellen ausgeschlossen sind. Für die menschliche Wahrnehmung spielen Tast-, Geruchssinn und Gehör eine wesentliche Rolle. Diese können allerdings für die Bildinterpretation nicht genutzt werden, da diese Information in einem Bild nicht gegeben ist.

Sehen bezeichnet nach Boden [1989] den Prozeß, der Bilder der realen Welt in eine für den Betrachter günstigere Beschreibung überführt, bei dem nur noch die wichtige Information erhalten bleibt und unwichtige entfällt. Diese Trennung von wichtiger und unwichtiger Information wird von biologischen Informationssystemen "automatisch" durchgeführt. Wie sie grundlegend funktioniert ist bislang jedoch noch nicht klar. Beobachtbar ist jedoch, daß sie nicht nur auf der Ebene der Informationserfassung wirkt, sondern auch auf höheren Aggregationsstufen, wo es nicht mehr unmittelbar um die Datengewinnung geht, sondern wo verarbeitende Prozesse im Mittelpunkt stehen. Das Prinzip der selektiven Auswahl wirkt also überall: an die nächst höhere Stufe der Verarbeitung werden lediglich bestimmte Aggregationsformen weitergegeben. Diese wiederum wählen hieraus weiter aus und bündeln sie zu neuen Ergebnissen.

Die Unterteilung nach Wichtigem und Unwichtigem ist für den Wahrnehmungsprozeß demnach von grundlegender Bedeutung. Sie setzt jedoch voraus, daß **Modelle** (d.h. Vorstellungen, Ideen, Vorwissen) vom Gesehenen vorhanden sind. Nur dann ist es möglich, Dinge überhaupt als solche zu unterscheiden bzw. zu bewerten, was im jeweiligen Zusammenhang wichtig und was unwichtig ist. Diese elementare Voraussetzung für das Erkennen wird uns jedoch dann erst bewußt, wenn wir keine Modelle haben, um eine Situation zu verstehen. Wir sehen bzw. verstehen solange nichts, bis wir einen Interpretationshinweis haben. Erst dadurch entsteht eine Vorstellung - entsteht die interne Repräsentation. Beispielsweise zeigt ein Blick auf eine Wiese für einen Botaniker eine Fülle von einzelnen Pflanzen und Spezies, während ein Laie lediglich eine grüne Grasfläche ausmachen kann; eine Landkarte kann nur lesen, wer die zugehörige Legende kennt und zu interpretieren weiß.

Die Art der zugrundeliegenden Modelle ist für die Wahrnehmung von zentraler Bedeutung. Der menschliche Wahrnehmungsapparat ist darauf ausgelegt, Objekte zu erkennen, selbst wenn sie in den verschiedensten Erscheinungen auftreten. Erkennung ist also möglich, obwohl es sehr selten vorkommt, daß ein und dasselbe Objekt unter der gleichen "Aufnahmesituation" wiederbetrachtet wird: Beleuchtung, Perspektive, Orientierung, ja selbst Veränderungen im Objekt selbst (Alterung), haben kaum Einfluß. Die zugrundeliegenden Modelle zeichnen sich also durch eine große Flexibilität aus. Die Vorstellungen sind von sehr allgemeiner Art: ohne je alle möglichen Pferde, Häuser oder Autos gesehen zu haben, ist es möglich, diese Objekte als solche zu klassifizieren. Offensichtlich werden allgemeine Modelle verwendet, die sich dadurch auszeichnen, daß sie alle Eigenschaften von Objektklassen beschreiben. Diese Modelle bezeichnet man auch mit dem Adjektiv "generisch". Das Individualobjekt unterscheidet sich dann in den jeweiligen Ausprägungen der Eigenschaften (z.B. Autotyp oder -farbe).

Fundamental für die Wahrnehmung ist die Fähigkeit zu lernen. Mit dieser Fähigkeit ist es dem Menschen möglich, ständig aus seiner Umgebung Information aufzunehmen und zu verarbeiten. Lernen bedeutet, kognitive Fähigkeiten unter äußerem Einfluß zu verbessern [Boden 1989]. Oft wird Lernen nur mit explizitem - oft recht anstrengendem - Lernen in der Schule oder im Beruf in Zusammenhang gebracht. Ein Großteil des Lernens geschieht jedoch unbewußt. Der Mensch lernt Objektmodelle durch alltäglichen Umgang mit ihnen. Eine wichtige Eigenschaft des Lernens beim Menschen ist, daß es inkrementell erfolgen kann, d.h. daß das Wissen kontinuierlich in Interaktion mit der Umwelt erweitert werden kann. Die aktuelle Erscheinung eines Objektes enthält meist nur Teilaspekte. Durch zunehmende Vertrautheit erschließen sich immer neue Aspekte und vorhandene Modelle können zunehmend verfeinert oder ergänzt werden. Die entstehenden Modelle sind monoton, was bedeutet, daß das frühere, noch unvollständige Modell weiterhin Gültigkeit behält, lediglich weiter verfeinert wird. Diese Tatsache läßt sich dadurch verdeutlichen, wie man nach und nach mit einer fremden Stadt vertraut wird. Zunächst wird man sich einige markante Gebäude oder Hauptstraßen merken - die übrigen Gebiete der 'mentalen Karte' bleiben leer. Erst nach und nach werden sie gefüllt. Das ursprüngliche Modell bleibt allerdings weiterhin gültig, auch wenn später bessere Beschreibungen bekannt sind.

1.2 Maschinelles Sehen

Objekterkennung bedeutet, durch Nutzung visueller Information in Verbindung mit Hintergrundwissen Objekte als Individuen oder als einer Klasse zugehörig zu erkennen [Strat 1994]. Automatische Objekterkennungssysteme versuchen, das menschliche Sehen nachzubilden. Um diesen komplexen Vorgang auf eine Maschine zu übertragen, sind zwei elementare Komponenten erforderlich [Flynn & Jain 1991]:

Modellierung: Der Prozeß der Modellierung beschäftigt sich damit, welche Modelle für das Erkennen benötigt werden und wie diese gewonnen werden können. Er umfaßt daher das Problem der Repräsentation von Modellen, sowie das der Modellakquisition.

Modell - Nutzung: In einem Schritt der Zuordnung werden Modelle und wahrgenommene Information zusammengebracht. Somit ist Erkennen möglich. Die zugrundegelegten Modelle und die Zuordnungsstrategie stehen dabei in einem diametralen Verhältnis, was den Aufwand betrifft: einfache Modelle bedingen eine ausgefeilte Zuordnungsstrategie, die viel Vorwissen integriert, während sehr detaillierte Modelle einfachere Strategien erlauben. Diese Komponente ist dem Menschen meist nicht bewußt. Das Erkennen erfolgt üblicherweise so unmittelbar, daß die Prozesse der Modellauswahl und der Zuordnung nicht als solche registriert werden.

Vergegenwärtigt man sich den heutigen Stand der Technik im Bereich des Maschinellen Sehens, so zeichnet sich dieser vor allem dadurch aus, daß bislang dort Erfolge erzielt worden sind, wo sich die Modellierung vergleichsweise einfach gestalten läßt. Hier werden modellbasierte Verfahren eingesetzt [Haralick & Shapiro 1993]. Dies trifft für Aufnahmen in kontrollierter Umgebung zu, wo die Art der auftretenden Objekte bekannt ist und auch die Blickwinkel und Beleuchtungsverhältnisse sich gut beschreiben bzw. vorgeben und steuern lassen ("Blocks-Welt", Industrieanwendungen). Auch in Fällen, wo sehr viel Vorinformation über Position und Art der Objekte zur Verfügung steht, ist eine Objekterkennung bzw. Rekonstruktion möglich (vgl. z.B. [Sester & Förstner 1989], [Schickler 1992], [Schutte 1994]).

Meist lassen sich Objekte über ihre geometrischen Merkmale beschreiben. Diese Objektmodelle werden üblicherweise "erdacht" und fest einprogrammiert. Für geometrisch beschriebene Objekte lassen sich beispielsweise alle topologisch unterschiedlichen Projektionen in sog. Ansichts-Graphen abspeichern (vgl. z.B. [Eggert, Bowyer & Dyer 1992]). Der große Vorteil dieser Modelle besteht im nachfolgenden Matching-Schritt für die Objekterkennung: hierbei genügt eine 2D-2D-Zuordnung, welche von bedeutend einfacherer Komplexität ist, als der eigentlich nötige 2D-3D-Vergleich (vgl. hierzu Verfahren der relationalen Zuordnung [Vosselman 1992]). Neben den modellbasierten Ansätzen gibt es darüber hinaus praxisreife Verfahren, die durch Erkennen von Objektmerkmalen im Zusammenspiel mit einer effizienten Zuordnungsstrategie schließlich Information über das Objekt ableiten (vgl. z.B. Programme zur Ableitung von digitalen Geländemodellen [Krzystek 1991], oder der relativen Orientierung [Haala, Hahn & Schmidt 1993]). Diese Methoden nutzen meist elementare Objektmerkmale, wie z.B. markante Punkte. Die Hauptlast der Erkennungsaufgabe liegt dabei nicht in der Modellierung, sondern im Zuordnungsschritt.

Modelle werden meist für eine bestimmte Aufgabenstellung erstellt und sind dann in der Lage, diese zu lösen - solange alle Voraussetzungen gegeben sind. Ändert sich allerdings auch nur eine Vorbedingung in den realen Daten, so schlagen diese Verfahren fehl. Auch sind diese manuell erstellten Modelle normalerweise nicht skalierbar, d.h. auf andere Problemstellungen übertragbar.

Bei komplexeren Aufgabenstellungen - interessanterweise gerade denjenigen, die der Mensch meist ohne Schwierigkeiten bewältigt - stoßen die meisten der bisherigen Methoden an ihre Grenzen. Sobald es beispielsweise um die Erkennung von Objekten in natürlicher Umgebung geht, ist ein großes Quantum an Vorwissen erforderlich, welches die bisherigen Verfahren oft nur implizit nutzten. In natürlicher Umgebung kommen zum einen sehr verschiedene Objekttypen vor, zum anderen erscheinen diese Objekte jeweils in unterschiedlichen Formen. Es ist unmöglich, für alle Objekte, die in realen Situationen (und seien sie auch noch so eingeschränkt auf spezielle Ausschnitte des realen Lebens, wie z.B. Verkehrsszenen, Vorstadtlandschaft, etc.) auftreten können, spezifische Modelle bereitzustellen. Dies gilt besonders für die Analyse von Luftbildern. Selbst vermeintlich einfache Strukturen, wie Häuser oder Feldeinheiten kommen in einer solchen Fülle von Formen und Größen vor, die eine explizite Modellierung eines jeden Objekts unmöglich macht. Ein zusätzliches Problem ist dadurch gegeben, daß durch Veränderungen in der Umgebung auch ständig Veränderungen bzw. Erweiterungen dieser Modelle nötig sind.

Oft lassen sich natürliche Objekte auch nicht allein aufgrund ihrer Geometrie beschreiben (z.B. Baum, Himmel). Hier ist dann nach zusätzlichen Objektmerkmalen zu suchen. Auch die Erkennungsstrategie kann in ein Modell mit aufgenommen werden. Die Nutzung generischer Modelle unterstützt dies, da diese eine grob-zu-fein-Gliederung der Objekte angeben. Somit können Objekte aufgrund ihrer globalen Merkmale als zu einer Klasse zugehörig erkannt werden. Einzelobjekte lassen sich dann als Unterklasse einer allgemeineren Klasse einordnen. Damit wird erreicht, daß nicht für jedes neu hinzukommende Objekt bzw. für jedes Einzelobjekt einer Klasse ein spezifisches Modell gebildet werden muß.

Aus diesen Beobachtungen ergibt sich die Anforderung nach einer Flexibilisierung der Modellierung für ein Erkennungssystem von Objekten in natürlicher Umgebung.

Mit einer flexibleren Modellierung kann das starre Einprogrammieren von Modellen umgangen werden, was eine leichtere Erweiterbarkeit eines Systems mit sich bringt. Für die automatische Wissensakquisition werden in der Künstlichen Intelligenz (KI) im Bereich des Maschinellen Lernens (ML) Techniken erforscht. Lerntechniken können Wissen explizit machen, welches in Beispielen implizit gegeben ist. Auf diese Weise können verborgene Zusammenhänge aufgedeckt werden. Solche Techniken sind für den Wissenserwerb in der Bildanalyse ebenfalls von größter Bedeutung. Daher wird von verschiedenen Seiten die Verbindung von Computer Vision und Maschinellern Lernen angestrebt, was sich u.a. in einem Spezialheft der *IEEE Transactions on Pattern Analysis*

and *Machine Intelligence* [1994], aber auch in einem Themenschwerpunkt “Erkennen und Lernen” der DAGM 1994 [Kropatsch 1994] manifestiert.

Die angestrebte Flexibilisierung der Modellierung zielt damit zum einen auf eine Automatisierung der Modellgewinnung und zum anderen auf die Bereitstellung und den Einsatz allgemeiner Modelle. Die Lernfähigkeit ermöglicht eine Anpassung des Systems an eine sich ändernde Umgebung. Der Modellierungsprozeß kann automatisch angestoßen werden, wenn neue, bislang unbekannte, Objekte auftauchen.

1.3 Übersicht über die Arbeit

Die Idee der automatischen Wissensakquisition besteht somit darin, den menschlichen Wissenserwerb mit rechen-technischen Methoden zu simulieren. Das Fernziel ist dabei ein System, welches Objekte und Situationen, die es präsentiert bekommt, in eine für die spätere Nutzung günstige Form überführt, und somit in seinen Wissensvorrat mit aufnimmt.

Gegenstand der vorliegenden Arbeit ist die erste Komponente eines Objekterkennungssystems - die Modellierung. Dabei ist zu erarbeiten, welche Modellbeschreibungen günstig sind, und wie sie gewonnen werden können. Die Nutzung der Modelle zur Objekterkennung, die Zuordnung, steht dabei im Hintergrund. Sie soll allerdings für die Modellierung immer als das Ziel, auf das diese Prozesse ausgerichtet sind, im Hintergrund stehen, da die Modelle letztendlich natürlich dazu dienen sollen, Objekte zu erkennen. Das vorgestellte System ist allerdings derart auf die spätere Objekterkennung abgestimmt, daß die ermittelte strukturierte Objektbeschreibung direkt übertragen werden kann. Die Beschreibungsmerkmale sind solche, die auch von einem Bildverarbeitungssystem segmentiert werden können, und daher vergleichbar. In vorliegender Arbeit werden Modelle für zweidimensionale Objekte der Erdoberfläche aus symbolischem Bildmaterial abgeleitet.

Aufgrund dieser Problemstellung ergeben sich folgende Teilaspekte, denen anschließend jeweils ein Kapitel gewidmet wird:

- Zunächst soll das Ziel der Arbeit herausgestellt werden, um so die eingeschlagene Vorgehensweise zu motivieren. In einem einführenden Kapitel wird daher ein Überblick über den Ansatz der Arbeit sowie die zugrundeliegenden Voraussetzungen und Ziele gegeben.
- Es zeigt sich, daß die maschinellen Möglichkeiten der Objekterkennung noch weit hinter denen des Menschen liegen. Zur Motivation des Vorgehens dieser Arbeit, nämlich des inkrementellen, interaktiven Wissenserwerbs aus Beispielen, wird daher analysiert, welche Theorien zur menschlichen Wahrnehmung existieren. An dieser Stelle werden solche vorgestellt, die die Prozesse der menschlichen Wahrnehmung und Wissensrepräsentation zu erklären versuchen. Auch wenn keine der Theorien alle Phänomene erklären kann, lassen sich doch Prinzipien ableiten, die ihr Entsprechendes im maschinellen System finden können. Absicht ist, gemeinsame Vorstellungen herauszuarbeiten, um so mögliche Ideen für die Umsetzung auf ein automatisches System geben zu können.
- Das für die Wahrnehmung und das Lernen nötige Vorwissen wird in Form von Modellen in ein System gebracht. Daher wird der komplexe Begriff des Modells ausgehend vom intuitiven Verständnis erklärt und eingeordnet. Hierbei werden die verschiedenen Modelltypen herausgearbeitet, mit besonderem Fokus auf generische Modelle. Im Sinne eines Überblicks werden anschließend Verfahren zur Objektrepräsentation, d.h. Möglichkeiten der Speicherung von Modellen, vorgestellt.
- Daran schließen sich Methoden zur Gewinnung der Modelle an. Hier wird diskutiert, welche Möglichkeiten der Modellakquisition existieren und diese bewertet.
- Ein Kapitel über das Maschinelle Lernen soll die Problemstellung, die diesem Themengebiet zugrundeliegt, sowie die gängigen Verfahren erläutern.
- Anschließend wird das in der Arbeit entwickelte Konzept einer interaktiven Interpretations- und Lernstrategie vorgestellt. Dieses Werkzeug wird für die Wissensakquisition in der Bildanalyse genutzt. Anhand des Beispiels der Modellierung von kartographischen Objekten wird das vorgestellte Konzept erläutert.
- Diskussion und Ausblick beschließen die Arbeit.

1.4 Problemstellung und Lösungsansatz

Ziel der vorliegenden Arbeit ist die effiziente Bestimmung und Bereitstellung von Modellen für die automatische Bildanalyse. Es soll daher eine Art von Wissenserwerbskomponente entwickelt werden, welche Modelle aus vorgegebenen graphischen Vorlagen erlernt. Die gelernten Konzepte sollen möglichst allgemein sein, und für die Interpretation von Bildern genutzt werden können. Der Schwerpunkt der Arbeit liegt daher in der Übertragung der graphischen Repräsentation in eine interne Darstellung, sowie in der Verallgemeinerung dieser Beschreibungen zu generischen Objektmodellen mittels Lernverfahren. Durch die Ausrichtung an der Aufgabe des Bildverstehens ergeben sich Anforderungen an die Beschreibungssprache und an die erforderlichen Modelle.

1.4.1 Ablauf der Bildinterpretation

Die automatische Bildanalyse ist charakterisiert durch eine explizite Verwendung von Wissen über die im Bild dargestellte Szene. Dieses Wissen muß dem System a priori zur Verfügung stehen, um es für die Erkennung zu verwenden. Hierzu zählt strukturelles Wissen über das Objekt, aber auch kontextuelles Wissen über die Gesamtszene, u.a.:

- Wissen über die auftretenden Objekte und deren Struktur (deskriptive Eigenschaften),
- Wissen über Unterschiede zu anderen Objekten (diskriminierende Eigenschaften),
- Wissen über funktionelle Eigenschaften der Objekte,
- Wissen über Relationen zwischen den Objekten in der Szene (kontextuelles Wissen),
- Wissen über die Verfahren und Methoden der Objekterkennung,
- Wissen über die Verfahren und Methoden der Objektklassifizierung,
- Wissen über Erkennungsstrategien und Konfliktlösungsmöglichkeiten.

Im Prozeß der Bildinterpretation steht das zu interpretierende Bild auf einer Seite, das Modell auf der anderen. Objekte erkennen bedeutet, eine Zuordnung zwischen Bild und Modell zu finden. Üblicherweise geschieht dieser Vergleich nicht auf der Bildebene, sondern wird auf der Basis von extrahierten, symbolischen Bildmerkmalen durchgeführt. Prinzipiell läßt sich die Zuordnung auf verschiedenen Repräsentationsstufen durchführen - vorausgesetzt die Repräsentationen entsprechen sich (vgl. Abbildung 1.1). Jeder Repräsentationsformalismus bedarf einer Methode der Zuordnung. Die Verfahren reichen von einfachen mathematischen Transformationen bis zu komplexen Suchverfahren.

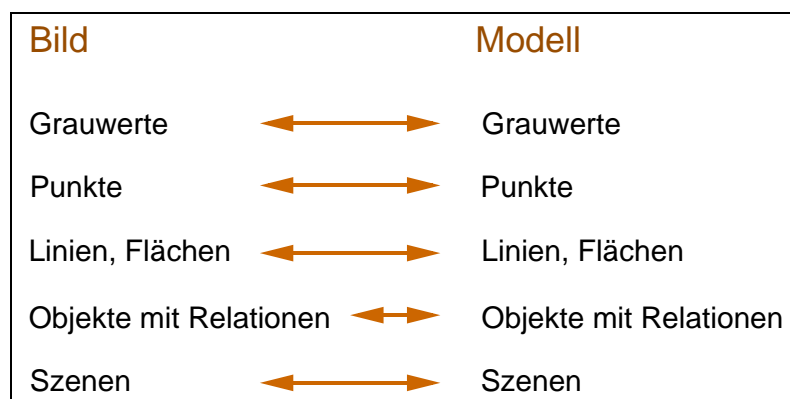


Abbildung 1.1: Zuordnung von Bild- und Modell auf verschiedenen Repräsentationsstufen

Die übliche Vorgehensweise in der Bildanalyse besteht darin, die Bildinformation zunächst auf einige wesentliche Merkmale zu reduzieren. Hierzu dienen low-level Bildverarbeitungsrouitinen, z.B. kantenorientierte oder regionenbasierte Operationen. Die so gewonnene Information zeichnet sich dadurch aus, daß sie potentiell semantisch wichtige Objektinformation enthält: während ein einzelnes Pixel noch jede beliebige Bedeutung haben kann, steht eine extrahierte Kante hingegen für eine Objekt-, Schatten- oder Texturgrenze [Schenk 1992].

An die Extraktion dieser Merkmale schließt sich der Zuordnungsprozeß an, der aus diesen die interessierenden Objekte ableitet. Lokal betrachtet kann prinzipiell jedes extrahierte Merkmal zum gesuchten Objekt gehören. Erst eine globalere Betrachtung kann die Frage der Objektzugehörigkeit klären. Dazu werden üblicherweise Suchverfahren verwendet, welche alle möglichen Kombinationen der Merkmale durchspielen. Dabei steigt die Anzahl der Möglichkeiten exponentiell mit der Anzahl der Merkmale an. Um diese kombinatorische Komplexität handhaben zu können, müssen geeignete Restriktionen eingeführt werden. Das Problem der Zuordnung als Isomorphismusproblem ist NP-vollständig. Eine rein datengetriebene (bottom-up) Vorgehensweise ist auch als biologisches Modell nicht tragbar. Tsotsos [1987] zeigt, daß durch massive Parallelität allein menschliche Leistungen nicht erklärt werden können, da die neuronalen Leitungen hierfür zu langsam sind. Bildinterpretation ist somit nur unter Hinzunahme von Zusatzinformation möglich. Diese kann einerseits durch die Aufgabenstellung bereits gegeben sein, indem nur bestimmte Merkmale überhaupt in Betracht gezogen werden, zum anderen dadurch, daß die Anzahl der zu vergleichenden Merkmale reduziert wird. Durch Gruppierungsverfahren können elementare Merkmale aufgrund bestimmter Vorgaben zu neuen zusammengefaßt werden. Die neuen Merkmale stehen auf einer höheren Ebene der Repräsentation und haben eine reichere Bedeutung. Unzulässige Zuordnungen und Kombinationen können somit leichter und vor allem früher ausgeschaltet werden.

Welche Merkmale allerdings gruppiert werden dürfen, ist in der Regel vom gesuchten Objekt abhängig. In einer strukturierten Objektbeschreibung sind diese Merkmale abgelegt. Auf diese Weise kann eine Objekterkennung erfolgen, wenn ein Teil des Objekts identifiziert worden ist. Der Kontext gibt Querverweise zwischen Objekten an. In die Objektbeschreibung können sinnvollerweise auch besondere oder auffällige Merkmale integriert werden. Nach diesen Merkmalen kann dann gezielt gesucht werden, um somit - aufgrund ihrer Seltenheit - den Suchaufwand zu reduzieren.

Ein möglichst objektunabhängiges Bildinterpretationssystem zeichnet sich dadurch aus, daß es für viele verschiedene Aufgabengebiete einsetzbar ist. Aus diesem Grund verbietet sich der allzu frühe Einsatz spezifischen Objektwissens. Der Schwerpunkt heutiger Arbeiten geht daher dahin, daß versucht wird, die Segmentierung auf ein möglichst hohes Niveau zu heben, *ohne* bereits Modellwissen einzusetzen. Grundlage des sogenannten Perceptual Grouping ist, Gestaltgesetze zu nutzen und somit eine Trennung von "Figur und Grund" durchzuführen, ohne angeben zu müssen, um welches Objekt es sich dabei handelt ([Lowe 1985], [Heitger & von der Heydt 1993], [Bowyer 1993]). Die somit erzielte abstrakte Bildinformation kann leichter mit Modellen verglichen werden. Diese Merkmale umfassen also nicht lediglich Kantenstücke oder Regionen, sondern höherwertigere Information wie gerade Kanten, Parallelen, Kreise, Rechtecke. Weitere Ansätze versuchen, alles Objekt- und Strategiewissen in einem Modell zu vereinen und von einer problemunabhängigen Matchingstrategie aufzulösen (vgl. z.B. [Brooks 1983], [Draper, Collins, Brolio, Hanson & Riseman 1989], [Stilla & Jurkiewicz 1991]).

Die Bereitstellung des für die Bildanalyse nötigen Wissens erfolgt bislang derart, daß es explizit einprogrammiert wird. Dies ist sehr unflexibel und kann somit nicht allen auftretenden Situationen gerecht werden. Daher wird in dieser Arbeit der Einsatz von Lernverfahren für die Modellierung diskutiert.

1.4.2 Automatische Ableitung strukturierter Modelle

Der Problembereich, für den der automatische Wissenserwerb demonstriert werden soll, ist die Interpretation von Luftbildern. Die Anwendung beschränkt sich auf 2D-Objekte, in vorliegendem Falle sind es kartographische Objekte wie Häuser, Straßen oder Felder. Die graphischen Bildbeispiele werden in einer Form vorgegeben, die sich an segmentierten Bildern orientiert. Die Sammlung von Linienstücken könnte beispielsweise das Ergebnis eines Kantenoperators sein. Abbildung 1.2 gibt ein Beispiel für das verwendete Ausgangsmaterial (Haus- und Straßenbegrenzungen), welches hier einem realen Bild überlagert ist.

Dieses Bild besteht aus einer Sammlung von Punkten und Linien. In einem ersten Schritt wird es in eine interne Repräsentation überführt, indem die sogenannten Basisobjekte des Systems erkannt werden. Alle Basisobjekte sind durch eine Reihe von Attributen charakterisiert (z.B. Größe, Form, . . .); auch die Basisrelationen zwischen den Objekten werden berechnet. Die neue Repräsentationsstufe enthält somit alle - für das System erkennbaren - Bildobjekte, deren Charakteristika sowie ihre Relationen untereinander. Eine Generalisierung ergibt sich, wenn die Beschreibungen individueller Objekte verallgemeinert werden können. Dies kann beispielsweise durch eine globalere Beschreibung der Objektmerkmale (etwa durch Mittelung) oder durch die Aggregation von Objekten zu neuen Objekten erfolgen. Die Information, welche Objekte sich zusammenfassen lassen, muß nicht vorgegeben werden, sondern kann direkt aus den Daten mit Hilfe von Lernverfahren ermittelt werden. Damit



Abbildung 1.2: Bild mit überlagerten Haus- und Straßengrenzen - Eingabe für das System sind diese überlagerten Daten

läßt sich eine strukturierte Objektbeschreibung gewinnen. Dieses Objektmodell umfaßt somit eine Klassenbeschreibung der auftretenden Objekte, welche deren Eigenschaften, Relationen und Methoden umfaßt. Zudem wird automatisch eine spezifische Beschreibung der individuellen Objekte der Szene abgeleitet. Solch eine generische Objektbeschreibung läßt sich somit auch als Multi-Repräsentations-Ansatz verstehen. Die verschiedenen Ebenen stehen dann für unterschiedliche Sichten auf das Objekt: für die Detektion genügt die grobe Ebene der Objektklassenbeschreibung, für die Lokalisierung und Rekonstruktion individueller Objekte wird die spezifische Beschreibung genutzt, welche sich auf einer feineren Ebene befindet. Diese Beschreibung bildet damit das symbolische Äquivalent einer Bildpyramide (vgl. z.B. [Bobick & Bolles 1989]).

Das abgeleitete Modell beinhaltet objektspezifisches Wissen, welches die Strategie zu dessen Erkennung bereits impliziert. Die elementaren Bestandteile des Modells sind Objekte, die auch in Bildern extrahierbar sind. Auf diese Weise kann das Modell für die Bildererkennung direkt in einem Matching-Verfahren genutzt werden.

Dem vorgestellten System liegen also mehrere Repräsentationsebenen zugrunde: zunächst eine graphische Darstellung der Beispiele. Diese wird in eine interne Darstellung der gefundenen Basisobjekte überführt, welche schließlich in eine Darstellung der gelernten (generalisierten) Objekte mündet. Abbildung 1.3 gibt eine Übersicht über die in der Arbeit verwendeten Repräsentationen.

1.5 Einordnung der Arbeit

Die Grundproblematik dieser Arbeit ist der automatische Wissenserwerb. In vielen Aufgabenstellungen, besonders in den Ingenieurwissenschaften, kommen zunehmend intelligente bzw. autonome Systeme zum Einsatz - sei es für die automatische Werkstückkontrolle bzw. -vermessung oder für die Fahrzeugsteuerung (u.a.). Systeme, die autonom arbeiten, benötigen hierfür Wissen. Dieses sollte explizit sein, besonders wenn die Systeme nicht nur in kontrollierter Umgebung arbeiten, sondern wechselnden Einflüssen einer natürlichen Außenwelt ausgesetzt sind. Dieses Wissen manuell ins System zu bringen, ist für komplexe Aufgaben kaum mehr möglich. Hier bieten sich Verfahren zum automatischen Wissenserwerb an, die ein flexibles Reagieren auf neue Gegebenheiten prinzipiell ermöglichen.

Wie und unter welchen Voraussetzungen der Wissenserwerb automatisch ablaufen kann, wird in vorliegender Arbeit exemplarisch am Beispiel der Bildanalyse bzw. Karteninterpretation aufgezeigt. Der Vorteil des Ansatzes der Arbeit beruht auf der engen Verknüpfung von Lern- und Erkennungsverfahren - hierin unterscheidet er sich auch von anderen Verfahren. Die Grundlage für dieses Zusammenspiel zwischen Lernen und Erkennen liegt insbesondere in der Repräsentation der gewonnenen Modelle, welche in objektorientierter Form gegeben ist. Sie ist für beide Aufgabenstellungen identisch - daher ist der Weg zur direkten Nutzung der abgeleiteten Modelle

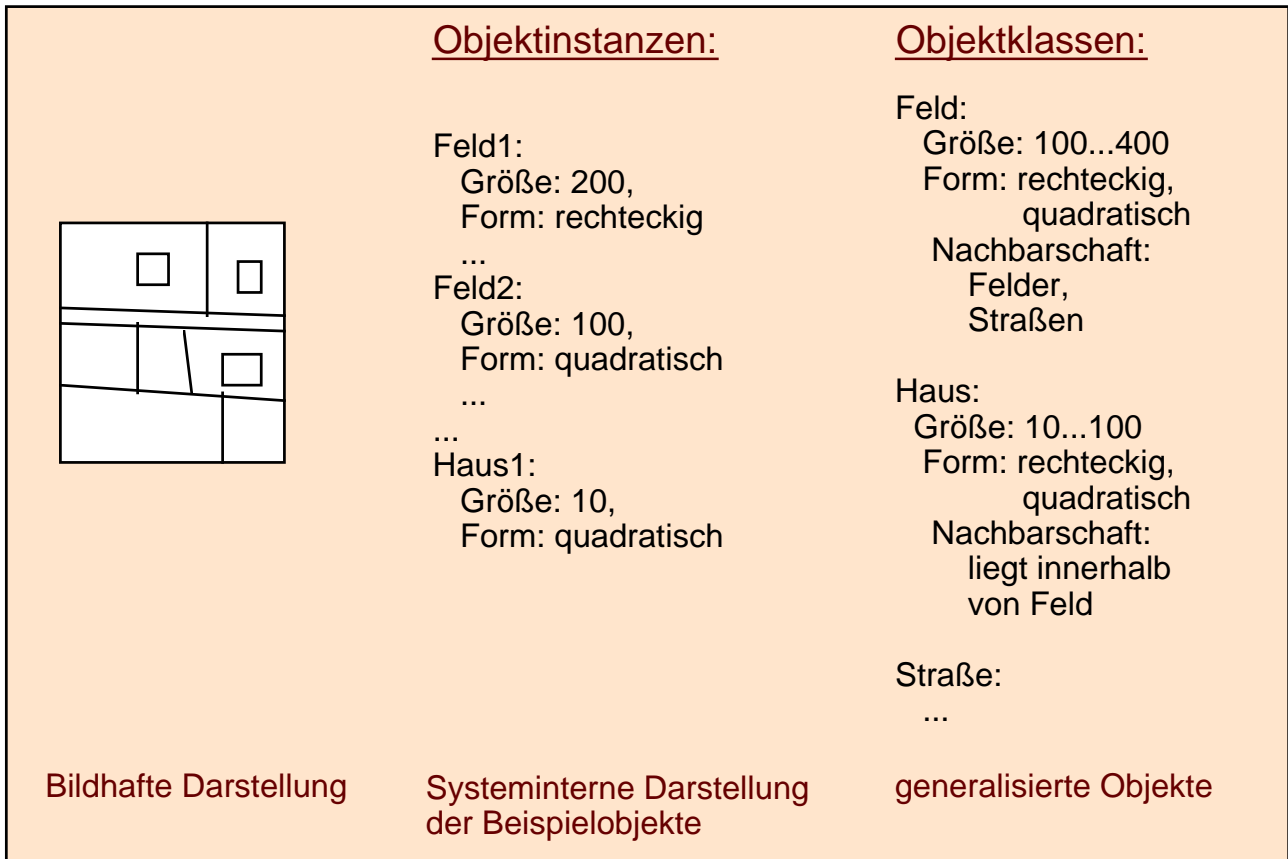


Abbildung 1.3: Übersicht über die verwendeten Darstellungsformen

im Bildanalyseprozeß offen. Der wesentliche Beitrag dieser Arbeit liegt darin, Information aus der Außenwelt aufzunehmen und diese durch Lernprozesse in neues Wissen zu überführen, welches ab dann genutzt werden kann. Damit ist das zur Verfügung stehende Vokabular prinzipiell sukzessive erweiterbar.

Teil II

Grundlagen

2 Wahrnehmung

2.1 Grundlegendes zur Wahrnehmung

Wir nehmen die Umgebung durch unsere Sinne wahr. Dabei kommt dem visuellen System eine besondere Rolle zu. Sobald wir die Augen öffnen, sehen wir Dinge und ordnen sie meist unmittelbar bestimmten Objekten zu. Wir erkennen Gegenstände, selbst wenn diese in einem anderen Licht oder in einem anderen Zusammenhang erscheinen, ja wir können sie sogar zuordnen, wenn wir sie noch nie gesehen haben und lediglich Ähnlichkeiten mit Bekanntem feststellen können. Die Wahrnehmung hilft nicht nur, Dinge unserer Umgebung zu erkennen und zu benennen, sondern sie steuert auch unser gesamtes Verhalten: Dinge können als Hindernis, als etwas Greifbares oder als essbar erkannt und entsprechend mit gesonderten Verhaltensweisen weiterbehandelt werden. Die Information, die wir permanent aus der Umgebung aufnehmen, in sinnvolle Einheiten, d.h. Objekte zu gliedern und dann somit zu erkennen, geschieht bereits auf einer Ebene, die unserem Bewußtsein nicht zugänglich ist. Daher ist es schwierig, die Objektwahrnehmung überhaupt als Problem zu erkennen, was die Suche nach den zugrundeliegenden Prozessen erschwert.

Prinzipiell läßt sich das Problem in die Wahrnehmung eines Objektes 'als Gestalt' und in sein Erkennen 'als bestimmtes bekanntes Objekt' untergliedern. Ersteres betrifft die Wahrnehmung an sich, das Problem der Trennung von 'Figur und Grund' (d.h. Objekt und Hintergrund), während sich daran die eigentliche Objekterkennung anschließt. Daß die Figur-Grund-Trennung einen eigenständigen Prozeß darstellt, wird dann deutlich, wenn sich bei der Betrachtung eines Bildes eine Organisation nicht unmittelbar einstellt. Erst nach längerem Hinsehen wird das Fleckenmuster aus Abbildung 2.1 neu gruppiert und man sieht Objekte mit Tiefe und Form². Dieses Phänomen machen sich Tiere auch für ihre Tarnung zunutze.



Abbildung 2.1: Figur-Grund-Trennung: Erkennung ist erst möglich, nachdem eine Gruppierung erfolgt ist (aus [Marr 1982])

Figur-Grund-Trennung und Objekterkennung sind jedoch sehr eng miteinander verzahnt: die Suche nach bestimmten Objekten wird gleichsam direkt mit einem entsprechenden Filter für dessen spezielle Merkmale durchgeführt. Daher ist die Objektrepräsentation für die Wahrnehmung von möglichen Objektkandidaten wichtig.

Die Definition eines Objekts erweist sich als problematisch. Sie ist von der jeweiligen Aufgabenstellung abhängig und auch vom wahrnehmenden Subjekt und dessen Erwartungen. Ein Blick von der Zuschauertribüne eines Pferderennens kann sich beispielsweise auf das "Objekt" Pulk, auf den Sieger, auf das Pferd oder dessen Reiter konzentrieren; oder auch auf Details des Reiters: seine Haare, seine Augenfarbe. Uexküll ([Lorenz 1977]) definiert in diesem Zusammenhang ein Objekt als "etwas, was sich zusammen bewegt".

²Auflösung siehe Fußnote auf Seite 26



Abbildung 2.2: Ames Zimmer: Dem Prinzip der Größenkonstanz wird - aufgrund der speziellen Konstruktion des Zimmers - scheinbar widersprochen (aus [Bruce & Green 1991])

Auf der Suche nach objektiv beobachtbaren Wahrnehmungsphänomenen lassen sich insbesondere folgende ausmachen:

Figur-Grund-Trennung: Hiermit ist die Trennung von Objekt und Hintergrund gemeint. Die Festlegung von Objekt und Hintergrund ist allerdings von der Betrachtungsweise abhängig und damit austauschbar.

Größen-, Formkonstanz: Größe und Form von Objekten einer Szene, die sich im Raum bewegen, werden als konstant gesehen (vgl. Abbildung 2.2).

Farb-, Helligkeitskonstanz: Objekte, die im Schatten liegen, behalten die Farbe.

Positionskonstanz: Szenen werden stabil wahrgenommen: Bei Bewegungen des Kopfes bewegt sich die Szene nicht.

2.2 Wahrnehmung und Erkenntnis

Die Wahrnehmung bewältigt das Problem, aus der Flut von Information, die unsere Umgebung bietet, einzelne Dinge zu erkennen. Psychologische Forschungen stimmen darin überein, daß wir unsere Umgebung erst dadurch wahrnehmen können, daß wir die gesehenen Bilder einer Art Interpretationsprozeß unterwerfen. Erst durch das Filter unseres Wissens ("die Brille unserer Wahrnehmung") wird Information zu Wissen. Was wir wahrnehmen und wie wir das tun, ist bestimmt durch die Struktur unseres Erkennungsvermögens. Dieses wirkt sich unmittelbar auf unsere Beobachtungen, Vorstellungen und Erfahrungen aus und prägt diese. Ohne Erkenntnisstrukturen ist keine Wahrnehmung möglich. Alle Organismen verfügen über sie (wenn auch in unterschiedlichen Ausprägungen); sie sind Basisvoraussetzung für das Überleben.

Kant³ bezeichnet dieses Grundwissen, mit dem alle unsere Sinneswahrnehmungen verarbeitet werden, mit dem Begriff "Schema". Nur hiermit kann Wahrgenommenes in Wissen umgewandelt werden. Erkennen ist durch diese Denkformen und strukturierten Denkabläufe determiniert. Man erkennt nie die Wirklichkeit an sich, alles abgespeicherte Wissen ist bereits durch diese "mentale Brille", die wir von Geburt an haben, gefiltert. Kant geht davon aus, daß dieses Schemawissen uns a priori gegeben ist.

Dieser Vorstellung des Apriorischen widerspricht die durch Lorenz begründete Evolutionäre Erkenntnistheorie [Lorenz 1977]. Lorenz stimmt zwar mit Kant überein, daß ohne solche Schemata keine Erfahrung und kein Erkennen möglich ist. Im Sinne Darwins wird allerdings angenommen, daß unser Wahrnehmungsapparat im Laufe

³zitiert nach [Lorenz 1977]

der Entwicklungsgeschichte des Lebens entstanden ist. In ständiger Auseinandersetzung mit der Wirklichkeit haben die Organismen Methoden entwickelt, die Dinge wahrzunehmen, die für ihr Fortleben relevant sind. Dies erklärt die Eigenschaften aber auch die Begrenzungen unserer Erkenntnisfähigkeit. Dinge, mit denen wir ständig zu tun haben, können wir ohne weiteres erkennen. Was 'unseren Horizont' überschreitet, ist zwar durch gedankliche Anstrengung erfahrbar, aber nicht unmittelbar vorstellbar (unsere Vorstellung scheitert z.B. wenn es um 4 oder höherdimensionale Räume geht). Diese ständige Anpassung und Änderung läßt sich als Regelkreis vorstellen, der bewirkt, daß Auswirkungen von Maßnahmen immer sofort wieder in das System zurückgeführt werden, um sie so zu prüfen und zu werten.

Offensichtlich hatten Fähigkeiten, die der Mensch 'einfach so' - ohne nachzudenken - beherrscht, in der Evolutionsgeschichte höhere Priorität, als solche, die gemeinhin als Intelligenzleistungen angesehen werden, z.B. Rechenaufgaben lösen [Campbell 1989]. Dieses unbewußte Wissen, was unsere gesamte Wahrnehmung und Wissensspeicherung steuert, ist tief im Unterbewußten abgelegt - was sich darin zeigt, daß wir es meist gar nicht als Leistung betrachten, da es uns so vertraut und geläufig ist.

Auch in der individuellen Entwicklung der Lebewesen läßt sich ein solcher Regelkreis der ständigen Adaption beobachten. Empirische Untersuchungen zeigen, daß einige perzeptuelle Fähigkeiten bereits von Geburt an vorhanden sind, viele jedoch erst später hinzukommen. Sie verbessern sich von Geburt an bis etwa zum Alter von sechs Jahren. Versuche deuten auf angeborene Strukturen hin. Diese bedürfen allerdings äußerer Stimulation - die oft sogar in bestimmten Lebensabschnitten erfolgen muß, um ihre eigentliche Funktion zu erfüllen. Ihre Ausprägung erfolgt durch Interaktion mit der Umwelt. Es ist gleichsam ein biegsames Gerüst vorhanden, das sich entsprechend den äußeren und inneren Gegebenheiten anpassen kann. Somit läßt sich auch erklären, daß bei Ausfall oder aufgrund von Fehlen bestimmter Organe andere deren Aufgaben (teilweise) übernehmen können. Offensichtlich können solche Leistungen nicht starr angelegt werden, da sie von der Entwicklung anderer Körperteile und Organe abhängig sind: das Stereosehen ist beispielsweise vom Augenabstand abhängig.

2.3 Theorien zur Wahrnehmung

Das Phänomen Wahrnehmung läßt sich an Menschen und anderen Lebewesen beobachten. Welche Vorgänge ihr allerdings zugrundeliegen, ist nicht klar und wird unterschiedlich erklärt. Es lassen sich drei klassische Theorien unterscheiden (vgl. z.B. [Rock 1986], [Uhr 1973], [Bruce & Green 1991], [Anderson 1989], [Wilding 1982]). Keine kann allerdings alle Phänomene erklären.

Die **deduktive oder empirische Betrachtungsweise** geht von elementaren Sinneseindrücken aus, die zu einem Gesamteindruck führen, indem sie mit bestehenden Ideen oder Vorstellungen verknüpft werden. Ausgehend von einer *tabula rasa* des Geistes werden diese Ideen im Laufe des Lebens durch Erfahrung erworben und erweitert. Jede Wahrnehmung basiert daher auf einzelnen Sinneseindrücken und ev. weiterer Information, die über Schlußfolgerungen in Zusammenhang gebracht werden müssen. Diese Schlußfolgerungen basieren auf Erfahrung und laufen unbewußt ab.

In starkem Gegensatz hierzu behaupten die Vertreter der **Reiztheorie**, daß alle Information direkt aus der Außenwelt aufgenommen und somit wahrgenommen wird, ohne komplizierter Schlußfolgerungen zu bedürfen. Ihrer Ansicht nach existiert für jede Art von Wahrnehmung ein spezifischer Reiz, der unmittelbar zu einem Eindruck führt, ohne weiterer Interpretation zu bedürfen. Alle Objekte sind durch komplexe Reize bereits voll spezifiziert. Dies führte u.a. zu der polemischen Feststellung, daß es somit auch eine "Großmutterzelle" geben müsse (vgl. z.B. [Franck 1991]). Allerdings lassen sich elementare Reizstrukturen beim Frosch z.B. nachweisen. Er verfügt über eine Sammlung von Sinneszellen, die unmittelbar auf dunkle, sich bewegende Konturen reagieren ("bug detector" [Boden 1989]). Gibson [1950] (zitiert nach [Bruce & Green 1991], [Marr 1982]) erweiterte diese Theorie, indem er komplexe, abstrakte Reize postulierte, z.B. die Veränderungen von Strukturen - nicht nur die eigentlichen Strukturen. So ist beispielsweise die Änderung der Texturdichte als Reiz ein unmittelbarer Hinweis auf räumliche Tiefe. In der Reiztheorie hängt die Wahrnehmung mehr vom Objekt ab als vom wahrnehmenden Organismus. Da Gibson bereits von 'reicher' Objektinformation ausgeht, benötigt seine Theorie keine Inferenzen mehr, die Wahrnehmung erfolgt unmittelbar.

Zwischen diesen beiden Extrempositionen läßt sich die Gestalttheorie [Wertheimer 1958] ansiedeln. Sie propagiert, daß Wahrnehmung immer bereits auf der Basis von 'Gestalten', von ganzen Einheiten, abläuft. Dabei geht man von angeborenen Vorstellungen wie Form, Größe, etc. aus, welche die Wahrnehmung organisieren. Diese

Organisation der nicht-zusammenhängenden Sinneseindrücke geschieht spontan und muß nicht über komplexe Systeme verrechnet werden. Wichtig in diesem Zusammenhang ist die Feststellung, daß das Ganze immer mehr ist als die Summe seiner Teile. Dieser Prozeß basiert auf den sog. Prägnanz-Gesetzen, die dadurch gekennzeichnet sind, daß eine Organisation in Richtung "gute Gestalt" erfolgt. Diese Gesetze sind Gleichgewichtsgesetze, zu ihnen gehören u.a. das Gesetz der Nähe, der guten Form, der Ähnlichkeit, der Umschlossenheit, der guten Fortsetzung [Hilgard & Bower 1973] (siehe auch Abbildung 3.3 Seite 25). Das Phänomen der Gestaltwahrnehmung wird mit Feldkräften erklärt, die vergleichbar der magnetischen Anziehungskraft arbeiten. Diese Kräfte wirken dabei nach dem Prinzip der Einfachheit: im Bild vorhandene Linien werden so zusammengefaßt, daß sie die einfachste Gestalt bilden. Um 1940 kam man mit der Informationstheorie [Shannon & Weaver 1949] zu einem Maß für Einfachheit. Einfachheit wurde hier mit dem Grad der Überraschung des Auftretens - und damit mit seiner Wahrscheinlichkeit - in Verbindung gebracht.

Die Theorien unterscheiden sich somit grundlegend in ihren Objektmodellen und der damit verbundenen Erkennungsstrategie: Für die Empiristen bestehen die Objekte lediglich aus ihren elementaren Merkmalen und nur durch komplexe Verrechnungen können sie erkannt werden. Die Gestaltisten gehen von höherwertigen Merkmalen aus, die unmittelbar erkannt werden. Diese stehen allerdings noch nicht für Objekte, sondern bewirken lediglich eine Trennung in bedeutungsvolle und unbedeutende Regionen. Die eigentliche Objekterkennung benötigt objektspezifisches Wissen, welches auch über Schlußfolgerungen eingebracht werden muß. Für Gibson hingegen sind die Reize, die von einem Objekt ausgehen, direkt und unmittelbar ein Hinweis auf des Objekt selber, so daß die Objekterkennung ohne Schlußfolgerungen auskommt und somit lediglich einem Aufsuchen eines bestimmten Reizes im Objektvorrat entspricht.

Lernen für die Wahrnehmung wird von den drei Theorien unterschiedlich bewertet. Für Gibson ist kein Lernen erforderlich. Die Gestaltisten propagieren, daß kein Lernen nötig ist, um elementare Strukturen, auf denen Wahrnehmung basiert, zu erfassen. Die Objektbeschreibungen allerdings erfordern Lernverfahren. Für die Empiristen ist Lernen und Erfahrung elementar wichtig für die Schlußfolgerungen und damit Basis für die Wahrnehmung.

Keine der Theorien vermag allerdings alle Wahrnehmungsphänomene zu erklären. Es kann jedoch allgemein gesagt werden, daß die Wahrnehmung von Objekten zunächst auf automatischen Prozessen basiert, danach durch Hypothesen über Objekte und ihre relevanten Merkmale gesteuert wird. Die Theorien zur Wahrnehmung unterscheiden sich dadurch, welche Information unmittelbar (bottom-up) aus den Daten extrahiert wird, und ab wann (unbewußtes) Vorwissen (Hypothesen, Erwartungen und Erfahrung) mit in den Prozeß aufgenommen werden muß, um die erfaßte Information weiterzuverarbeiten (top-down). Die angenommenen Merkmale können sehr elementar und einfach sein, oder auch komplexe Reizmuster darstellen. Dieses Mosaik von Einzelinformationen wird durch die Gestaltprinzipien zu höheren Einheiten organisiert.

2.4 Wahrnehmung und Wissen

Eng verknüpft mit der Wahrnehmung ist das Problem des Wissens und seiner Repräsentation. Da Wahrnehmung nur auf der Basis einer internen Repräsentation der umgebenden Welt möglich ist, muß dem Organismus das hierzu nötige Vorwissen in irgendeiner Form vorliegen. Wie dieses erworben wird und auch dessen Speicherung ist bislang noch nicht geklärt und Gegenstand der Forschung. Auch von welcher Art die zugrundeliegenden elementaren Einheiten sind, ist weiterhin offen. Es lassen sich jedoch einige Beobachtungen anstellen, die auf bestimmte Strukturen hinweisen.

Reflektiert man über die Repräsentation unseres Wissens, so könnte man vermuten, daß es in klaren, geschlossenen Bildern organisiert ist. Der Schriftsteller Kempowski beschrieb in einem Vortrag sein Nachdenken damit, daß ein kleiner Idiot in seinem Kopf schnell ein Album mit Bildern durchblättere. Ebenso scheinen uns Erinnerungen gleichsam als Photographie wieder vor dem geistigen Auge zu entstehen. Psychologische Forschungen [Resnikoff 1989] zeigen jedoch, daß das menschliche Wissen nicht derart organisiert ist. Bei der Speicherung des Wissens erfolgt eine gewaltige Datenreduktion, d.h. eine Konzentration auf einige wesentliche Grundstrukturen. Beim Zugriff auf dieses Wissen wird interpoliert. Diese Fähigkeit, Komplexität zu ignorieren, gehört nach Ansicht der KI-Forscher zum gesunden Menschenverstand. Das physiologische System des Menschen besitzt die Fähigkeit, Lücken zu füllen. Was uns als kohärentes Wissen, als vollständiges Bild erscheint, ist in Wirklichkeit das Ergebnis einer Interpolation der selektiven "wichtigen" Information. Dies belegen z.B. Versuche, bei denen Kandidaten gepunktete, stilisierte Kreise gezeigt werden. In der anschließenden Beschreibung (in der

Erinnerung) sind es vollkommene, durchgezogene Kreise. Auch scheinbar klare Bilder von Vorgängen und Situationen sind somit immer Ergebnis einer Interpretation und nie der “unmittelbare” Sinneseindruck. Aus der Interpretation von Strichzeichnungen erhält man Hinweise darüber, daß elementare bzw. wichtige Information in den Ecken gespeichert ist. Anschaulich belegen läßt sich dieses Phänomen mit der bekannten Attneave’schen Katze [Resnikoff 1989] (vgl. Abbildung 2.3). Mit Methoden der Informationstheorie läßt sich nachweisen, daß die Information an den Ecken höher ist als auf Geraden.



Abbildung 2.3: Attneave’sche Katze: für die Interpretation wichtige Information ist in den Ecken gespeichert - die Verbindung dieser Eckpunkte ergibt bereits ein anschauliches Bild der Katze.

Wir haben den Eindruck, unsere Beschreibungen der uns umgebenden Dinge seien immer vollständig. Tatsache ist jedoch, daß das Wissen eines Organismus immer durch seine Entstehungsgeschichte geprägt ist. Daher fallen die “Weltbilder” unterschiedlicher Lebewesen auch unterschiedlich aus. Es hängt davon ab, was für den jeweiligen Organismus von Bedeutung ist. Eine Biene ist stark auf das Farbsehen angewiesen, um ihre Futterplätze zu finden. Sie wird daher ihr Futter (bestimmte Blumen) hauptsächlich an der Farbe erkennen. Ihr “Bild” der Pflanze (ihr Wissen über sie) ist also hauptsächlich durch die Farbe geprägt. Wenn wir Objekte erkennen, ist es uns meist nicht bewußt, daß wir es ebenfalls auf der Basis einiger kennzeichnender Merkmale tun, und daß unsere Merkmale keineswegs eine erschöpfende Beschreibung des Objekts darstellen. Ein Objekt besteht in unserer Vorstellung ausschließlich aus dieser uns bekannten Beschreibung. Daß es darüber hinaus noch Eigenschaften haben kann, wird meist übersehen. Diese weiteren Eigenschaften hatten in unserer Entwicklungsgeschichte offenbar keine Bedeutung, weshalb wir für sie keine Sensoren entwickelten. Die unterschiedlichen Bilder, welche sich verschiedene Organismen allerdings vom gleichen Objekt machen, sind nie widersprüchlich, es sind Facetten eines Gesamtbildes.

Wenn auch die Prozesse, die eine selektive Auswahl wichtiger Information bestimmen, noch nicht erforscht sind, so gibt es einige Prinzipien, die eine wichtige Rolle spielen [Resnikoff 1989]:

Streben nach einem maximalen Informationsgewinn: Information und ihre Kosten werden nach einem Ausgleichsprinzip gegeneinander aufgewogen. Dies führt dazu, daß Details, die sehr kostenintensiv zu speichern wären, weggelassen werden. Dieses Prinzip steuert auch die Wahrnehmung, indem die Gruppierung gewählt wird, welche die einfachsten „Erklärung“ für einen visuellen Reiz bietet.

Hierarchische Organisation der informationsverarbeitenden Strukturen: Hierarchische Ordnung findet man in Organisationen (größere Firmen, Behörden, Militär), aber auch in Computerarchitekturen und in biologischen Systemen (Kurz-, Langzeitgedächtnis, foveales - peripheres Sehen). Diese Struktur bringt eine Effizienzsteigerung mit sich, da durch die verschiedenen Hierarchieebenen eine Aufgabenverteilung erfolgen kann.

Ein Objekt ist nie das physikalische, geometrische Gebilde allein, sondern ist gleichzeitig in einen Zusammenhang mit anderen Objekten eingebunden. Fällt ein Begriff, so entstehen vor unserem “geistigen Auge” eine Fülle von Beziehungen zu anderen Objekten, gleichsam als würde man an einem Knoten eines Netzes ziehen und so alle damit verbundenen Knoten noch mitbewegen. Darüber hinaus sind mit Objekten auch noch weitere Informationen verbunden, z.B. die Art, wie man das Objekt handhabt oder als was es noch genutzt werden kann.

Nach Rosch (zitiert aus [Anderson 1989]) lassen sich verschiedene Abstraktionsebenen bei der Objektrepräsentation unterscheiden. Die allgemeinste Ebene, *basic level* steht im Zentrum. Beschreibungen dieser Ebene benutzt

der Mensch hauptsächlich. Sie basieren primär auf perzeptuellen Merkmalen. Der Mensch hat keine Schwierigkeiten, Merkmale von Objekten dieser Ebene anzugeben. Die nächst höhere Ebene gruppiert Objekte mit gemeinsamer Funktion oder Kategorie, während die nächst niedere Ebene eine Spezifikation des Objekts darstellt (vgl. Abbildung 2.4). Dies belegt, daß der Mensch mit hierarchischen Begriffsstrukturen arbeitet. Welche

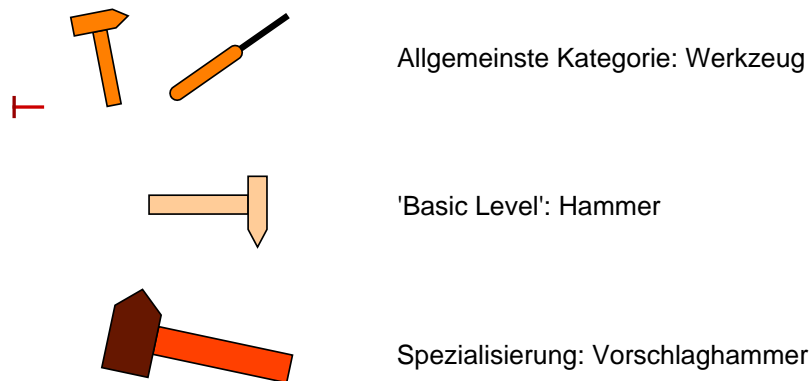


Abbildung 2.4: Beispiel für verschiedene Abstraktionsebenen eines Begriffs

weiteren verschiedenen Repräsentationen möglich sind, zeigen die Stufen der Repräsentation, die Kinder in ihrer Entwicklung bis zum 11 Lebensjahr durchlaufen:

Repräsentation in Bezug auf Aktionen: Beispiele hierfür sind: alles, was man werfen kann, ist ein Ball; alle Tiere sind ein „Wau“. Somit sind die Klassifikationen bei Kindern zunächst recht grob. Diese werden allerdings nach und nach verfeinert. Neues wird mit Bekanntem erklärt, bis Widersprüche auftreten. Diese bilden dann den Anstoß zur Verfeinerung einer Repräsentation.

Bildhafte Repräsentation: Ein Beleg, daß bei Kindern eine bildhafte Erinnerung vorliegt, verdeutlichen Versuche folgender Art: Kindern werden Bilder von Objekten gezeigt. Ein Wiedererkennen ist möglich, solange ein Objekt in der gleichen Weise präsentiert wird, wie in der ursprünglichen Vorlage, nicht aber, wenn das Objekt gedreht ist.

Abstrakte Regeln: Erst später lernt der Mensch abstrakte Regeln, mit denen er mit solcher Information umgehen kann. Auch bei Erwachsenen allerdings ist die Zeit, die gebraucht wird, um ein gedrehtes Objekt zu erkennen, proportional zum Drehwinkel [Boden 1989].

Es zeigt sich, daß verschiedene Repräsentationen für ein und dasselbe Objekt möglich sind. Diese hängen vom wahrnehmenden Subjekt und dessen Reifegrad, sowie vom Zweck und der jeweiligen Nutzung des Objekts ab.

2.5 Implikationen für das Maschinelle Sehen

Die Strukturen, die der Mensch bei der Geburt schon hat, reichen aus, in Interaktion mit der Umwelt alles weitere Wissen zu erwerben. Solche Strukturen müssen auch in einem Computersystem bereitgestellt werden, um ähnliche Fähigkeiten zu entwickeln. Für die Übertragbarkeit menschlicher Wahrnehmungsleistungen auf einen Computer müssen diese analysiert werden. Hier stellen sich die Fragen nach der Art, der Repräsentation und der Organisation der zugrundeliegenden Strukturen.

Das Bildverstehen hat die Aufgabe, die von einer Kamera (bzw. einem Sensor) erfaßte Information zu interpretieren, d.h. eine Beschreibung der abgebildeten Szene abzuleiten. Somit wird also das Ziel verfolgt, eine der menschlichen Wahrnehmung vergleichbare Leistung zu vollbringen. Aus der vorangehenden Analyse menschlichen Verhaltens ergeben sich somit folgende Implikationen für die Übertragung auf eine Maschine:

- Organisation der Wahrnehmung durch Gruppierung von Einzelmerkmalen zu komplexeren Objekten (d.h. Strukturen, Oberflächen, etc.).
- Nutzung von allgemeinen (generischen) Beschreibungen.

- Informationsreduktion durch Speicherung der wichtigen Information und Interpolation in den Lücken.
- Die Bedeutung der Integration verschiedener Wissensquellen: z.B. Farbe, Textur, Stereo, Bewegung.
- Wahrnehmung ist kein passiver Vorgang: durch selektive Aufmerksamkeit (active perception [Bajcsy 1985]) wird eine Konzentration der Information erreicht.
- Bedeutung des Wissen:
 - Das Wissen um diese Objekte (die Objektmodelle) kann in verschiedener Art abgelegt sein: von einfachen Merkmalslisten bis hin zu funktionalen oder generischen Modellen.
- Bedeutung des Wissenserwerbs durch Lernen:
 - Der Mensch hat nicht all sein Wissen von vornherein verfügbar, sondern entwickelt es im Laufe seines Lebens in Interaktion mit seiner Umwelt durch Lerntechniken weiter.

Einige dieser Prinzipien wurden bereits in die Methoden der Bildanalyse übernommen. Der Stand der Technik in der Bildverarbeitung und Mustererkennung ist im einführenden Kapitel 1.4 bereits ausgeführt worden. Die meisten Programme gehen von einem reinen visuellen perzeptuellen System aus und ignorieren daher wichtige Merkmale wie Farbe, Stereo, Bewegung. Vielfach wird jedoch das Fehlen einer grundlegenden Theorie des Bildverstehens beklagt [Haralick 1985].

Marr's Theorie [Marr 1982] versucht, die informationsverarbeitenden Schritte beim Sehen in einen Rahmen zu passen, der Physiologie, Psychologie und KI umspannt. Er fordert von einem vollständigen Informationsprozeß, daß drei Bestandteile vorhanden sind: Theorie, Algorithmen und Implementierung. Die Aufgabe der Objektkennung wird durch eine Rekonstruktion der Szene erreicht. Marr geht dabei - ähnlich wie Gibson - von der Vorstellung aus, daß aus einem Bild unmittelbar sehr viel mehr Information gewonnen werden kann als nur Helligkeitswerte. Insbesondere vertritt er die Ansicht, daß die Oberflächen im Bild gefunden werden können, ohne spezielles Modellwissen einzusetzen. Somit ist eine Figur-Grund-Trennung möglich. Was jedoch Figur und was Grund ist, wird erst durch die jeweilige Fragestellung ersichtlich. Dazu ist top-down-Wissen nötig.

Kritiker Marr's (z.B. [Sommer 1992]) sehen die Beschränkung auf rein visuelle Merkmale als starke Einschränkung. Es werden daher neuerdings Techniken vorgeschlagen, die alle mögliche Information der Szene in den Auswerteprozess mit einbezieht. Auch wird als Alternative zum Rekonstruktionsansatz eine mehr zweckgebundene Analyse vorgeschlagen. D.h. es soll eine Fokussierung auf das Wesentliche erfolgen (vgl. z.B. [Aloimonos 1990], [Pahlavan, Uhlin & Eklundh 1993]), wie es auch bei der menschlichen Wahrnehmung offensichtlich ist: will man zur Tür gehen, so sucht man nicht überall nach einem Ausgang, sondern nutzt sein Vorwissen über den Sinn und die Architektur von Türen (spatial reasoning). Ein Blick an die Decke oder auf den Fußboden wird daher zur Problemlösung nichts beitragen. Auch stellt sich die Frage, ob sich das assoziative Gedächtnis des Menschen allein auf symbolischer Ebene simulieren läßt. Hier zeigen sich deutliche Vorteile bei den subsymbolischen Ansätzen (Konnektionismus), die zum einen als kontext-adressierbare Speicher genutzt werden, aber auch "unscharfe" Modelle verarbeiten können.

Die Imitation menschlicher Lernfähigkeit erfordert ein System, welches aus elementaren Basisstrukturen mittels vorgegebener Methoden neues Wissen generieren kann. Durch Verfahren des Versuchs und Irrtums kann so aus einer aktuellen Situation eine bessere Anpassung an äußere Gegebenheiten erreicht werden. Dieser Lernvorgang gleicht einem Regelkreis, bei dem nach einer Aktion jeweils entschieden wird, ob die Situation sich in irgendeiner Form verbessert hat. "Besser" kann entweder im Hinblick auf ein vorgegebenes Ziel definiert sein, oder aber beispielsweise auch im Sinne der Informationstheorie als beste Beschreibung.

3 Wissen, Modelle und Repräsentation

Intelligenz bedeutet, daß ein informationsverarbeitendes System sich eine interne Repräsentation der umgebenden Welt gebildet hat, welche ihm hilft, sich in dieser Welt zu bewegen [Franck 1991]. Somit bestimmt die Repräsentation das Wesen der Intelligenz.

Die Fragen der Modellierung und der Wissensrepräsentation sind eng miteinander verknüpft. Jeder Wissensspeicherung geht ein Modellierungsschritt voraus, bei dem festgelegt wird, welcher Ausschnitt der Realität in das Modell mit aufgenommen wird. Dies beinhaltet zum einen die Auswahl geeigneter Merkmale des abzubildenden Wissens; hierzu zählt aber auch, welche Beziehungen zwischen der gegebenen Information existieren, welche Regeln vorliegen, sowie welche Schlußfolgerungen möglich sind. Diese Festlegungen erfolgen auf einer konzeptuellen Ebene. Die Modellierung legt somit den Abstraktionsgrad vom Übergang der “realen Welt” auf eine problembezogene Darstellung fest. Dieser Schritt läßt sich nie korrekt abbilden, immer ist der Grad der Vollständigkeit eine Funktion der zur Verfügung stehenden Merkmale [Schalkoff 1989].

Dieses Modell muß in einem Computer abgebildet werden. Hierzu existieren eine Reihe von Repräsentationsformalismen. Die Wahl einer bestimmten Darstellungsform ist von der jeweiligen Aufgabe und vom Aufgabenziel abhängig. Sie bestimmt damit auch die Möglichkeiten und die Effizienz der Analyseprozesse.

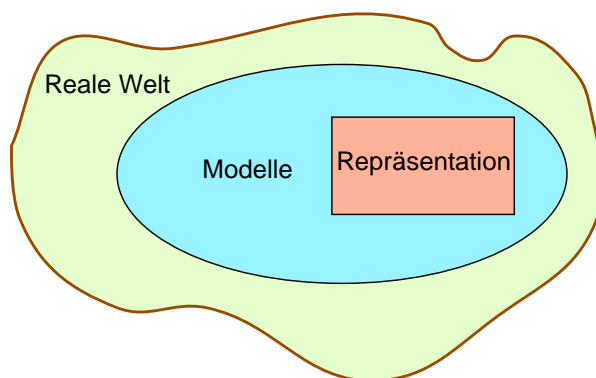


Abbildung 3.1: Von der Realität zur Repräsentation

Der Übergang von der Realität auf die Modelle und von dort auf die Repräsentationen ist immer mit einem Verlust behaftet. Abhängig von der Sicht auf die Realität können unterschiedliche Repräsentationen günstig sein. Die Modelle sind gekennzeichnet durch eine Auswahl an bestimmten charakteristischen Merkmalen der Realität. Sie stellen damit eine bestimmte Sicht auf die Realität dar. In der Repräsentation werden diese Merkmale in einer bestimmten Sprache beschrieben - sie bildet damit eine Sicht auf das Modell (Abbildung 3.1).

Nach einer allgemeinen Definition des Modellbegriffs werden schwerpunktmäßig die Modelle im Hinblick auf die Bildanalyse betrachtet und somit auf Objektmodelle eingeschränkt. Diese Modelle lassen sich nach unterschiedlichen Kriterien einteilen, wobei die Klassifikation nach spezifischen und generischen Objektmodellen im Vordergrund steht. Anschließend wird eine Übersicht über gängige Repräsentationsformen im KI-Bereich gegeben. Diese Darstellung wird der Vollständigkeit halber angeführt, um die später verwendete Repräsentationsform zu motivieren.

3.1 Vorüberlegungen und Definitionen

Der Modellbegriff ist sehr vielschichtig. Wenn auch für manche Bereiche fest steht, was Modell bedeutet, so fallen beispielsweise die Abgrenzungen zum Begriff des Musters und dem des Wissens im allgemeinen nicht leicht.

Es werden nun kurz einige Stichworte aufgeführt, die intuitiv mit dem Begriff des Modells verknüpft sind.

- Modell im Sinne von “Model“, d.h. ein getreues Abbild eines Objekts, eine Vorlage, nach der es (meist durch Abdruck) erzeugt werden kann, z. B. : Buttermodel, Gipsabdruck,

- Modell eines Künstlers: die Vorlage für ein Kunstwerk,
- Modell als idealisiertes Abbild des Objekts,
- Modell als Abbild einer Szene, d.h. das Zusammenspiel von Einzelobjekten,
- Mathematisch-physikalische Modelle, die beispielsweise die Verknüpfung zwischen Beobachtungen und gesuchten Größen in einem Experiment angeben,
- Modell als Abbild eines Vorgangs,
- Prozeßbezogene Modelle: Strategien, Konstruktionshinweise, Pläne, Abfolgen von Ereignissen,
- Wissen stellt immer ein Ergebnis eines Abstraktionsprozesses dar. Daher kann Wissen auch als Modell verstanden werden.

Der *Brockhaus* [1993] definiert das Modell der (Natur-) Wissenschaft als:

Modell soll die als richtig angesehenen Eigenschaften des Vorbilds ausdrücken, die unwichtigen oder nebensächlichen außer Acht lassen, um durch Vereinfachungen zu einem übersichtbaren oder mathematisch berechenbaren oder zu experimentellen Untersuchungen geeigneten Modell zu kommen.

Weiter gilt, daß

biologische, physiologische Funktionsweisen durch Modelle und Modellversuche erhellt werden können.

In der Soziologie ist ein Modell

ein begrifflich oder intuitiv anschaulich gewonnenes Bild von Personen, Personengruppen, gesellschaftlichen Zusammenhängen und Leistungen, das der verstehenden Aufhellung der sozialen Wirklichkeit dienen soll.

Welcher Art das angesprochene zu modellierende "Vorbild" ist, ist in den Definitionen nicht festgelegt und von Fall zu Fall zu unterscheiden. Damit können Modelle inhaltlich verschiedener Art sein: sie sind zunächst als objektbezogen anzusehen, wenn es darum geht, ein Abbild eines gegenständlichen Vorbilds zu erzeugen. Dabei kann es sich um ein Objekt handeln, oder auch um eine Gruppierung von Einzelobjekten in einer Szene und somit eine übergeordnete Darstellung betreffen. Auf diese Weise werden nicht nur Einzelobjekte, sondern eine Ansammlung zusammengehöriger Dinge in einem Szenenmodell beschrieben. Auf der anderen Seite können Modelle auch Abläufe oder Strategien beschreiben, die sie damit in idealisierter, abstrahierter Form darstellen. Auch diese Modelle müssen alle als wesentlich erachteten Komponenten enthalten. Zu dieser Art von Modell zählen beispielsweise mathematische, physikalische Beziehungen, wie Transformationen, Abbildungsvorschriften oder Projektionen, aber auch das Wissen um die Abfolge bestimmter Vorgänge oder Strategien zur Konfliktlösung. Ein Modell steht daher für die Beschreibung unserer umgebenden Welt und umfaßt Objekte, Begriffe und Prozesse.

3.2 Klassifizierung von Objektmodellen

Die hier betrachteten Modelle sollen für die Bildinterpretation genutzt werden. Daher stehen Objekt- oder Szenenmodelle im Vordergrund. Ebenfalls erfolgt eine Konzentration auf die Gegebenheiten der visuellen Wahrnehmung, was die Art der zur Verfügung stehenden Modelle und deren Eigenschaften beeinflußt.

Eine klassifizierende Einteilung von Modellen kann nach verschiedenen Kriterien erfolgen. Zunächst lassen sie sich nach dem Grad ihrer Spezialisierung unterscheiden. Die folgende Einteilung reflektiert dabei eine zunehmende Generalisierung vom Spezifischen zum Allgemeinen.

- Spezifische Modelle kennzeichnen Beschreibungen, die ein Objekt (möglichst) vollständig wiedergeben. Solch eine Beschreibung ermöglicht es, ein individuelles Objekt eindeutig zu erkennen. Sie gibt alle Details des Objekts wieder, die für die Erkennung und eindeutige Klassifizierung nötig sind. Diese sogenannten deskriptiven Modelle können aber auch für die Erzeugung eines synthetischen Objekt-(bild-)es genutzt werden. Sie entsprechen damit einer Art Bauanleitung für ein bestimmtes Objekt.
- Generische oder allgemeine Modelle beschreiben Klassen von Objekten. Mit solchen Modellen lassen sich Objekte als einer Klasse zugehörig erkennen. Diese Art von Modellen kann man als diskriminatorisch bezeichnen, da sie geeignet sind, bestimmte Objektklassen zu identifizieren, bzw. von anderen abzugrenzen. Sie enthalten allerdings nicht die Eigenschaften von Individualobjekten und erlauben daher nicht die Konstruktion von Objekten. Erst in Verbindung mit einem realen Objekt können die fehlenden Attribute und Parameter festgelegt werden. Ansonsten stellen sie lediglich "leere Hüllen" dar, die erst im Interpretationsprozeß gefüllt werden.
- Eine weitere Verallgemeinerung führt schließlich zu übergeordneten Modellen, die unabhängig von einem spezifischen Objekt gelten. Sie sind dazu da, generell Formen und Gestalten in einer unspezifizierten Ansammlung von Sinneseindrücken zu erkennen (siehe Abbildung 3.3 auf Seite 25).

Die Unterscheidung in diese drei Arten von Modellen muß jeweils im Rahmen des Aufgabenziels gesehen werden. Obwohl beim Übergang vom allgemeinen zum spezifischen Modell immer mehr Detailinformation über das Objekt hinzukommt, stellen alle Arten von Beschreibungen Modelle im oben definierten Sinne dar. Ist das Ziel der Aufgabe, mögliche Objektkandidaten im Bild zu erkennen, so werden sehr generelle Objekteigenschaften benötigt. Die Beschreibung umfaßt somit nur sehr wenige Schlüsselaspekte von Objekten, wie sie z.B. in den Gestaltgesetzen angegeben sind. Wird hingegen ein spezifisches Objekt gesucht, so wird dazu eine möglichst genaue Beschreibung des Objekts benötigt. Sie muß so detailgetreu und ausführlich sein, um das Objekt verwechslungsfrei erkennen zu können. Dennoch sind auch dieses lediglich die Aspekte, die für diese konkrete Aufgabe des Erkennens nötig sind. Das reale Objekt kann noch weitere Eigenschaften haben, die für die Erfüllung des Aufgabenziels keine Bedeutung haben und damit nicht angegeben sind (z.B. wie es riecht, wie es sich anfühlt oder ob man es greifen kann).

3.2.1 Spezifische Modelle

Eine spezifische Beschreibung charakterisiert alle Eigenschaften eines Objekts, die für eine bestimmte Aufgabe benötigt werden. Die Pläne eines Architekten enthalten beispielsweise alle Maße und Materialangaben, die nötig sind, um ein Haus zu bauen. Somit sind spezifische Modelle vollständige Beschreibungen - allerdings immer in Relation zur gestellten Aufgabe⁴. Spezifische Modelle werden für die Objektbeschreibungen im CAD- oder GIS-Bereich eingesetzt. Es lassen sich verschiedene geometrische Modelle differenzieren. Sie reichen von parametrischen Darstellungen über Enumerationsverfahren und Randdarstellungen (Boundary Representation) hin zu Beschreibungen mittels konstruktiven Verknüpfungen von Volumenprimitiven (CSG - Constructive Solid Geometry) [Bill & Fritsch 1991], [Meier 1986] (vgl. Abbildung 3.2). CAD - bzw. GIS-Modelle sind vollständig

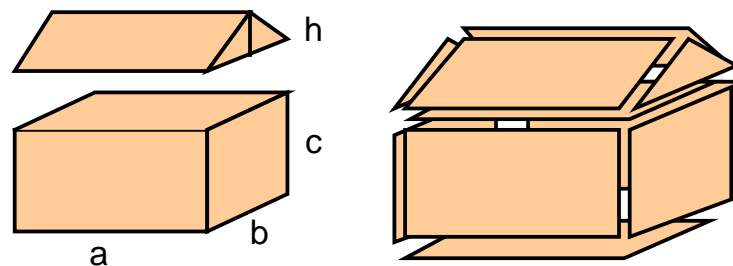


Abbildung 3.2: Zwei Möglichkeiten des geometrischen Modellierens: parametrische Beschreibung (links), Randdarstellung (rechts)

⁴Wichtig für den Aspekt der Modellierung ist es, daß die beschriebene Information erst in einem Interpretationsschritt verwertet werden kann (z.B. durch Nutzung der Legende des Plans). Ferner enthält diese Beschreibung normalerweise keine Angaben über Relationen zwischen einzelnen Objekten. Die Information, daß sich das Dach auf der Hausgrundfläche befindet, ist nur implizit durch seine geometrische Lage (Koordinaten) gegeben. Für die Bildinterpretation sind solche expliziten Angaben allerdings fundamental.

beschrieben und können daher nicht unmittelbar zur Erfassung leicht abweichender oder gar neuer Objekte eingesetzt werden [Braun, Kolbe, Lang, Schickler, Steinhage, Cremers, Förstner & Plümer 1994].

3.2.2 Generische oder allgemeine Modelle

Generische Modelle gelten nicht nur für ein bestimmtes Objekt, sondern beschreiben eine Kategorie gleichartiger Objekte. Sie haben deklarativen Charakter und generalisieren die spezifischen Eigenschaften einzelner Objekte zu einer Klassenbeschreibung. Generische Objektmodelle enthalten freie Parameter, die erst mit ihrer Nutzung in einem Interpretationsprozeß festgelegt werden. Ein "Kreis" um einen gegebenen Punkt ist beispielsweise eine generische Beschreibung, da der Radius - als die einzige freie Größe - noch nicht festliegt. Wird allerdings in der Realität (z.B. im Bild) eine Kreisstruktur gefunden, so kann durch die Bildinformation die zur spezifischen Beschreibung fehlende Größe fixiert werden. Dies bringt mit sich, daß eine generische Beschreibung zur Konstruktion eines spezifischen Objekts (Synthese) nicht verwendet werden kann. Die generische Beschreibung eines Tisches könnte beispielsweise lauten: ein Tisch hat eine Tischplatte, die auf vier Füßen steht. Mit dieser Aussage sind keine Angaben über die Größe und die Anordnung dieser Objektteile gemacht, d.h. ein Schreiner könnte allein aus diesen Angaben einen Tisch nicht bauen. Allerdings kann ein Tisch als solcher allein aus diesen Merkmalen erkannt werden. Dies kann durch Erkennen der Objektteile erfolgen. Ist die Identifikation allerdings erfolgt, so ist damit anschließend eine Rekonstruktion möglich. Die Bildinformation liefert die festen Größen für die freien Parameter (Größe, Form, Abstand, Position) der generischen Modellbeschreibung.

Generische Beschreibungen können in ein hierarchisches Modell eingeordnet werden. Hier lassen sie sich als Zwischenstufe auf dem Weg zum spezifischen Objektmodell ansehen. Sie enthalten alle wesentlichen Merkmale einer Objektklasse, ohne jedoch bestimmte Ausprägungen eines einzigen Objektes festzulegen. Auf diese Weise lassen sich generische Beschreibungen als 'leere Hüllen' ansehen. Sie geben vor, über welche Teile und Merkmale ein Objekt dieser Klasse verfügt, gegebenenfalls sind auch Angaben über mögliche Wertebereiche der Parameter gemacht. Die realen Werte eines spezifischen Objekts ergeben sich jedoch erst im Moment der Interpretation, wenn dieses Modell mit einem realen Objekt verglichen und instanziiert wird.

Allgemeine Darstellungen enthalten lediglich die wichtigsten Merkmale von Objekten und lassen die weiteren offen. Eine Reihe von Programmen zum Erkennen polyedrischer Objekte basieren auf der Analyse von Knoten, in denen die einzelnen Objektkanten zusammenkommen ([Guzman 1968], [Clowes 1971]) sowie der Objektkanten selbst [Waltz 1975]. Die Grundidee besteht darin, daß die Art der Knoten gewisse Rückschlüsse zuläßt auf die Flächen, die in diesem Punkt zusammenkommen. Waltz ermittelte eine erstaunlich geringe Anzahl von Bedingungen, die zur Konsistenzprüfung von Polyedern genügen. Diese ermöglichen eine Interpretation von Strichzeichnungen, mit der Entscheidung, ob die dargestellten Objekte der Klasse Polyeder angehören oder nicht. Die Konsistenzbedingungen wurden durch Beobachtung ermittelt und zeigten sich gültig für eine Klasse von Objekten.

Eine weitere Art von generischen Modellen läßt sich über ihren Entstehungsprozeß definieren. Solche dynamischen prozeduralen Modelle können z.B. Produktionssysteme oder Syntaktische Grammatiken sein. Hier erfolgt die Objektbeschreibung in Form von Produktionsregeln [Fu 1982].

3.2.3 Übergeordnete Modelle

Im Kapitel zur Wahrnehmung wurden derartige Modelle bereits angesprochen. Sie dienen dazu, die Beobachtung bzw. das Bild in potentiell bedeutungsvolle Einheiten zu untergliedern, und somit Objekt und Hintergrund zu trennen. Diese Modelle sind beispielsweise in den Gestaltungsgesetzen formuliert. Sie beschreiben Organisationsmöglichkeiten wie das Gesetz der Nähe, der guten Form, der Ähnlichkeit, der Umschlossenheit oder der guten Fortsetzung, aber auch Starrheits- und Glattheitsbedingungen.

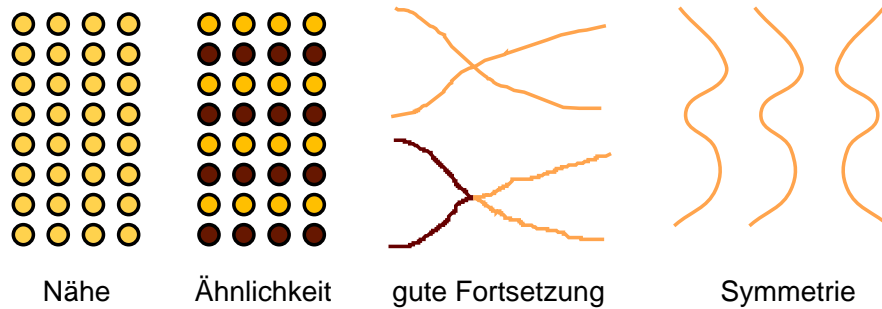


Abbildung 3.3: Beispiele für Gestaltgesetze

Gemäß dem Gesetz der Nähe werden die Punkte in Abbildung 3.3 in vertikale Reihen gruppiert, im zweiten Falle allerdings - aufgrund der Ähnlichkeit - eher horizontale Reihen. Das Gesetz der guten Fortsetzung besagt, daß die beiden Kurven als jeweils von links nach rechts durchgängig betrachtet werden und nicht - wie unten - als zwei Kurven, die sich im Schnittpunkt treffen und dann nach links bzw. rechts abknicken. In der letzten Figur werden die beiden rechten Kurven aufgrund ihrer Symmetrie zusammengefaßt.

Ist eine solche Organisation nicht möglich, so kann nichts erkannt werden. Als Beispiel sei hier die Tarnung von Tieren genannt. Gliedert sich das Tier vollkommen in seine Umgebung ein, so sind verschiedene Gruppierungen möglich⁵.

3.3 Objektmerkmale

Die Modelle werden üblicherweise über ihre Eigenschaften oder kennzeichnenden Merkmale definiert. Nur solche Merkmale können allerdings sinnvollerweise für die Modellierung in der Bildanalyse herangezogen werden, die auch beobachtbar sind: beispielsweise ist es unsinnig, zur Analyse von Schwarz-Weiß-Bildern das Merkmal "Farbe" heranzuziehen.

Es stehen eine Reihe von Merkmalen zur Verfügung, die zunächst durch quantitative und durch qualitative Aspekte beschrieben werden können. Sie lassen sich folgendermaßen unterteilen:

- geometrische Merkmale (quantitativ),
- topologische Merkmale (quantitativ),
- physikalische Merkmale (quantitativ und qualitativ),
- semantische Merkmale (qualitativ).

Im Bereich der Bildverarbeitung stehen die geometrischen, quantitativen Merkmale im Vordergrund. Hier werden Objekte meist global über ihre Form beschrieben. Dazu sind Merkmale wie Länge, Form, Größe nötig. In der Binärbildanalyse sind topologische Merkmale wie z.B. "Anzahl der Löcher" verbreitet. Für Fernerkundungszwecke werden vornehmlich die spektralen Eigenschaften der Objekte mit Hilfe spezieller Sensoren aufgenommen und können somit über diese wieder klassifiziert werden. Daher werden hier meist physikalische Eigenschaften genutzt.

Über unmittelbare Objekteigenschaften hinaus lassen sich Objekte auch noch mittels weiterer Merkmale beschreiben:

Kontextbezogene Merkmale: Eine Objektbeschreibung kann also den Bezug zu anderen Objekten mit aufnehmen. So läßt sich beispielsweise ein Getränk als das, was sich in der Tasse befindet, angeben. In manchen Fällen kann der Kontext bereits das ausschlaggebende Merkmal sein. Vielfach wird er jedoch unterstützend als zusätzliche Eigenschaft mit herangezogen.

⁵Vergleiche hierzu auch die Abbildung von Seite 15. Es handelt sich dort um einen Hund, einen Dalmatiner, der sich - etwa in der Bildmitte - schnüffelnd vom Betrachter wegbewegt.

Funktionsbezogene Merkmale: Weiter lassen sich Objekte bezüglich ihrer Funktion beschreiben. So kann ein Objekt als etwas zum Werfen oder als etwas Trinkbares definiert werden. Hier steht dann die Funktion, seine Gebrauchsmöglichkeit im Vordergrund. Wichtig ist natürlich, daß sich diese Funktion in einem visuellen Merkmal ausdrückt.

3.4 Prozedurale Modelle

Prozedurale Modelle beschreiben Abläufe, Strategien sowie Wissen über die Aufeinanderfolge einzelner Prozesse bzw. deren Zusammenhänge. Damit liegen diese Art von Modellen allen wissenschaftlichen Untersuchungen zugrunde. Sie formulieren den (angenommenen, meist idealisierten) Zusammenhang zwischen gegebenen und gesuchten Größen. Ist dieser Zusammenhang nicht deterministisch, so sind Kontroll- bzw. Steuerungsstrukturen nötig [Niemann 1981], um ihn offenzulegen. Zwischen einer Ausgangssituation und einem Zielzustand sind eine Serie von Zwischenschritten einzuschalten. Das Problem stellt sich somit als Suchproblem dar.

In Programmsystemen ist dieses Kontrollwissen meist explizit kodiert, d.h. die angewandten Strategien sind fix. Für den Bereich der Bildanalyse besteht dieses Wissen beispielsweise in der Abfolge bestimmter Operationen zur Lösung einer Erkennungsaufgabe. Da diese Aufgabenschritte oft objektabhängig sind, muß hier die Ablaufstrategie offengelegt und damit explizit gemacht werden. Neuere Repräsentationsformen bzw. Inferenzstrategien können dies prinzipiell leisten (Regelbasierte Systeme, Semantische Netze, Objektorientierte Daten- und Wissensverwaltung).

3.5 Formalismen zur Wissensdarstellung

Die Abbildung der Modelle in einer computergerechten Form wird über Repräsentationsformalismen gelöst. In den Anfängen der KI wurde die Wissensrepräsentation nicht als eigenständiges Problem angesehen. Vielmehr wurde das Wissen meist implizit im Programm oder in den Datenstrukturen abgelegt. Erst seit Ende der 60 iger Jahre existiert die Wissensrepräsentation als eigene Disziplin [Sagerer 1985].

Wissensrepräsentation beschäftigt sich ganz allgemein damit,

in einer Sprache oder einem kommunikativen Medium Beschreibungen oder Bilder, die in besonderer Weise der Welt oder einem Zustand der Welt entsprechen, anzufertigen. In der KI geht es darum, Beschreibungen der Welt in solch einer Weise darzustellen, daß eine intelligente Maschine durch formale Manipulation dieser Beschreibungen zu neuen Schlüssen über ihre Umgebung kommen kann. (Zitiert nach [Brachman & Levesque 1985])

Diese Definition beinhaltet, was Reimer [1991] für eine Wissensrepräsentation fordert: die wesentlichen Bestandteile einer Repräsentation sind eine formale Sprache sowie eine Interpretationsvorschrift, welche die Manipulationsmöglichkeiten des Wissens steuert. Auch Sommer [1992] betont, Repräsentation bedeute Wissen in Verbindung mit Zugriff auf dieses. Ein Datenstruktur alleine stellt noch kein Wissen dar, es ist vergleichbar mit einer Enzyklopädie, die erst durch einen intelligenten Leser ihren Wert erlangt [Barr & Feigenbaum 1982].

Der Mensch beschreibt seine Umgebung üblicherweise in natürlicher Sprache. Diese stellt eine sehr mächtige Repräsentationsform für Wissen dar. Allerdings ist die für eine maschinelle Verarbeitung nicht geeignet, da sie zuviel Allgemeinwissen voraussetzt [Campbell 1989] und zu komplex ist. In der Informatik wurden verschiedene Repräsentationsformen entwickelt. Versuche, formale Sprachen zur Wissensdarstellung zu entwickeln, orientierten sich zum einen an einem bereits bekannten und durchgehend erforschten Schema, nämlich der mathematischen Logik. Hierauf basierende Entwicklungen führten zu Produktionssystemen, die das Wissen in Form von Fakten und Regeln darstellen. Andererseits kamen aber auch Impulse aus der Psychologie, die nach einem Modell für das menschliche Gedächtnis und seine Informationsverarbeitung sucht. Hieraus entstanden zwei wichtige Repräsentationsformen, die Semantischen Netze und die Frames.

Die Grenzen zwischen den verschiedenen Ansätzen sind fließend. Grundsätzlich können die Repräsentationsformen ineinander überführt werden. Manche Darstellungsformen eignen sich allerdings für bestimmte Aufgaben besser als andere. Es existiert allerdings kein formales Maß für die Adäquatheit eines Schemas. Diese muß immer im Hinblick auf eine bestimmte Zielfunktion gesehen werden, z.B. die Einfachheit oder Natürlichkeit der Darstellung.

Im folgenden werden einige wichtige Repräsentationsformen vorgestellt (vgl. KI-Literatur, z.B.: [Nilsson 1982], [Rich 1988], [Winston 1977], [Ringland & Duce 1988], [Sharples, Hogg, Hutchinson, Torrance & Young 1989]). Diese Übersicht soll das Spektrum der Möglichkeiten andeuten, und dadurch auch die später verwendete Repräsentationsform motivieren. Bei der Darstellung erfolgt eine Unterscheidung danach, ob die Repräsentation mehr auf Objekte ausgerichtet ist, die in bestimmten Beziehungen zueinander stehen, oder eher unabhängig davon organisiert ist.

3.5.1 Nicht-Objektbezogene Repräsentationsansätze

Logik

Frühe Ansätze der Wissensdarstellung nutzen den Formalismus der mathematischen Logik. Hiermit soll das menschliche Schlußfolgern nachgebildet werden. Die Aussagenlogik kennt die einzelne Begriffe sowie vorgegebene Inferenzregeln, mit denen Wahrheitswerte (wahr oder falsch) abgeleitet werden können. Diese Deduktionen sind immer korrekt. Eine Erweiterung der Aussagenlogik stellt die Prädikatenlogik 1. Ordnung dar, welche zusätzlich Prädikats- und Funktionssymbole kennt. Damit lassen sich zusätzlich Aussagen über Objekte selbst und ihre Beziehung zu anderen Objekten machen. Die Aussage “Jedes Haus hat ein Dach” läßt sich mittels Prädikatenlogik folgendermaßen ausdrücken:

$$\forall x.Haus(x) \mapsto hat_dach(x),$$

d.h. für jedes Objekt x , welches ein Haus ist, gilt, daß es ein Dach hat. Mit vorgegebenen Inferenzregeln kann neues Wissen abgeleitet werden. Diese Inferenzen lassen sich mittels Theorem-Beweisern automatisieren.

Die Macht der Logik besteht nicht darin, was ausgedrückt werden kann, sondern was ungesagt bleiben kann. Beispielsweise impliziert das Axiom “Alle Rechtsanwälte aus Chicago sind schlau” [Winograd 1985] eine ganze Reihe von Ableitungen, die andere Formalismen explizit beschreiben müssen, z.B. den Schluß, daß eine dumme Person aus Chicago kein Rechtsanwalt sein kann, bzw. daß ein dummer Rechtsanwalt nicht aus Chicago stammen kann.

Der Vorteil logikbasierter Repräsentation liegt in der klaren Syntax und Semantik. Die Modularität garantiert weitgehend unabhängiges Einfügen und Verändern von Fakten. Der Inferenzprozeß ist allerdings für große Datenmengen aufwendig, da dieser datengetrieben erfolgt: prinzipiell kann eine Regel auf alle Fakten der Datenbank angewandt werden, was ein exponentielles Anwachsen der Möglichkeiten bedeutet.

Es existieren zahlreiche Erweiterungen der Logik.

Nicht-Monotones Schließen: Das Wissen um die Welt ist Änderungen unterworfen. Daher kann es vorkommen, daß neue Information hinzukommt, die zu früherem Wissen im Widerspruch steht. Alle Folgerungen, die von den früheren falschen Prämissen ausgehen, müssen revidiert werden [Brewka 1989].

Mehrwertige Logik: Während die mathematische Logik lediglich zwei Wahrheitwerte - wahr oder falsch - kennt, können in der “Fuzzy Logic” (unscharfe Logik) [Zadeh 1974] weitere Zwischenstufen unterschieden werden, um Ausdrücke zu qualifizieren, die weder wahr noch falsch sind. Durch sog. Zugehörigkeitsfunktionen können sprachliche Ausdrücke einer bestimmten Wertemenge zugeordnet werden (z.B.: ein Auto mit Maximalgeschwindigkeit von 200 km/h ist “schnell” mit einer Zugehörigkeit von 1.0, eines mit 160 km/h kann nur noch mit 0.7 zur Klasse der schnellen Autos gerechnet werden, zu 0.3 zu der mittleren). Der wesentliche Vorteil der Fuzzy Logic besteht in der relativ einfachen Verknüpfbarkeit zusammenhängender Prozesse und deren jeweiligen Zugehörigkeiten. Dies führt insbesondere dazu, daß komplexe Schaltkreise auf diese Weise sehr kompakt gehalten werden können (z.B. Fuzzy Kamera- bzw. Waschmaschinensteuerung).

Modallogik: Diese Erweiterung erlaubt die Darstellung von Modalitäten (kann, nötig), bzw. die Darstellung von unvollständigem Wissen (z.B.: “Karl ist entweder Polizist oder Lehrer”).

Die Programmiersprache Prolog [Clocksin & Mellish 1981] basiert auf Prädikatenlogik erster Ordnung. Mit Prolog wird das Wissen deklarativ beschrieben. Schlußfolgerungen sind nach dem Prinzip der Rückwärtsverkettung realisiert.

Produktionssysteme

Newell [1973] führte die Produktionssysteme als Modell menschlicher Begründungsprozesse ein. Ein Produktionssystem besteht aus einem Satz von Produktionsregeln, einer Faktenbasis, sowie einer Kontrollstrategie. Aufgabe dieses sog. Inferenzmechanismus ist, die Anwendbarkeit der Regeln zu prüfen bzw. die geeigneten auszuwählen, diese auf die Fakten anzuwenden und Konflikte zu lösen. Regeln sind derart organisiert, daß Vorbedingungen formuliert werden, die zu bestimmten Aktionen führen. Üblicherweise hat eine Regel folgende Form:

```
WENN Praemisse (Vorbedingungen)
DANN Konklusion (Aktion)
```

Ein Haus kann somit in Form einer Regel beschrieben werden:

```
WENN
  hat_Tuer(x)
  hat_Fenster(x)
  hat_Dach(x)
  hat_Wand(x)
  ...
DANN
  ist_Haus(x)
```

Auf diese Weise kann ein unbekanntes Objekt x als Haus erkannt werden, wenn es Türen, Fenster, Dach, ... hat. Die Vorbedingungen müssen dabei in der Faktenbasis enthalten sein, die Aktion ihrerseits können Änderungen in der Faktenbasis bewirken.

Produktionssysteme finden großen Einsatz im Bereich der Expertensysteme. Diese Methode bietet sich hier auch besonders an, läßt sich das Expertenwissen doch vielfach in Form von Fakten und Regeln beschreiben. Mit Produktionssystemen lassen sich klare Abläufe bzw. Folgen von Einzelaktionen sehr gut repräsentieren. Die Darstellung ist einheitlich und klar, die Regeln sind intuitiv verständlich. Durch die Aufstellung einzelner Fakten und Produktionsregeln erreicht das System eine starke Modularität. Regeln können unabhängig voneinander formuliert werden. Die Modularität ermöglicht einfaches Programmieren, bewirkt aber auf der anderen Seite, daß die Zusammenhänge zwischen den einzelnen Regeln mitunter schwer zu überprüfen sind. Somit ist der Kontrollfluß oft nicht leicht zu überblicken. Die einzelnen Regeln nehmen nicht direkt aufeinander Bezug, sondern nur indirekt über die Fakten der Datenbasis.

Wichtig ist, daß der regelbasierte Ansatz erlaubt, nicht-monotone Wissensübergänge darzustellen. Außerdem ist es möglich, jede Regel mit einem sog. Sicherheitsfaktor zu versehen, so daß auch unsicheres Wissen beschrieben werden kann. Dies wurde insbesondere im Expertensystem MYCIN [Shortliffe 1976] erfolgreich angewandt. Auch lassen sich Regeln als Meta-Regeln verstehen und zur Ablaufsteuerung einsetzen.

Formale Grammatiken

Eine Grammatik besteht aus einem Alphabet (den terminalen und nicht-terminalen Symbolen), einem Startsymbol, sowie einer Menge von Regeln, die angeben, welche Symbole durch andere ersetzt werden können. Ausgehend von einem Startsymbol kann jeder Satz im Rahmen der Grammatik erzeugt werden.

Grammatiken erlauben zum einen die Erzeugung neuer Sätze, aber auch die Entscheidung, ob ein unbekannter Satz mit der gegebenen Grammatik erklärbar ist [Cohen & Feigenbaum 1982]. Diese Repräsentationsform wird vorwiegend in der Sprachforschung eingesetzt. Interpretiert man allerdings ein Muster als aus verschiedenen Teilobjekten zusammengesetzt, so läßt sie eine Grammatik auch in der Mustererkennung einsetzen [Fu 1982], [Niemann 1981]. Ein Muster entspricht damit einem Satz. Eine Grammatik besteht aus einem terminalen und nicht-terminalen Alphabet, einem Startkonzept, sowie Regeln, die angeben, wie die einzelnen Sprachkonstrukte miteinander zu verknüpfen sind. Ein Haus kann beispielsweise folgendermaßen generiert werden (vgl. [Fu 1980]):

$$\langle house \rangle \longrightarrow left_of(\langle house \rangle, \langle sideview \rangle)$$

$$\begin{aligned}
\langle house \rangle &\longrightarrow \langle frontview \rangle \\
\langle frontview \rangle &\longrightarrow on_top_of(\langle gable \rangle, \langle wall \rangle) \\
\langle gable \rangle &\longrightarrow \triangle \\
\langle gable \rangle &\longrightarrow on_top_of(\langle chimney \rangle, \triangle) \\
\langle wall \rangle &\longrightarrow \square \\
\dots &\longrightarrow \dots
\end{aligned}$$

Wird als Startkonzept $\langle house \rangle$ angegeben, so kann dieses erzeugt werden, indem die nicht-terminalen Symbole (Symbole in spitzen Klammern) der Regeln solange expandiert werden, bis nur noch terminale Symbole (hier: graphische Symbole Quadrat, Dreieck) im Satz enthalten sind. Ein Haus kann dann beispielsweise ein aus einer Frontansicht bestehen, diese wiederum setzt sich aus einer Wand und einem Dach zusammen. Wand und Dach sind im einfachsten Fall Dreieck über einem Quadrat.

Attribut-Wert-Listen

Diese Form der Beschreibung setzt voraus, daß sich die Konzepte über eine Sammlung von Eigenschaften beschreiben lassen, die als Listen gespeichert werden. Diese Repräsentationsform ist sehr einfach und leicht zu implementieren.

```

Haus1: [Groesse: 100 qm, Anzahl Fenster: 20, Anzahl Tueren: 2, ...]
Haus2: [Groesse: 120 qm, Anzahl Fenster: 24, Anzahl Tueren: 5, ...]
...

```

Allerdings lassen sich regelhafte Zusammenhänge zwischen einzelnen Objekten nicht ohne Weiteres darstellen. Dazu müssen diese Relationen als Funktionen von Attributen beschrieben werden. Im allgemeinen bläht sich die Merkmalsliste auf diese Weise stark auf. Solch eine Transformation ist daher nur sinnvoll durchführbar, wenn nur wenige Relationen beteiligt sind.

3.5.2 Objektbezogene Repräsentationsansätze

Frames

Minsky [1975] führte den Formalismus der Frames ein. Er geht zurück auf den Schema-Begriff der Kognitionspsychologie und versucht, ein Modell des menschlichen Gedächtnisses zu bilden. Der Ansatz basiert auf der Annahme, daß sich das menschliche Gedächtnis an Standardsituationen orientiert. Wie im Kapitel 2 dargestellt, legt der Mensch sein Wissen in Stereotypen ab. Diese stellen allgemeine Klassenbeschreibungen von gängigen Situationen bzw. Objekten dar⁶. Für diese Stereotypen werden sog. Rahmen (Frames) bereitgestellt. Kennzeichnend für Frames ist eine starke Objektzentrierung: das Wissen über eine Struktur ist zusammenhängend in einem Frame repräsentiert. Ein Frame besteht dabei aus einer Reihe von Einzeleinträgen (Slots), in denen die Eigenschaften des jeweiligen Objekts formuliert sind. Für Standardsituationen sind die Slots bereits mit Default-Werten gefüllt.

Frames zeichnen sich weiterhin durch eine hierarchische Strukturierung aus. An der Spitze der Hierarchie stehen sehr allgemeine Konzepte. Diese spezialisieren sich in einzelne Unterklassen, wobei hier das Prinzip der Vererbung vorherrscht: Objekte vererben alle Eigenschaften an ihre Unterklassen. Diese wiederum können zusätzliche neue Merkmale besitzen. An unterster Stelle der Hierarchie stehen reale Objekte, die sog. Instanzen mit ihren konkreten Merkmalsausprägungen. Das Vererbungsprinzip vermindert Redundanz, allerdings müssen Ausnahmen speziell behandelt werden.

Damit stellen Frames zunächst ein statisches, beschreibendes System dar. Eine prozedurale Komponente erhält diese Repräsentationsform, indem als Slot-Einträge auch angeheftete Prozeduren (attached procedures) möglich sind. Zudem lassen sich Slot-spezifische Inferenzregeln formulieren, die beispielsweise eine Priorität der Abarbeitung festlegen können.

⁶Es muß allerdings nicht unbedingt ein reales Objekt (eine reale Situation) existieren, dem dieser Prototyp entspricht - es kann sich auch um abstrakte Stereotypen handeln.

```

Haus isa Gebaeude
  hat Name: Haus-Name
  hat Stockwerk: (Default: 2)
  hat Grundflaeche
  hat Wohnflaeche
    if-needed: Grundflaeche * Stockwerk
  hat Eigentuemer:
    if-needed: suche Institution/Person mit Eigentum = Haus-Name
  ...

Universitaet isa Institution
  .....
  hat Eigentum: Rektoramt

Rektoramt isa Haus
  hat Stockwerk: 5
  hat Eigentuemer: ?
  ...

```

Im Beispiel wird diese prozedurale Komponente verdeutlicht: ist der Eigentümer eines Gebäudes unbekannt, so kann er dadurch ermittelt werden, daß umgekehrt nach einem Eigentümer gesucht wird, dem das Gebäude gehört.

Semantische Netze

Semantische oder assoziative Netze [Quillian 1968] sind ebenfalls als Modell menschliches Gedächtnis entstanden, wobei hier der Aspekt der Assoziativität im Vordergrund steht. Einzelne Konzepte sind über Relationen, sog. assoziative Beziehungen miteinander verbunden. Wird ein Konzept angesprochen, so werden über die Verbindungen alle benachbarten Konzepte ebenfalls aktiviert. Formal lassen sich solche Netze als gerichtete Graphen darstellen und damit als Netz von Knoten und Kanten veranschaulichen, wobei die Knoten für Konzepte (Objekte, Sachverhalte, Ereignisse) und die Kanten für Relationen stehen. Werden lediglich die Verbindungen angegeben (ev. mit einem numerischen Wert als Gewicht versehen), so stehen die Kanten für die Stärke einer Beziehung. Die Kanten lassen sich allerdings auch als unterschiedliche Typen von assoziativen Beziehungen interpretieren. Letztere bezeichnet man als semantische Netze, während die ersteren eher den konnektionistischen Ansätzen zuzuordnen sind (vgl. Kapitel 5.5). Der grundsätzliche Unterschied besteht allerdings darin, daß die Knoten in Neuronalen Netzen aktiv sind, während sie im Semantischen Netz eine passive Einheit darstellen, die von einer zentralen Steuerung aus angestoßen wird.

Diese Struktur erlaubt nicht nur die Speicherung von Fakten, sondern stellt darüber hinaus die assoziativen Beziehungen zwischen diesen dar und ermöglicht somit den Zugriff auf benachbartes Wissen. Wie Knoten oder Kanten zu benennen sind, ist nicht allgemein festgelegt. Hinter einer gleichaussehenden Darstellung kann sich unterschiedliche Semantik verbergen. Eine Kante kann jede Art von Relation darstellen. Die häufigste ist die *ist-ein* (is-a) Beziehung. Diese verbindet speziellere Begriffe mit allgemeineren und legen dadurch auch eine Vererbungshierarchie fest. Die Relation *hat-Teil* verbindet Objekte mit ihren Teilelementen und steht damit für eine Bestandteilshierarchie.

Der grundlegende Vorteil dieser Repräsentationsform besteht darin, daß sie die semantische Nähe zwischen einzelnen Konzepten widerspiegelt. Dies bewirkt einen schnellen Zugriff auf benachbarte Daten. Ein assoziatives Netz eignet sich zudem für eine parallele Abarbeitung: die Aktivationsausbreitung benötigt kein Warten auf vorherige Ereignisse.

Die rein deklarative Beschreibung eines Semantischen Netzes wird für größere Wissensgebiete schnell übersichtlich. Durch sog. Partitionen kann eine Strukturierung erreicht werden. Eine Partition umfaßt dabei mehrere Knoten im Netz und kann dann als neuer Einzelknoten behandelt werden. Eine weitere Schwäche Semantischer Netze liegt darin, daß manche wichtigen Wissensarten nicht unmittelbar abbildbar sind: Regelhafte Zusammenhänge, eingeschränkte Bedingungen oder unvollständiges Wissen lassen sich nur umständlich und ineffizient darstellen.

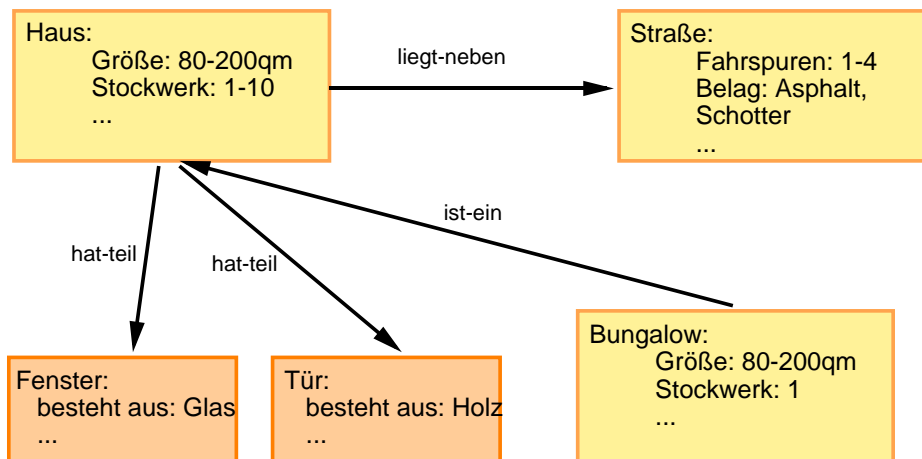


Abbildung 3.4: Beispiel für ein Semantisches Netz

Entitäten–Relationen-Modell

Das Entitäten-Relationen-Modell dient im Datenbank-Bereich primär dem Entwurf und der formalen Beschreibung von Wissen. Es geht über die klassischen Modelle (hierarchisch, Netzwerk, relational) hinaus, indem eine Modellierung von Konzeptklassen und (höherwertigen) Beziehungen zwischen ihnen erfolgt. Hierzu werden verschiedene (z.T. graphische) Beschreibungssprachen eingesetzt (z.B. NIAM (Nijssens Information Analysis Method), EXPRESS mit EXPRESS-G).

3.5.3 Objektorientierte Darstellung

Das objektorientierte Prinzip in der Informatik durchzieht sowohl die Disziplin der Wissensrepräsentation, aber auch die der Programmiersprachen (z.B. Smalltalk [Kay & Goldberg 1977]). So existieren bereits für viele Sprachen objektorientierte Erweiterungen (z.B. C++). Auch im Bereich der Datenbanken gehen neue Forschungen dahin, diesen Ansatz günstig einzusetzen. Objektorientiert bedeutet eine Verschiebung der Sichtweise auf Softwaresysteme: während in herkömmlichen Ansätzen Prozeduren im Mittelpunkt stehen, die Daten verarbeiten, werden hier die Daten mit ihren Eigenschaften und Manipulationsmöglichkeiten beschrieben. D.h. neben Objektstrukturen wird auch das Objektverhalten modelliert. Objekte können somit zwei Dinge: Information über sich selbst speichern und Nachrichten (Messages) empfangen und verarbeiten. Der Informationsspeicher wird Slots oder Attribute genannt, die Nachrichtenverarbeitung geschieht über sog. Methoden. Objekte agieren miteinander, indem sie bestimmte Methoden anwenden, die für jedes Objekt spezifisch definiert sind. Das Prinzip des "Information Hiding" bedeutet, daß Objekte bzw. Methoden unter einem bestimmten Namen angesprochen werden und damit auch eine gewisse Funktionalität haben, die unabhängig ist von der jeweils unterliegenden Implementierung und auch vom speziellen Objekt. Eine Addition z.B. kann durchgeführt werden für Zahlen, Matrizen oder Symbole - für jedes Objekt liegt intern eine andere Definition der Funktion vor; von außen braucht dies allerdings nicht sichtbar zu sein! Eigenschaften und Methoden werden für eine abstrakte Objektklasse formuliert. Individuelle Objekte entstehen durch Instanziierung dieser Klassen mit entsprechenden Werten.

Dieses objektorientierte Prinzip ist am ehesten in den objektzentrierten Ansätzen (Frames, Semantische Netze) verwirklicht. Allerdings ist dort kein Nachrichtenaustausch über Methoden vorgesehen.

3.5.4 Verarbeitung von Wissen

Um aus gegebener Information neue Schlüsse zu ziehen, und damit neues Wissen abzuleiten, wird in der Informatik vornehmlich das Prinzip der Deduktion angewandt. Hiermit ist ein Schluß von einer allgemeinen Tatsache auf eine besondere möglich. Es existieren jedoch noch weitere Formen der Inferenz, welche der Mensch im täglichen Leben ständig nutzt. Zu ihnen zählen insbesondere die Induktion (Schluß von einer Anzahl von Tatsachen auf eine allgemeine) oder die Abduktion, die aus einem gegebenen Zustand Hypothesen auf die ihn auslösende Tatsache oder Aktion aufstellt (z.B.: Straße ist naß, also muß es geregnet haben). Diese Schlüsse basieren meist auf Erfahrungswerten. Induktive und abduktive Schlüsse sind nicht beweisbar. Eine weitere Form der Inferenz

ist das visuelle Schließen, bei dem ausgehend von einer “mentalen Karte” neues Wissen abgeleitet wird (spatial reasoning).

Für die eigentliche Wissensverarbeitung existieren verschiedene Inferenzstrategien. Gängig sind Vorwärts- und Rückwärtsverkettung, sowie das Blackboardprinzip. Die Vorwärtsverkettung geht von den Daten aus, und versucht daraus gewisse Aktionen abzuleiten. Dieser Ansatz wird auch als bottom-up oder datengetrieben bezeichnet. Umgekehrt startet das Verfahren der Rückwärtsverkettung mit dem Aufgabenziel und versucht, die darin enthaltenen Vorbedingungen zu beweisen. Daher wird diese Methode auch modellgesteuert oder top-down genannt. Mit Hilfe eines Blackboards läßt sich eine Strukturierung der Abarbeitung erreichen. Das Wissen ist in unabhängigen Modulen (Dämonen) organisiert. In einem zentralen Datenspeicher (dem Blackboard oder einer Wandtafel) werden alle (Zwischen-) Ergebnisse abgelegt. Die Dämonen werden aktiviert, wenn Informationen auf dem Blackboard auftauchen, die für sie relevant sind. Die Abarbeitung der Strategien der einzelnen Dämonen ist parallel möglich. Die Strategien basieren üblicherweise auf Suchverfahren.

3.5.5 Diskussion der Repräsentationsformen

Zur Repräsentation von Wissen in einem Computer existieren eine Vielzahl von Beschreibungssprachen, die sich für verschiedene Aufgaben unterschiedlich gut eignen. Daher ist je nach Aufgabenziel eine adäquate Darstellung zu wählen, die im Hinblick auf jeweilige Zielvorstellungen wie semantische Korrektheit, Effizienz oder Wiederverwendbarkeit ein Optimum bietet.

Die Ansätze lassen sich grob in zwei Strömungen unterteilen [Winograd 1985]:

Deklarative Darstellung: Wissen bedeutet, zu “wissen daß”, und ist als Menge von Tatsachen abgelegt.

Prozedurale Darstellung: Wissen bedeutet, zu “wissen wie”, und ist als Menge von Regeln abgelegt.

Deklarative Ansätze beschreiben statische Aspekte des Wissens, d.h. Objekte und ihre Relationen, während prozedurale Ansätze das Wissen über seine Nutzung definieren. Damit sind prozedurale Beschreibungen immer aufgabenspezifisch und somit problemabhängig. Diese Einschränkung bedeutet, daß das Wissen gezielt auf eine bestimmte Fragestellung hin angewandt wird. Für die Inferenzstrategie ist dies von Vorteil, da nicht die gesamte Wissensbasis blind nach möglichen Antworten durchsucht werden muß - wie es in deklarativen Systemen der Fall ist. Die problemunabhängige Darstellung deklarativer Beschreibungsformen hingegen hat den Vorteil, daß viel “ungesagt” bleiben kann und nicht explizit beschrieben werden muß: sind die Konzepte und ihre Relationen erst einmal definiert, so können sich die verschiedensten Ableitungen daraus ergeben.

Somit läßt sich feststellen, daß objektspezifisches Wissen sich sehr gut in deklarativen Systemen darstellen läßt, während regelhafte Zusammenhänge besser in prozeduralen Sprachen ausdrückbar sind. Die Vorteile einzelner Systeme zu nutzen, wird daher angestrebt. Dies zeigt sich daran, daß alle Ansätze meist nicht mehr nur in ihrer “reinen Form” existieren, sondern Erweiterungen besitzen, diesen Nachteil auszugleichen. Objektorientierte Ansätze stellen solch hybride Systeme dar. Sie sind primär deklarativ, objektbezogen organisiert, verfügen allerdings über eine prozedurale Komponente in Form der angehefteten Prozeduren.

4 Modellierung

Modellierung kennzeichnet den Prozeß, der implizit gegebenes Wissen in eine explizite Form überführt. Nur in dieser expliziten Form ist das Wissen nutzbar. Es handelt sich bei der Modellbildung um einen interaktiven Vorgang, bei dem das modellierende System ein internes Abbild der es umgebenden Welt erstellt. In einem lebenden Organismus muß dieser Prozeß ständig ablaufen, um sich einer sich ändernden Umwelt anzupassen, aber auch um Änderungen des Wissens um die Welt zu registrieren. Welcher Art die erstellten Modelle sind, d.h. welche Merkmale für das Modell verwendet werden, ergibt sich meist erst im Prozeß der Modellierung [Sommer, Emde, Kietz, Morik & Wrobel 1993]. Daher sind die Vorgänge Modellierung und Repräsentation auch eng miteinander verzahnt: vor der eigentlichen Wissensdarstellung im Rechner muß im Modellierungsschritt entschieden werden, welche Art von Information abgespeichert wird. Dies beinhaltet die Auswahl geeigneter Merkmale zur Beschreibung des Wissens.

4.1 Allgemeine Vorüberlegungen

Die Begriffe Wissenserwerb und Modellierung werden im folgenden als verwandt angesehen. Der Wissenserwerb stellt dabei das Endergebnis der Modellierung dar - eine Tatsache, die uns oft nicht bewußt ist. Üblicherweise wird Wissen als etwas Vollständiges, Abgeschlossenes bzw. Wahres betrachtet, während der Begriff des Modells doch stark abstrahierende, subjektive Komponenten enthält. Wie im Kapitel über Wahrnehmung allerdings gezeigt wurde, gibt es kein Wissen ohne "Schemata", mit Hilfe derer es erst erworben und bewertet werden kann. Wir betrachten die Welt immer aus der Sicht dieser Modelle, und nur was durch diese "Filter" sichtbar ist, kann erkannt werden. Im Folgenden geht es allerdings nicht um den Wissenserwerb im allgemeinen, sondern dieser wird auf konkrete Objektmodelle eingeschränkt.

Beim Menschen läuft die Modellierung der ihn umgebenden Welt üblicherweise unbewußt ab. Erst wenn die Modellierung auf bewußter Ebene durchgeführt wird, wird klar, daß einige grundlegende Fragestellungen zu berücksichtigen sind. Dies ist dann der Fall, wenn etwa eine Bildbeschreibung angefertigt oder eine Person beschrieben wird (Objektmodell), oder wenn es um eine Wegbeschreibung bzw. um die Schilderung eines Arbeitsablaufs geht (Prozeßmodell). Immer ist das Ziel der Modellierung, die Realität in geeigneter Weise zu abstrahieren, um so eine umfassende Beschreibung zu erhalten. So ergeben sich meist aus dem Aufgabenziel - also der angestrebten Nutzung des Modells - Anforderungen an die Qualität und Reichhaltigkeit der Beschreibung. Die Darstellung hängt darüber hinaus von der zugrundeliegenden Sprache, der Repräsentationsform ab⁷. Die Repräsentation umfaßt dabei Merkmale der realen Welt - allerdings werden nie alle möglichen Merkmale genutzt. Welcher Aspekt der Realität in das Modell mit aufgenommen wird, ist das Problem der Modellierung. Sowohl die Wahl der Merkmale, als auch ihre Repräsentation hängt von der zugrundeliegenden Fragestellung und vom Aufgabenziel ab. Reimer [1991] schreibt:

- Eine Repräsentation erfaßt einige Merkmale der repräsentierten Welt.
- Eine Repräsentation erfaßt nicht zwangsläufig (in der Regel nie) alle Merkmale der repräsentierten Welt.
- Nicht alle Merkmale einer Repräsentation stehen zwangsläufig für Merkmale der repräsentierten Welt.

Hiermit begründet sich die Notwendigkeit einer Interpretationsvorschrift, die angibt, wie die Repräsentationsstrukturen im jeweiligen Zusammenhang zu verstehen sind. Er veranschaulicht dies anhand der "Realität", die aus drei ineinandergeschachtelten Kreisen, sowie einem separaten Kreis besteht (Abbildung 4.1, links). Jede Repräsentation betont dabei spezielle Aspekte der Realität: in Version a) steht die Strichlänge für die Größe des Objektes. Im zweiten Fall stehen die Pfeile für die Relation "hat gleiche Farbe", während in Version c) die Pfeile die Relation "liegt innerhalb" beschreiben. Es wird deutlich, daß für verschiedene Sichten auf ein Konzept unterschiedliche Dinge von Bedeutung sind, die mitunter sehr stark differieren können. Als weiteres Beispiel hierfür führt Winograd [1985] das Konzept "Tag" an: diesem Konzept können Attribute wie Wochentag, Monat und Jahr beigegeben werden. Betrachtet man das Konzept jedoch im Sinne eines Ablaufs eines einzelnen Tages, so sind ganz andere Eigenschaften (Aufstehen, Frühstück, ...) wichtig.

⁷Eine Sprache, die das Konzept "Farbe" nicht kennt, kann dieses auch nicht als Beschreibungsmerkmal nutzen.

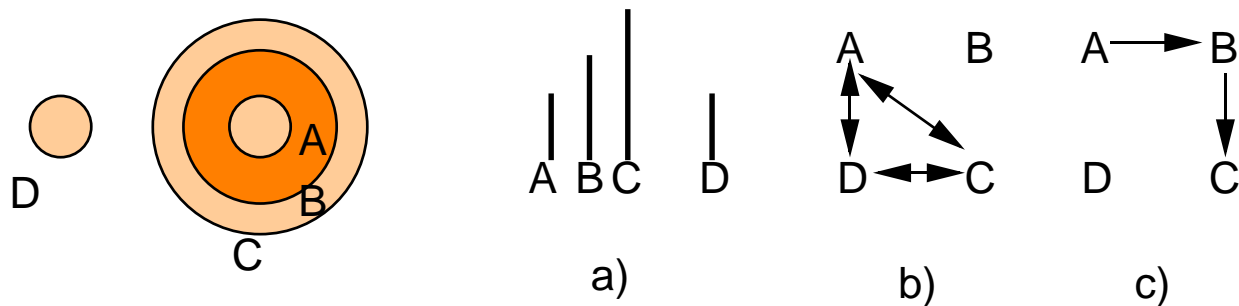


Abbildung 4.1: Beispiel nach [Reimer 1991] für unterschiedliche Repräsentationen (rechts) der Realität (links)

Gleichermaßen können auch Objektbeschreibungen entsprechend der Aufgabenstellung differieren. Steht beispielsweise die Nutzung eines Hammers im Vordergrund, so wird seine Größe und sein Gewicht eine große Rolle spielen. Gilt es jedoch, einen Vorschrift anzugeben, wie ein solcher zu greifen ist, werden andere Eigenschaften wichtig. In der Bildanalyse kann es beispielsweise darum gehen, in einem Bild dunkle Partikel von einem hellen Hintergrund zu trennen. In diesem einfachen Fall erfordert die Beschreibung der Partikel lediglich eine Angabe eines gewissen Grauwertes - alle weiteren Merkmale der Partikel (z.B. Größe, Form, Lage zu anderen Objekte) sind irrelevant. Zu beachten ist hier jedoch, daß sowohl das Aufgabenziel ("Trenne Partikel von Hintergrund"), als auch die Bedingungen der Aufnahme (dunkle Partikel auf hellem Hintergrund) sehr einfach sind und damit zur Modellierung bereits einfache Merkmale genügen. Sind die äußeren Gegebenheiten eindeutig, so hilft dies bei der Festlegung einer Beschreibung: gibt es z.B. nur eine Straße, die von A nach B führt, so genügt die Angabe, "fahre von A nach B" (implizit wird hier ferner noch angenommen "auf der Straße"). Stehen jedoch mehrere Möglichkeiten (und Wege) zur Verfügung, ist das Modell mit detaillierteren Angaben zu verfeinern. Ziel ist es, die für die jeweilige Aufgabenstellung "reichste" Beschreibung zu nutzen. Damit ist eine Beschreibung immer nur im Hinblick auf ein bestimmtes Aufgabenziel vollständig.

4.2 Anforderungen an die Modellierung

Das Ziel einer Modellierung besteht darin, eine möglichst allgemeine Beschreibung eines gegebenen Objekts zu ermitteln. In Anlehnung an die menschliche Vorgehensweise sind daher im Vorfeld der Modellierung folgende Aspekte zu klären:

- Zunächst bestimmt das Aufgabenziel, welche Art von Modell bereitgestellt werden muß: deskriptive Modelle beschreiben Objekte in erschöpfender Weise, mit all ihren Eigenschaften; diskriminatorische Modelle erfassen lediglich die Eigenschaften, die nötig sind, um die Objekte von anderen zu unterscheiden [Niemann 1981].
- Speziell in der Bildverarbeitung ist zu klären, in welcher Dimension die Objekte bereitzustellen sind. Zweidimensionale Objekte sind im Objekt- und Bildraum im Wesentlichen identisch. Dies gilt nicht für 3D-Objekte. Hier sind entweder die verschiedenen Ansichten des Objekts in den jeweiligen Projektionen zu modellieren, oder es muß eine komplette 3D-Beschreibung vorgegeben werden.
- Von elementarer Bedeutung ist die Festlegung der Repräsentation, in der die Beschreibung erfolgt. Vielfach sind Repräsentationen bereits vorhanden, nicht aber in der geeigneten Sprache, die für das Verständnis oder die weitere Verwendung sinnvoll und nötig ist. Eine Beschreibung in natürlicher Sprache ist zwar sehr reichhaltig, allerdings für die maschinelle Verwertung zu komplex. Darüber hinaus enthält sie sehr viel implizites Wissen über die Welt, welches (derzeitigen) maschinellen Systemen nicht zur Verfügung steht. Aber auch scheinbar "computergerechte" Repräsentationsformen, die gegenüber der natürlich-sprachlichen Darstellung eine starke Abstraktion erfahren haben, eignen sich nicht unmittelbar, so z.B. die CAD-Modelle des Konstruktionsbereichs. Auch für den Menschen sind solche Darstellungen erst mit einem Quantum an Vorinformation zu verarbeiten. Gleichermaßen muß die CAD-Repräsentation von einem anderen System erst einmal "verstanden" werden.
- Nur solche Merkmale des zu beschreibenden Objekts können genutzt werden, die beobachtbar bzw. oder bekannt sind (Farbe, Länge, Größe, ...). Wichtig ist auch, wie robust diese Merkmale sind, d.h. wie sicher sie extrahiert werden können.

- Für die spätere Nutzung des Modells kann es wichtig sein, eine Prioritätenliste der Merkmale zur Verfügung zu haben: Seltene Merkmale sollten zuerst verwendet werden. Somit kann ein schneller Zugriff auf die gesuchten Objekte erfolgen.
- Neben den Objektattributen (strukturelle Eigenschaften) sind auch noch ihre Relationen zu anderen Objekten (kontextuelle Eigenschaften) zu beschreiben. Viele Objekte lassen sich erst oder gerade durch ihre Relationen eindeutig klassifizieren.
- Ein Modell basiert nicht nur auf Attributen oder Eigenschaften, sondern es enthält ferner noch Methoden, oder Funktionen, die die Extraktion dieser Merkmale erlauben oder mögliche Verbindungen zu anderen Eigenschaften angeben. Solche Methoden sind zum Teil grundsätzlicher Art, und als problemunabhängiges Systemwissen abspeicherbar (um Objekt herumgehen, Kopf bewegen, um Verdeckungen zu umgehen), aber auch objektklassenspezifisch (wie z.B. Objekt berühren, klopfen: "ist es Glas ?") und objektspezifisch (beißen: "ist es Gold ?").
- Generische Modelle lassen sich sowohl für die Modellierung, als auch für die anschließende Zuordnung sinnvoll einsetzen. Diese erfassen alle gemeinsamen Eigenschaften einer Klasse von Objekten, und können somit direkt für die Objektdetektion genutzt werden - ohne alle Individualobjekte kennen und beschreiben zu müssen.

4.3 Möglichkeiten der Modellgenerierung

Prinzipiell existieren folgende Möglichkeiten zur Ermittlung der Modelle für die Bildanalyse:

Manuelle Modellkonstruktion: Vorab werden die charakteristischen Eigenschaften des zu modellierenden Objekts herausgestellt. Auch kontextuelle Merkmale des Objekts werden so gefaßt und beschrieben. Wenn möglich und bekannt, werden auch solche Merkmale erfaßt, nach denen im Interpretationsprozeß zunächst zu suchen ist, um so die Analyse zu beschleunigen. Diese Überlegungen werden meist ad hoc durchgeführt. Oft ist eine Kenntnis der wichtigen Merkmale nicht vorhanden. Es werden diejenigen übernommen, die für den Menschen wichtig erscheinen. Die gesamte Modellierung ist sehr zeitintensiv. Sie erfordert detaillierte Kenntnisse sowohl über das Objekt, aber auch über das Objekterkennungssystem. Solch eine Art der Modellierung wird für Domänen mit vielen, verschiedenen, sich ändernden Objekten schnell unhandlich.

Modellierung durch Adaption vorhandener Modelle: In vielen Bereichen, insbesondere in der industriellen Konstruktion, existieren bereits eine Vielzahl von Objektmodellen. Diese Modelle wurden allerdings für einen anderen Zweck, d.h. mit einer anderen Zielsetzung entwickelt, und sind daher nicht unmittelbar für die Bildanalyse nutzbar. So fehlen etwa den Modellen des CAD-oder GIS-Bereichs explizite Angaben bezüglich der Sichtbarkeit bestimmter Objektteile. Auch Relationen zwischen einzelnen Objektteilen, z.B. der Nachbarschaft zweier Regionen, sind meist nicht vorhanden. Diese Merkmale sind jedoch für die Bildanalyse von elementarer Bedeutung und müssen daher beim Übergang von CAD- nach BV-Repräsentation bestimmt werden [Flynn & Jain 1991], [Ikeuchi & Kanade 1988], [Hansen & Henderson 1989]. Aus vorhandenen Systemen können in erster Linie spezifische Modelle entnommen werden, die individuelle Objektinstanzen beschreiben. Die engere Verbindung von GIS und Bildanalyse wird verschiedentlich vorgeschlagen (vgl. z.B. [Fritsch 1993]). Plietker [1994] stellt ein Verfahren vor, bei dem die spezifischen GIS-Modell zur Erkennung von Straßen in Bildern eingesetzt wird. Die Nutzung allgemeiner Modelle hingegen - wie sie beispielsweise im Modellierungsschritt eines GIS bereitgestellt werden müssen - ist derzeit noch Gegenstand der Forschung. Sie eröffnet allerdings ein großes Potential an Möglichkeiten.

Automatisches Lernen von Modellen aus Beispielen: Oft sind Beispiele für die gesuchten Objekte vorhanden - allerdings nicht in geeigneter Repräsentation. Ein automatisches Lernsystem ermittelt zunächst die wichtigen Merkmale und sucht, die innere Struktur der Beispiele aufzudecken. Der Vorteil dieser Methode besteht darin, daß sich das Modell rein aus beobachtbaren Merkmalen, den Merkmalen der Beispielobjekte, zusammensetzt, die von weiteren Verfahren übernommen werden können. Somit ist dieses Verfahren objektiv. Allerdings hängt hier die Qualität der abgeleiteten Modelle direkt von der Qualität der Vorlage ab. Auch die Art des Sensors (d.h. seine "Wahrnehmungsfähigkeit") bildet sich unmittelbar auf das Modell ab. Das Lernsystem übernimmt die Aufgabe der Bestimmung der relevanten Merkmale und ihrer Zusammenhänge. Das bedeutet, daß Objektmodelle aufgabenspezifisch ermittelt werden können. Die

abgeleitete Beschreibung ist zudem generischer Natur: sie gilt für alle Beispiele, aus denen sie ermittelt wurde, allerdings auch noch für alle weiteren Objekte dieser Klasse.

Die manuelle Erstellung von Modellen ist nur für klar abgegrenzte Gebiete möglich. Innerhalb dieser lassen sie sich anwenden. Allerdings schlagen sie fehl, wenn eine der Annahmen nicht mehr gegeben ist - ein Umstand, der auch mit "brittleness" bezeichnet wird. Nachteilig ist zudem, daß es unmöglich ist, Modelle für alle möglichen komplexen Probleme des Bildverstehens aufzustellen [Weng, Ahuja & Huang 1993]. Daher bietet sich die Nutzung automatischer Verfahren für eine flexible Modellierung an. Die Forderung ist dabei, daß die abgeleiteten Modelle für den Menschen verständlich sind. Das Ideal ist eine erweiterbare Wissensbasis, die inkrementell neue Objekte aufnimmt. Hiermit wird es ermöglicht, auf Veränderungen und Erweiterungen (der Umwelt) zu reagieren.

5 Maschinelles Lernen

Das Fachgebiet Maschinelles Lernen ist eine Teildisziplin der Künstlichen Intelligenz (KI), und beschäftigt sich damit, Lerntechniken in einem Computer zu realisieren⁸. Lernen ist ein Vorgang, der üblicherweise nur Lebewesen zugesprochen wird, was eine Nutzung dieses Begriffs im technischen Bereich problematisch macht. Ähnliche Assoziationen wie beim Begriff 'Künstliche Intelligenz' werden hier wachgerufen. Eine allgemeine Definition hilft, diesen Begriff auch im maschinellen Bereich wert- bzw. emotionsfrei nutzen zu können.

Lernen bewirkt Änderungen in einem System, welche dazu führen, daß gleiche Aufgaben besser oder effizienter bewältigt werden können [Simon 1984].

Solche Änderungen beziehen sich immer auf das dem System zugrundeliegende Wissen. Daher ist Lernen auch immer an das Problem der Wissensrepräsentation und -verarbeitung geknüpft. Obige Definition läßt sich leicht an vertrautem Lernverhalten verifizieren: beim Lernen aus einem Buch wird die gelesene Information verinnerlicht, d.h. in eine interne Repräsentation überführt. Nur in dieser Form ist sie als Wissen zu bezeichnen und nutzbar. Damit wurde das Wissen des Systems "Mensch" erweitert. Das Gelernte kann künftig genutzt werden. Dieser Lernschritt wird allerdings immer subjektiv vollzogen, da das Wissen in unterschiedlicher Form abgespeichert werden kann. So gibt es verschiedene Methoden, um sich Zahlen, beispielsweise Telefonnummern zu merken: als Einzelzahlen, als 3-er Blöcke, über Eselsbrücken wie Geburtsdatum, oder visuell durch die Lage auf dem Tastenfeld (z.B. horizontal:147 oder diagonal: 753). Durch Lernen können neue Verhaltensmuster oder neues Wissen erworben werden. Lernen geht also immer von bestimmten Gegebenheiten aus, die im Hinblick auf ein bestimmtes Aufgabenziel modifiziert oder erweitert werden.

Lernen läßt sich als Zusammenspiel zwischen einem Schüler und einem Lehrer charakterisieren. Der Schüler sucht sein Wissen zu erweitern oder Neues zu lernen, wobei ihn der Lehrer unterstützt. Das zu lernende ist in der Umgebung des Schülers implizit gegeben, meist in Form von Beispielen. Je nach Grad der Interaktion durch den Lehrer können verschiedene Lerntechniken unterschieden werden (vgl. auch Abbildung 5.2 Seite 41): Lernen durch direkte Imitation bzw. durch auswendig lernen, Lernen über Analogie, Lernen durch Schlußfolgerungen. Diese Techniken finden ihr Analogon auch im maschinellen Bereich.

Maschinelles Lernen läßt sich somit charakterisieren als

ein Prozeß, bei dem ein Computer sein Wissen und seine Fähigkeiten vergrößert [Kodratoff & Tecuci 1988].

In der KI beschäftigen sich zwei Forschungsgebiete mit Lerntechniken: der Konnektionismus (Neuronale Netze) und das Maschinelle Lernen. Beide Ansätze behandeln das gleiche Grundproblem, sie gehen daher auch von der gleichen Ausgangssituation aus: die gegebene Information soll in eine neue Form überführt werden, welche im Hinblick auf ein bestimmtes Aufgabenziel günstiger ist. Der elementare Unterschied zwischen beiden Forschungsrichtungen besteht in der Repräsentationsform der bereitgestellten Daten - und damit auch in den nutzbaren Lernverfahren. Bei neuronalen Netzen wird die interne Struktur des Wissens als "Black Box" angesehen, während strukturelle Beschreibungen versuchen, diese offenzulegen. Neuronale Netze sind numerische Verfahren, bei denen ein gewünschtes Ergebnis einer Anfangssituation gegenübergestellt wird. Beide sind in Form von Zahlen repräsentiert. Das Ergebnis, welches die Daten ineinander überführt, ist ein Satz von Koeffizienten oder Parametern. Strukturelle Ansätze stellen das Wissen für den Anwender transparent dar. Dazu werden höhere Beschreibungssprachen verwendet. Insbesondere wird das im Lernschritt erworbene Wissen in einer Form repräsentiert, die für den Menschen verständlich ist - etwa in Form von Regeln.

Im folgenden wird das Grundprinzip maschineller Lernverfahren dargestellt, sowie Lernmethoden vorgestellt, soweit sie für die vorliegende Arbeit relevant sind. Dabei erfolgt eine Konzentration auf die Verfahren des induktiven Lernens.

5.1 Grundprinzip Maschineller Lernverfahren

Menschliches und auch Maschinelles Lernen erfolgt nicht im 'luftleeren Raum'. Voraussetzung sind immer ein Lernziel, sowie Information, aus der gelernt werden soll. Im Falle des Maschinellen Lernens ist diese meist in

⁸Damit unterscheidet es sich von Tutorsystemen, welche den Computer als Lehrmedium einsetzen.

Form von Beispielen gegeben. Diese stellen eine sogenannten extensionale Beschreibung für den zu lernenden Sachverhalt dar. Im Lernschritt wird dieser explizit gemacht und damit zu einer intensionalen Beschreibung [Reimer 1991].

Lernverfahren beruhen auf dem Prinzip der Induktion. Dieses bezeichnet den Schluß aus einer begrenzten Zahl von Fällen einer Gattung auf alle Fälle, bzw. das Erschließen von allgemeingültigen Sätzen aus Einzeltatsachen. Im Gegensatz dazu werden bei der Deduktion speziellere Kenntnisse aus allgemeineren abgeleitet. Deduktiv abgeleitete Aussagen können eindeutig als wahr klassifiziert werden, wenn sie sich aus gegebenen Axiomen ableiten lassen. Solch eine Falsifizierbarkeit ist für induzierte Schlußfolgerungen nicht möglich. Die abgeleiteten Aussagen gelten streng genommen nur für die Grundgesamtheit aller Fälle, aus denen sie geschlossen wurden. Es ist daher immer möglich, daß ein neuer Fall auftritt, für den diese Aussage nicht gilt. Nur eine endliche Menge von Beispielen kann getestet werden, potentiell gibt es aber unendlich viele.

Es lassen sich zwei Arten von Induktiven Lernverfahren unterscheiden: Lernen aus Beispielen und Lernen durch Beobachtung (Conceptual Clustering). Im Falle des Lernens aus Beispielen werden dem System Beispiele vorgegeben, deren Klassenzugehörigkeit jeweils spezifiziert ist. Die Aufgabe besteht darin, aus den vorgegebenen Beispielen ein sog. Konzept für die Klassen zu bestimmen. Aus den gegebenen Fakten sollen plausible Regeln abgeleitet werden, die zum einen die Fakten erklären, aber auch neue Fakten vorhersagen können. Damit ist zweierlei erreicht: eine kompakte Beschreibung der Beispiele und Regeln zur Klassifizierung unbekannter Fakten.

Lernen durch Beobachtung (auch unüberwachtes Lernen genannt) zeichnet sich dadurch aus, daß hier zwar auch Beispiele vorgegeben werden, allerdings nicht deren Klassenzugehörigkeit. Die Aufgabe ist damit zweigeteilt (vgl. [Kodratoff & Tecuci 1988]): zunächst gilt es, das Aggregationsproblem zu lösen, indem die Beispielmenge in zusammenhängende Gruppen eingeteilt wird. Anschließend sind diese Gruppen zu charakterisieren. Für diesen zweiten Schritt können Verfahren des Lernens aus Beispielen benutzt werden.

Das Problem des Lernens besteht darin, aus den gegebenen Fakten neues Wissen abzuleiten. Prinzipiell können aus den Fakten beliebig viele Hypothesen erzeugt werden, weshalb Lernen auch als Suchproblem verstanden werden kann: die Eigenschaften der Beispiele müssen in geeigneter Weise verknüpft werden, um zu einem gewünschten Zielzustand zu kommen. Die Kombinationsmöglichkeiten sind dabei vielfältig. Prinzipiell kann jede Eigenschaft mit jeder anderen verknüpft werden, was zu einer sehr großen Anzahl an Hypothesen führen kann. Um die Suche zu beschränken, benötigt das System zusätzlich Hintergrundwissen [Nilsson 1982]. Im Hintergrundwissen wird der Wertebereich der Attribute, sowie mögliche Verknüpfungen der Attribute vorgegeben (Konjunktion, Disjunktion, arithmetische Funktionen, ...). Es können aber auch weitere Fakten über den aktuellen Problembereich angegeben sein.

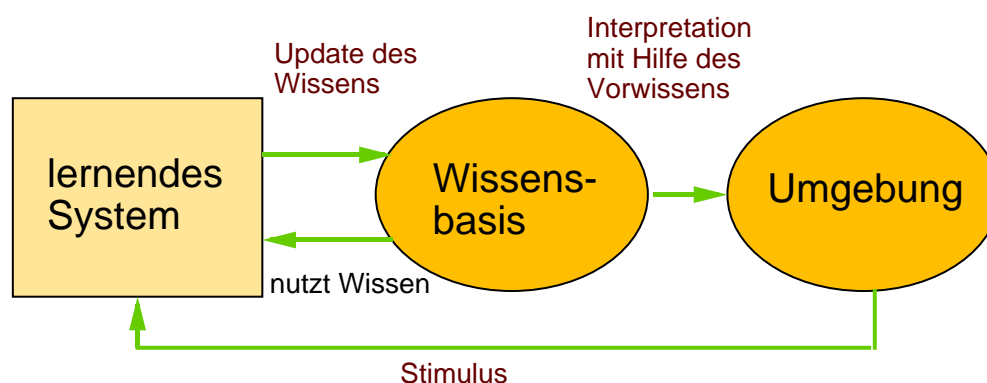


Abbildung 5.1: Voraussetzungen für Lernverfahren

Lernverfahren benötigen somit (siehe auch Abbildung 5.1):

- Beispieldaten (Umgebung),
- Hintergrundwissen (Wissensbasis),
- ein lernendes System,
- einen Stimulus, der den Lernprozeß anstößt (z.B. ein Lehrer, eine neue Situation, die Umgebung).

Folgendes Beispiel soll das Problem verdeutlichen (vgl. [Forsyth & Rada 1986]):

Aus einer Sammlung von Wettercharakteristika soll eine Vorhersage des Wetters des folgenden Tages durchgeführt werden. Eine Datenbank enthält eine Statistik über das Wetter eines Tages (z.B. Sonnenscheindauer, Windstärke, ...), sowie das Wetter des folgenden Tages.

Date	Rainfall	Sunshine	Windmax	Pressure	Nextday
17-April-85	0	10.6	15	1030	Fine
12-April-85	0.5	7.6	44	1008	Not-Fine
...
20-April-85	0.7	7.1	33	1000	Not-Fine

Das Lernziel besteht darin, Regeln zu finden, die aufgrund des aktuellen Wetters das des folgenden Tags prognostizieren. Grundannahme ist hierbei, daß das Wetter des Vortags in irgendeiner - unbekanntem - Weise mit dem aktuellen Wetter in Zusammenhang steht.

Im Lernschritt werden Regeln gesucht, mit denen alle Datensätze richtig zugeordnet werden können. In obigem Beispiel würde es genügen, zu spezifizieren, daß

$$\text{Rainfall} < 0.5$$

sein muß, um für den folgenden Tag schönes Wetter vorherzusagen. In realistischen Datensätzen reicht üblicherweise solch eine einfache Abfragerregel nach einem einzigen Kriterium nicht aus. Welche Art von Regeln erzeugt werden, sowie, wie die Eigenschaften der Beispiele zu neuen Regeln verknüpft werden können, ist durch das Hintergrundwissen des Systems vorgegeben.

Zur Erzeugung neuer Hypothesen können vor allem folgende Methoden verwendet werden:

Spezialisierung: eine bisherige allgemeine Beschreibung wird ausgehend von einem negativen Beispiel derart spezialisiert, daß sie das negative Beispiel nicht mehr enthält. Spezialisierung einer Regel läßt sich durch Hinzufügen von Prämissen erreichen. Eine spezialisierte Regel trifft nur noch für weniger Fälle zu.

Generalisierung: eine bisherige spezielle Beschreibung wird ausgehend von einem positiven Beispiel derart verallgemeinert, daß sie das positive Beispiel mit einschließt. Generalisierung wird z.B. durch Streichen von Prämissen in Regeln (Dropping Condition) erreicht⁹. Eine generalisierte Regel trifft anschließend für mehr Fälle zu.

Hier wird auch die Bedeutung positiver und negativer Beispiele deutlich. Negative Beispiele dienen dazu, die möglichen Konzepte einzuschränken, während positive diese erweitern. Liegen lediglich positive Beispiele vor, so läßt sich ein Konzept beispielsweise durch einfache Konjunktion der Merkmale aller Beispiele ermitteln.

Zur Verifikation des Ergebnisses wird üblicherweise der Gesamtdatensatz vor der Bearbeitung in Trainingsdaten und Testdaten aufgeteilt. Mit ersteren lernt das System, während letztere zum Testen der gelernten Konzepte genutzt werden. Als Qualitätsmaß für das Ergebnis läßt sich der Anteil richtiger Klassifizierungen der Testdaten angeben. Allgemein gibt es allerdings keine Definition dafür, was eine "gute" Beschreibung ist. Üblicherweise werden Kriterien wie Kürze oder Einfachheit benutzt [Kodratoff 1988].

Da eine Diskriminierung der einzelnen Klassenbeschreibungen rein aufgrund der Merkmale durchgeführt wird, ist von elementarer Bedeutung, daß die Beschreibung der Beispiele mittels der Merkmale reichhaltig genug ist. Weiter muß vorausgesetzt werden, daß keine widersprüchlichen Beispiele vorkommen (d.h. identische Merkmale, aber unterschiedliche Klasse). Die Qualität einer gelernten Beschreibung ist damit unmittelbar von der Güte und Korrektheit der Beispiele abhängig.

5.2 Klassifizierung der Verfahren

Lernverfahren lassen sich nach verschiedenen Kriterien klassifizieren [Michalski, Carbonell & Mitchell 1984]. So können diese beispielsweise nach der verwendeten Lernstrategie unterschieden werden, welche sich durch den Grad der Unterstützung durch den Lehrer charakterisieren läßt (vgl. Abbildung 5.2), aber auch nach der zugrundeliegenden Wissensrepräsentation.

⁹Weitere Generalisierungsmethoden sind u.a.: Wandeln von Konstanten zu Variablen, interne Disjunktion [Niemann 1981]

5.2.1 Klassifizierung nach dem Grad der Unterstützung durch den Lehrer

Auswendig lernen oder direktes Programmieren: Dies ist die einfachste Form des Lernens. Dennoch zählt es zu den Lernverfahren, da hier eine Änderung des Systemwissens erfolgt, was zu einer effizienteren Problemlösung führt. Allerdings besteht die Aufgabe des Schülers hier lediglich darin, gespeichertes Wissen abzurufen. Es ist bereits so aufgearbeitet, daß es unmittelbar genutzt werden kann, ohne Schlüsse ziehen zu müssen. Samuel's Schachprogramm [Samuel 1967] ist ein frühes Beispiel für ein solches Lernprogramm. Die Schachstellungen werden zusammen mit ihren jeweiligen Entscheidungen in Indextabellen abgespeichert. Hierdurch ist ein schneller und effizienter Zugriff möglich. Neue Züge können auf die gespeicherten Züge zurückgreifen, und wiederum neue Konstellationen bilden. Hiermit ist es dem System möglich, aus Erfahrung zu lernen.

Lernen durch Anweisungen: Das Problem besteht darin, die Sprache des Lehrers in eine interne Sprache zu übertragen, um so die Information nutzen zu können. Die Hauptlast liegt hier beim Lehrer, der das Wissen so aufbereiten muß, daß es mit den Mitteln des Schülers verstanden werden kann, und so sein Wissen erweitert.

Lernen durch Analogie: Von einer ähnlichen Situation muß ein sinnvolles Konzept für die aktuelle Situation gefunden werden. Eine neue Strömung innerhalb des Maschinellen Lernens beschäftigt sich mit dem sog. Case-Based-Reasoning [Aamodt & Plaza 1994]: neue Probleme werden gelöst, indem auf ähnliche vergangene Fälle (Problemsituationen) zurückgegriffen wird. Da der neue Fall anschließend auch abgespeichert wird, kann die Falldatenbank sukzessive erweitert werden.

Lernen aus Beispielen: Ein Satz von Beispielen und Gegenbeispielen für ein Konzept liegt vor. Die inneren Zusammenhänge dazwischen sollen ermittelt werden. Hier muß der Schüler nach sinnvollen Beschreibungen suchen, um die Beispiele klassifizieren zu können. Diese Verfahren lassen sich danach unterscheiden, ob sie inkrementell arbeiten, oder ob alle Beispiele zur Lösung bereitgestellt werden müssen. Inkrementell arbeitende Verfahren leiten bereits aus einem Teil der Beispiele eine Beschreibung ab, die - kommen weitere Beispiel hinzu - schrittweise verfeinert werden können.

Lernen durch Beobachtung (unüberwachtes Lernen): Das lernende System versucht, in einem Datensatz Regelmäßigkeiten oder Gesetze zu finden (Lernen durch Entdecken). Hierbei müssen als Hintergrundwissen allgemeine Prinzipien (wie z.B. Ähnlichkeitsmaße) zur Verfügung stehen.

Lernen durch Mutation: Genetische Algorithmen versuchen - gemäß dem in der Evolution vorherrschenden Prinzip der Mutation und Selektion - aus gegebenem Wissen neue Hypothesen, und damit neues Wissen zu erzeugen (für eine Übersicht siehe [Forsyth & Rada 1986])



Abbildung 5.2: Diametrales Verhältnis zwischen Aufwand des Lehrers und des Schülers

5.2.2 Klassifizierung nach der Wissensrepräsentation

Ein wichtiger Aspekt betrifft die Repräsentation der vorgegebenen Information. Diese liegt in Form von beobachtbaren Charakteristiken (auch Deskriptoren, Attribute oder Eigenschaften genannt) des jeweiligen Problembereichs vor. Sowohl die Art der Repräsentationsform als auch die der Deskriptoren der Beispieldaten sind entscheidend für die Relevanz der abgeleiteten Regeln: soll beispielsweise aus einer Schachkonstellation der nächste Zug abgeleitet werden, so ist der Einfluß der Farbe einer Schachfigur sicher nebensächlich¹⁰. Bevor ein lernendes System entworfen wird, muß sicher sein, daß die gewählte Beschreibungssprache die Unterscheidungen, die man treffen will, auch darstellen kann. Gelernt werden kann nur, was symbolisch ausgedrückt werden kann (Wittgenstein: "The limits of my language are the limits of my world"). Dies betrifft sowohl die Repräsentation der gegebenen Beispiele, als auch die Repräsentation, in der das Ergebnis schließlich präsentiert wird.

Als Merkmale lassen sich - je nach Problembereich und Fragestellung - stetige oder kontinuierliche Werte wählen. Im Bereich der stetigen Deskriptoren können nominale Deskriptoren (z.B. Namen von Personen), lineare Deskriptoren (geben geordnete Reihenfolge an, z.B. militärischer Rang) und strukturierte Deskriptoren unterschieden werden.

Im Konnektionismus bestehen Beispiel- und Zieldaten aus Zahlen, auch die erlernten Regeln bestehen aus einem Netz von Zahlen (Gewichten). Im Bereich des Maschinellen Lernens sind Repräsentationen in Form von Attribut-Wert-Listen gängig. Prinzipiell lassen sich jedoch alle Darstellungsformate der KI nutzen: Formale Grammatiken, Produktionsregeln, logische Ausdrücke, Graphen, Semantische Netze, Frames, u.a. Es gilt jedoch, daß eine zunehmende Komplexität der Beschreibungssprache auf der einen Seite zwar eine Erhöhung der Ausdrucksfähigkeit mit sich bringt, auf der anderen Seite allerdings die Komplexität der Lernalgorithmen erhöht. Das relevante Wissen sollte daher auf möglichst einfacher Art dargestellt werden. Dies gewährleistet auch die Einfachheit der nötigen Inferenzen: ein Vergleich zweier Merkmalslisten ist einfach und schnell durchzuführen, während dies für strukturierte Repräsentationen nur in einem Suchprozeß zu lösen ist.

5.3 Anwendungsgebiete Maschineller Lernverfahren

Die Idee, ein automatisches System zu besitzen, welches Wissen erwerben kann, ist sehr reizvoll. Computer als enorme Datenspeicher werden seit langem vielfältig eingesetzt und genutzt. Gerade aber die Bereitstellung dieser Daten stellt vielfach ein Problem dar. Wissensbasierte Systeme, Expertensysteme aber auch Geoinformationssysteme benötigen gewaltige Datenmengen, die in sinnvoller Weise zunächst dem Computer eingegeben werden müssen. Dies wird insbesondere dann zum Problem, wenn es sich nicht lediglich um bloße Daten handelt, sondern es eher um Wissen, also um komplexere Zusammenhänge geht. Gerade die Forschung auf dem Gebiet der Expertensysteme hat gezeigt, daß zuverlässige Aussagen nur auf der Basis eines großen Quantums an 'Expertenwissen' möglich sind. Dieses Wissen des Experten zu sammeln und in computergerechte Form zu wandeln ist Aufgabe des sogenannten Wissensingenieurs (Knowledge Engineer). Er versucht, über Befragungen, Beobachtungen oder auch Checklisten, dieses Wissen zu ermitteln [Fensel 1991]. Das grundlegende Problem hierbei ist oft, daß der Experte bei seinen Problemlösungen meist nicht nur nach festen Regeln vorgeht (deren Umsetzung relativ einfach wäre), sondern viele Faustregeln oder Heuristiken verwendet, bei denen die Gegebenheiten und Schlußfolgerungen nicht immer eindeutig sind. Zudem weiß er vieles, was er unbewußt für die Entscheidung verwendet. Zu diesem Wissen zählt, was unter gesunder Menschenverstand oder auch Erfahrung fällt.

Die Nutzung wissensbasierter System ist auch in starkem Maße davon abhängig, wie das Wissen verwaltet wird, sowie, wie der Zugriff hierauf erfolgen kann. Von enormer Bedeutung ist, ob das System Wissen über sich selbst hat. Unter *brittleness* (vgl. [Michalski, Carbonell & Mitchell 1986]) versteht man die Tatsache, daß ein System nur in einem bestimmten Bereich operieren kann, außerhalb dagegen handlungsunfähig ist. Angestrebt wird daher im Gegenteil 'graceful degradation', was bedeutet, daß das System im Randbereich seines Wissens nicht abbricht, sondern ein schwächeres Ergebnis liefert bzw. ein mit einer größeren Unsicherheit behaftetes. In vielen Bereichen unterliegt das Wissen einem ständigen Wandel, so daß es nicht ausreicht, es zu einem Zeitpunkt fix einzugeben; vielmehr ist Bedarf an Flexibilität, nach Anpassung und Erweiterung.

¹⁰Hier liegt auch ein Problem beim Design eines lernenden Systems: die Merkmale müssen in gewisser Relation zum zu beschreibenden Sachverhalt stehen. Ansonsten kann es - zufällig - zu spektakulären Zusammenhängen kommen, wie die berühmte statistische Korrelation zwischen Rückgang der Störche und der Geburtenzahl in Deutschland.

Wie in der Einleitung bereits dargelegt, wird im Bereich der Bildanalyse für komplexere Aufgabenstellungen sehr viel Zusatzwissen benötigt. Dieses schließt sowohl Wissen über die vorkommenden Objekte im Bild mit ein, aber auch Wissen über die Aufnahmesituation, und über Strategien zur automatischen Bildinterpretation. So stellt sich auch hier die Frage nach der Bereitstellung dieses Wissens. Die Techniken des Maschinellen Lernens können hier derart genutzt werden, daß der Experte Beispiele, sowie mögliche Schlußfolgerungen daraus bzw. Klassifizierungen dieser Beispiele präsentiert, ohne jedoch angeben zu müssen, wie er zu diesen kommt¹¹. Idealerweise findet ein automatisches System eine "passendere" Objektbeschreibung als der Mensch, dessen Beschreibung möglicherweise subjektiv geprägt sind.

Die größten Hoffnungen bezüglich des Maschinellen Lernens liegen daher im Bereich der automatischen Wissensakquisition - die bislang immer noch einen "Flaschenhals" [Michalski 1984] bei der Automatisierung darstellt. Weitere Anwendungsgebiete (vgl. [Michalski et al. 1984], [Sleeman 1994]) für Lernverfahren sind:

Automatisches Programmieren: Aus gegebenen Eingangs- und Ausgangsdaten sollen Regeln bzw. Formeln gefunden werden, die diese in konsistenter Weise verknüpfen.

Datenkompression: Diese entsteht quasi als "Abfallprodukt" der Anwendung eines Lernverfahrens, da das Ziel des Lernens meist darin besteht, eine gegebene Repräsentation in günstigerer Art darzustellen. Regeln, die einen Sachverhalt beschreiben, sind kompakter als eine Sammlung von Beispielen, die diesen indirekt belegen.

Lernen als Optimierungsproblem: Das Ziel ist, aus gegebenen Parametern diejenigen auszuwählen, für die das Wissen in optimaler Weise dargestellt wird.

Theorie-Bildung: Aus einer Sammlung von Beispieldaten können allgemeingültige Gesetze abgeleitet werden. Das Programm Bacon [Langley 1977] beispielsweise hat das 3. Kepler'sche Gesetz nach-'erfunden'.

Behandlung großer Wissensbasen: Lernen wird auch im Bereich des Wissenserwerbs, der Wissensspeicherung, sowie der Überarbeitung von großen Wissensbasen eingesetzt. Das ehrgeizige Projekt *Cyc*, welches darauf abzielt, eine große Sammlung an Allgemeinwissen anzulegen, gewinnt dies beispielsweise u.a. mittels Lernen aus Texten (z.B. aus der Zeitung) [Lenat 1990]. Das Problem der Überarbeitung wird bei großen Wissensbasen umso wichtiger: je größer sie sind, desto wahrscheinlicher sind Fehler, Inkonsistenzen oder Redundanz. Diese können durch Lernsysteme aufgedeckt werden.

Knowledge Discovery: Ableiten von Wissen aus Datenbanken, welches explizit so nicht dort abgelegt ist.

Entdecken von Struktur in ungeordneter Sammlung von Beobachtungen: Dies kann für verschiedene Wissensgebiete sinnvoll eingesetzt werden, z.B.: Biologie, Genetik, Medizin, Physik, Chemie, ...

5.4 Strukturelle Lernverfahren

Strukturelle Lernverfahren lassen sich in zwei Klassen einteilen [Quinlan 1990]:

Divide and Conquer-Methode: Ausgangspunkt ist eine sehr allgemeine Beschreibung, die alle Beispieldaten umfaßt. Diese wird nach und nach spezialisiert, indem sie so lange in einzelne Mengen unterteilt wird, bis jede Untermenge nur noch Beispiele einer Klasse enthält. Die Unterteilung erfolgt Anhand von bestimmten diskriminierenden Merkmalen. Diese Verfahren arbeiten auf der Basis des Prinzips der Spezialisierung: eine allgemeine Ausgangsbeschreibung wird nach und nach spezialisiert, bis sie die Trainingsbeispiele korrekt beschreibt (z.B. ID3).

Cover-Methode: Im Gegensatz dazu startet diese Methode mit einer speziellen Beschreibung. Diese wird nach und nach generalisiert, bis sie alle positiven Beispiele erfaßt und gleichzeitig keine negativen mehr abdeckt. Ausgangspunkt sind eine vorzugebende Anzahl an Beispielen, die jeweils eine spezielle Konzeptbeschreibung darstellen. Für jedes Beispiel wird eine minimale Disjunktion an Bedingungen gesucht, die das Konzept beschreibt. Alle Beispiele, die durch diese Bedingung abgedeckt (covered) sind, werden entfernt. Für die noch übrigen werden nach dem gleichen Verfahren Bedingungen gesucht, die anschließend über Disjunktion mit der bereits bestehenden verknüpft werden (Beispiele Relax, Cluster, Arch).

¹¹Dies ist natürlich nur für solche Bereiche oder Fragestellungen sinnvoll und nötig, wo eine Definition nicht unmittelbar bekannt ist.

Im Folgenden werden einige typische Lernprogramme kurz vorgestellt.

5.4.1 ID3

ID3¹² [Quinlan 1984] ist einer der bekanntesten Algorithmen zum Problem des Lernens aus Beispielen ([Fensel 1993], [Kodratoff 1994]). Aus einer Menge von positiven und negativen Beispielen wird ein Entscheidungsbaum für eine Klassifizierung dieser Beispiele generiert. Eine allgemeine Beschreibung wird solange anhand von Merkmalen unterteilt, bis jede Teilmenge nur noch Beispiele einer Klasse enthält. Als Kriterium für eine Unterteilung dient das Entropiemaß der Informationstheorie. Die Beispiele sind in Form von Attribut-Wert-Listen repräsentiert.

Abbildung 5.3 gibt ein Beispiel für die Eingabe und das Ergebnis des Algorithmus (aus [Quinlan 1986]).

No.	Attribute				Class
	Outlook	Temp.	Humidity	Windy	
1	sunny	hot	high	false	N
2	sunny	hot	high	true	N
3	overcast	hot	high	false	P
4	rain	mild	high	false	P
5	rain	cool	normal	false	P
6	rain	cool	normal	true	N
7	overcast	cool	normal	true	P
8	sunny	mild	high	false	N
9	sunny	cool	normal	false	P
10	rain	mild	normal	false	P
11	sunny	mild	normal	true	P
12	overcast	mild	high	true	P
13	overcast	hot	normal	false	P
14	rain	mild	high	true	N

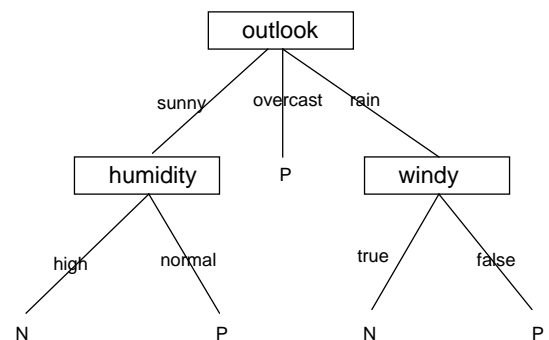


Abbildung 5.3: Lernen eines Entscheidungsbaums aus Beispielen mit ID3

ID3 wird in vorliegender Arbeit als Basisverfahren benutzt, um iterativ neue Konzepte für die Bildinterpretation zu lernen. Daher wird dieses Verfahren im Kapitel 8 ausführlich präsentiert, sowie Defizite und Erweiterungen vorgestellt.

5.4.2 Relax

Das Programm Relax [Fensel 1993] startet im Gegensatz zu ID3 von einer speziellen Beschreibung, die es solange sukzessive generalisiert, bis sie für alle Trainingsbeispiele korrekt ist. Als Ergebnis steht dann ein Satz von Regeln zur Verfügung, mit dem unbekannte Beispiele klassifiziert werden können.

Der Algorithmus startet, indem er jedes Beispiel als spezielle Regel auffaßt und generalisiert. Dies geschieht derart, daß das erste Merkmal der Regel “relaxiert” (d.h. weggelassen) wird, und geprüft wird, ob durch diese Generalisierung keine negativen Beispiele abgedeckt werden. Ist dies dennoch der Fall, wird dieser Schritt rückgängig gemacht. Auf diese Weise wird iterativ für alle Merkmale einer Regel geprüft, ob auf es verzichtet werden kann. Folgendes Beispiel soll diese erschöpfende Suche verdeutlichen:

Objekt	A: Ecken	B: gleichlange Seiten	C: rechte Winkel
Dreieck	3	0	1
...			

Alle 3 Merkmale (A,B,C) werden nun relaxiert (der Punkt in der Regel gibt an, welches Merkmal relaxiert wird), und es entsteht jeweils eine Regel *r* für das Konzept *c*, d.h. das Objekt “Dreieck”.

$$r_{123} : A = 3 \wedge B = 0 \wedge C = 1 \rightarrow c$$

¹²Das Kürzel steht für “Interactive Dichtomizer 3”

$$\begin{aligned}
 r_{.23} &: B = 0 \wedge C = 1 \rightarrow c \\
 r_{1.3} &: A = 3 \wedge C = 1 \rightarrow c \\
 r_{12.} &: A = 3 \wedge B = 0 \rightarrow c \\
 r_{1..} &: A = 3 \rightarrow c \\
 r_{.2.} &: B = 0 \rightarrow c \\
 r_{..3} &: C = 1 \rightarrow c \\
 r_{...} &: \rightarrow c
 \end{aligned}$$

Das Kriterium für die Auswahl einer dieser Regeln ist die Anzahl positiver Beispiele, die sie abdeckt, bzw. negativer, die sie ausschließt.

Es existieren verschiedene Erweiterungen des Programms, insbesondere zur Beschränkung der erschöpfenden Suche. Für stetige Attribute wurden Verfahren entwickelt, die eine Voreinteilung dieser Werte in bestimmte Intervalle vornimmt. Eine wichtige Eigenschaft des Programms ist die Behandlung unbekannter oder fehlender Werte in den Beispielen. Zum einen kann man annehmen, daß der Wert fehlt, da er für das Beispiel unerheblich ist. Relax sieht dieses Beispiel damit als vor-generalisiert an. Auf der anderen Seite gibt es die Möglichkeit, aus dem Trainingsdatensatz den wahrscheinlichsten Wert für dieses Beispiel dieser Klasse zu bestimmen und diesen einzusetzen. Vor der Bearbeitung werden also die unbekanntenen Werte durch den wahrscheinlichsten Wert der Klasse für das bestimmte Attribut ersetzt.

5.4.3 Winston's Arch

Das System von Winston [1975] lernt strukturelle Beschreibungen aus Beispielen. Ein Hauptmerkmal seines Ansatzes ist, daß sowohl die Beispiele, als auch die abgeleitete Beschreibung in Form eines semantischen Netzes, also in einer mächtigen Beschreibungssprache, repräsentiert werden. Die Objekte werden in Form von Eigenschaften und Relationen beschrieben. Das System lernt von positiven Beispielen und solchen, die das gesuchte Konzept *fast* darstellen, also nur knapp verfehlen (near miss). Das System hält zunächst das erste positive Beispiel als Konzept an. Ist das folgende Beispiel ein "near miss", so wird das Konzept soweit spezialisiert, daß es diesen Fall ausschließt. Ist es ein positives Beispiel, welches durch das gegebene Konzept noch nicht abgedeckt ist, so muß es dahingehend generalisiert werden.

Der Ablauf gestaltet sich folgendermaßen:

- Das erste positive Beispiel dient als Start-Regel
- Für alle folgenden Beispiele gilt: handelt es sich um einen "near miss", so ist die Regel derart zu spezialisieren, daß dieser Fall ausgeschlossen wird; ist es ein positives Beispiel, welches noch nicht durch die Regel abgedeckt wurde, so ist diese zu generalisieren, um das letzte Beispiel mit aufzunehmen.

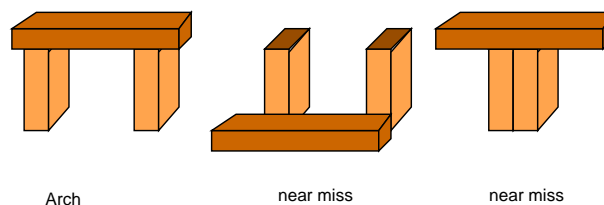


Abbildung 5.4: Beispiel und Gegenbeispiele (near miss) für das Konzept eines Bogens (Arch)

Auf diese Weise kann ein Bogen spezifiziert werden als bestehend aus zwei Stützen, die sich nicht berühren dürfen, sowie einem abdeckenden Querträger. Im Hintergrundwissen werden die zur Verfügung stehenden Objekte und Relationen spezifiziert, sowie die Generalisierungen, die bezüglich der Attribute möglich sind (z.B. die Objekte Parallelepipid und Pyramide generalisieren sich zu einem Polyeder, ...). Das Programm lernt nur von Fehlern, es kann nicht mit fehlerhaften Daten umgehen. Das Programm operiert im Objektraum - setzt damit voraus, daß die strukturierte Beschreibung bereits extrahiert ist. Dies stellt den "Flaschenhals" des Programms dar, es basiert auf low-level Programmen, die die komplexen Objekte und Relationen extrahieren. Arch ist ein klassisches Lernprogramm - es blieb allerdings bei der prototypischen Implementierung.

5.4.4 Conceptual Clustering

“Lernen aus Beobachtungen” (bzw. unüberwachtes Lernen) führt ein Zusammenfassen von Objekten aufgrund ihrer Ähnlichkeit durch. Problematisch ist hierbei die Definition der Ähnlichkeit bzw. des Ähnlichkeitsmaßes [Niemann 1981]. Im Falle der numerischen Taxonomie (vgl. z.B. [Vogel 1975], [Schalkoff 1992] [Chen & Schunck 1990]) wird die Ähnlichkeit üblicherweise über Abstandsmaße (euklidische Distanz, Mahalanobis-Distanz, ...) definiert. Eine Klassifizierung wird erreicht, indem die Ähnlichkeit innerhalb der einzelnen Klassen maximiert, und die zwischen den einzelnen Klassen minimiert wird. Als Nachteile der numerischen Methoden führen [Michalski & Stepp 1983] folgende an:

- Üblicherweise können nur numerische Attribute verwendet werden. Im Falle von symbolischen Attributen muß eine Dychtomisierung erfolgen. Es gibt allerdings auch Verfahren, welche numerische und symbolische Attribute gleichzeitig verwenden können (z.B. Autoclass [Cheeseman, Kelly, Self, Stutz, Taylor & Freeman 1988]).
- Es wird versucht, natürliche Cluster in Datensatz zu finden. Das Ergebnis ist somit eine Einteilung in verschiedene Merkmalsklassen - es entsteht jedoch keine charakteristische Beschreibung der so gewonnen Klassen.
- Für die Klasseneinteilung werden keine “Gestalt-Prinzipien” herangezogen, wie sie der Mensch normalerweise verwendet. D.h. die Merkmale werden generalisiert - und nicht die Beispiele als Gesamtheit. Im Gegensatz dazu geht man bei “conceptual clustering” -Verfahren von Beispielen aus, die sukzessive generalisiert bzw. spezialisiert werden, um schließlich zu dem endgültigen Konzept zu kommen. Auf diese Weise bleibt die Struktur der Beispiele erhalten.

Bei numerischen Verfahren wird die Ähnlichkeit grundsätzlich in einem numerischen Wert ausgedrückt, wohingegen bei symbolischen Verfahren die Ähnlichkeit z.B. als Zugehörigkeit zu einer bestimmten Menge definiert sein kann. Die Ähnlichkeit zweier Objekte A und B ist also eine Funktion der Objekte selbst, der sie umgebenden Objekte (E), und der Konzeptbeschreibung (C) der Objekte A und B: $f(A,B,E,C)$. Verdeutlichen läßt sich dies am Beispiel der Punktclustern in Abbildung 5.5, die offensichtlich zwei Kreise darstellen sollen. Ohne Hintergrundwissen über das Konzept eines Kreises, läßt sich keine sinnvolle Unterteilung der Punktmenge durchführen. Die Bildung zweier Cluster A und B, sowie die Zugehörigkeit eines Punktes zu einem dieser Cluster läßt sich daher nur über das Konzept eines Kreises bestimmen.

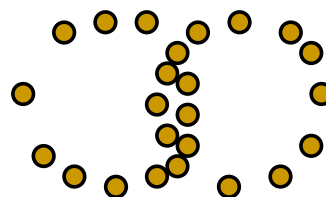


Abbildung 5.5: Beispiel für die Bedeutung von Hintergrundwissen für das Lernen: Voraussetzung für sinnvolle Einteilung der Punkte ist das Konzept eines Kreises

Cluster/2

In [Michalski & Stepp 1983] bzw. [Michalski & Stepp 1984] wird das Programm **Cluster/2** dargestellt. Es läßt sich als Cover-Prinzip beschreiben. Ausgangspunkt ist eine Initial-Beschreibung, welche meist durch ein oder mehrere Beispiele gebildet wird. Diese wird sukzessive generalisiert, bis alle positiven Beispiele erfaßt sind, und alle negativen Beispiele ausgeschlossen sind. Das Programm startet mit einer vorgegebenen Anzahl (k) von zu bildenden Clustern. Dazu werden aus der Beispielmenge k Start-Cluster entnommen. Der Ablauf gestaltet sich dann wie folgt:

- Starte mit einer Konjunktion von Bedingungen, die von der Zielklasse erfüllt werden, die aber die anderen Klassen nicht umfassen.

- Übernahme diese Konjunktion als Teil der Ergebnisregel.
- Entferne alle Beispiele, die durch diese Regel abgedeckt werden und wiederhole das ganze mit den übrigen Beispielen.

Das Programm nutzt als Ausgangsbeschreibung die gegebenen Beispiele, die nach und nach generalisiert werden. Damit ist gewährleistet, daß bei den entstehenden Konzepten die Gestalt-Prinzipien eingehalten sind, d.h. die Objekte werden als Gesamtes angesehen, und nicht lediglich durch eine zufällige Zusammenstellung von Merkmalen eingeteilt. Von Nachteil ist jedoch, daß das Programm für die Selektion des optimalen Konzepts eine erschöpfende Suche durchführt, und daher sehr zeitintensiv ist.

Cobweb

Cobweb [Fisher 1987] dient zur Klassifizierung von Daten. Dies können sowohl numerische, als auch symbolische Werte sein, welche in Form von Attribut-Wert-Listen gegeben sind. Der grundlegende Ansatz von Cobweb basiert darauf, daß neue Konzepte inkrementell gebildet werden. Ein Konzept zeichnet sich dabei dadurch aus, daß alle seine Beispiele gemeinsame Charakteristika aufweisen. Die Besonderheit des Systems besteht darin, daß keine klassifizierten Beispiele vorgegeben werden müssen, sondern daß die Konzepte unüberwacht gefunden werden. Je mehr Beispiele gegeben sind, desto detaillierter kann das entstehende Konzept werden. Das Ergebnis ist ein Klassifikationsbaum. Wie ID3 basiert Cobweb auf einem informationstheoretischen Ansatz. Der Entscheidungsbaum wird inkrementell aufgebaut, indem jede neue Instanz in diejenige Klasse eingefügt wird, für die es die kürzeste Beschreibung benötigt. Der Algorithmus verfügt über Möglichkeiten, einzelne Knoten zusammenzufassen, bzw. zu teilen, falls dies für die Gesamtinformation förderlich ist. Abbildung 5.6 gibt ein Beispiel für die Eingabe und das Ergebnis des Algorithmus.

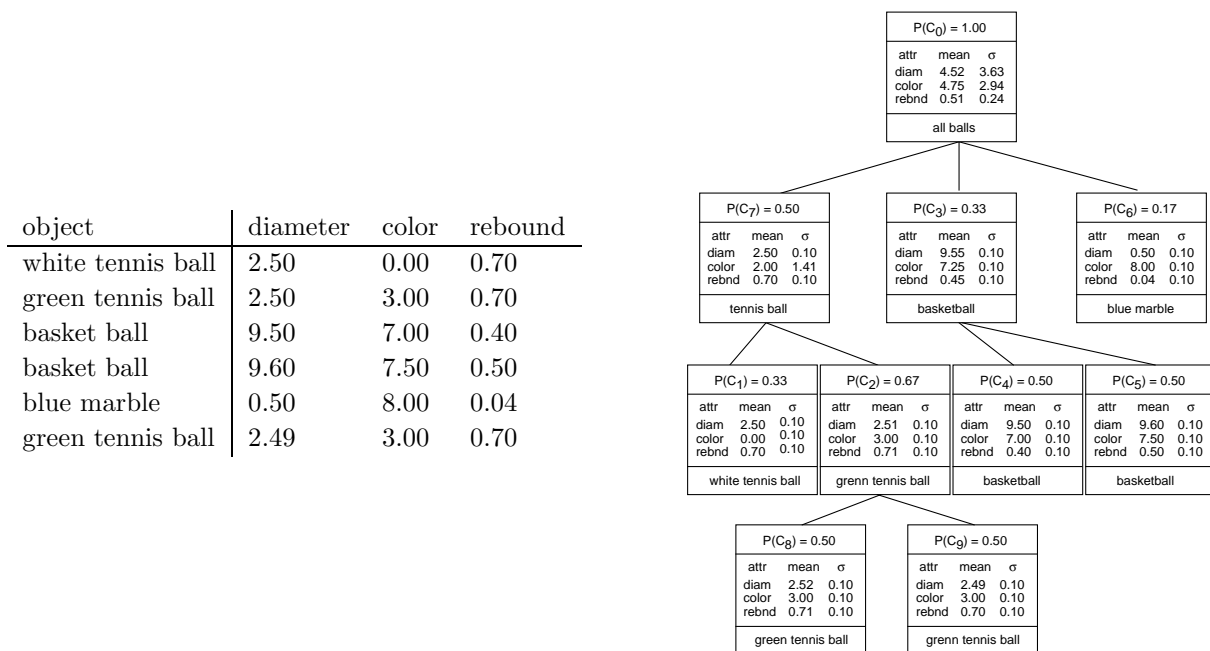


Abbildung 5.6: Klassifizierung von Bällen mittels Cobweb (aus [McKusick & Thompson 1990])

Eine Erweiterung von Cobweb namens Labyrinth [Thompson & Langley 1991] unterstützt die Ermittlung strukturierter Beschreibungen, indem das Programm wiederholt angewendet wird: wurden in einer Iteration Objektbeschreibungen ermittelt, so können diese in einem nächsten Schritt als Attribut eingehen.

5.5 Neuronale bzw. konnektionistische Lernverfahren

Wie im Kapitel 3.5 bereits angesprochen, lassen sich Neuronale Netze als Assoziativspeicher bestehend aus Knoten und Kanten interpretieren. Steht ein einzelner Knoten für ein Objekt bzw. ein Konzept, so spricht man von lokalem Konnektionismus, welcher einem semantischen Netz entspricht. Um verteilten Konnektionismus geht

es hingegen, wenn die einzelnen Knoten eigene Verarbeitungseinheiten darstellen. Ein Konzept ist dann nicht mehr in einem Knoten allen repräsentiert, sondern in einem Aktivierungsmuster verteilt. Intelligenz erwächst daher aus den Verbindungen zwischen einzelnen Knoten.

Neuronale Netze erlebten eine Hochphase mit der Erfindung des Perceptrons - allerdings bald darauf einen starken Niedergang mit der Veröffentlichung von Minsky & Papert [1969], die die Möglichkeiten solcher Netze als stark eingeschränkt ansahen. Mitte der 80iger Jahre begann ein neues Interesse an dem Thema, als mit dem Back-Propagation-Algorithmus (PDP-Gruppe) [Rumelhard & McClelland 1986] ein solides mathematisches Gerüst für neuronale Systeme bereitet wurde [Sharples et al. 1989].

Im Folgenden werden das Back-Propagation-Netz und assoziative Netze angesprochen, sowie deren mathematische Basis kurz skizziert (vgl. [Pao 1989]), bevor einige Beispiele für Anwendungen neuronaler Netze in der Bildanalyse vorgestellt werden. Diese Übersicht erhebt keinen Anspruch auf Vollständigkeit, vielmehr soll lediglich ein Eindruck von der zugrundeliegenden Mathematik vermittelt werden.

5.5.1 Backpropagation Algorithmus

In konnektionistischen Ansätzen werden die Eingabedaten in Form von Merkmalsvektoren repräsentiert. Die zugehörige Klassifikation ist als Ausgabegröße gegeben. Das Netz stellt den Zusammenhang zwischen Ein- und Ausgabegrößen her und wird im Verlauf des Lernens optimal eingestellt.

Die Eingabe ist als Sammlung von Objekten gegeben, welche durch Merkmalsvektoren \underline{x} charakterisiert sind. Diese sind einer bestimmten Ausgabe zugeordnet. Diese Konfiguration läßt sich als Gleichungssystem betrachten. Eine Matrix X aus m Musterzeilen mit je n Merkmalen wird mit einem (unbekannten) Gewichtsvektor multipliziert, um zu einer Sollausgabe zu gelangen. Dies stellt das einfachste Modell dar, bei dem lediglich das Eingabemuster gewichtet wird. Ziel ist es, die Gewichte so festzulegen, daß alle Muster richtig klassifiziert werden. Analytisch bedeutet dies, daß die Trainingsmuster in der Form von Zeilenvektoren (Matrix X : m Muster und n Attribute) angelegt werden, die Gewichte bilden einen Vektor w , die Ausgabedaten den Vektor b . Nun muß das Gleichungssystem

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1N} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{M1} & x_{M2} & x_{M3} & \cdots & x_{MN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} \quad (5-1)$$

gelöst werden. Die unbekanntes Gewichte lassen sich ermitteln über

$$\begin{aligned} \underline{w} &= X^{-1}\underline{b} \\ \text{bzw. } \underline{w} &= (X^t X)^{-1} X^t \underline{b} \end{aligned} \quad (5-2)$$

Um die Lösung der Pseudoinversen zu umgehen, wird der Gewichtsvektor meist iterativ gelöst. Dabei wird das Gewicht um bestimmte Inkremente $\Delta \underline{w}$ korrigiert.

$$\Delta \underline{w} = \eta \delta \underline{x} \quad (5-3)$$

mit

$$\begin{aligned} \delta &= b - \underline{w}^t \underline{x} && : \text{ Differenz zwischen aktuellem und Soll-Output} \\ \eta &&& : \text{ Proportionalitätsfaktor} \end{aligned}$$

Die zufälligen Initialgewichte werden iterativ so lange angepaßt, bis die korrekte Klassifizierung erreicht ist.

Es lassen sich nun zwischen Ein- und Ausgabe mehrere Zwischenebenen, sog. hidden layers, schalten. Man spricht dann von einem Backpropagation-Netz [Rumelhard & McClelland 1986]. Ein Netz besteht aus mehreren Layers, die sich dadurch auszeichnen, daß der Input eines jeden Knotens die Summe der gewichteten Ausgabe der vorherigen Ebene darstellt. Jeder Knoten wird also abhängig von seinem Input und seiner Aktivierungsfunktion aktiviert. Im Unterschied zum linearen Netz ist die Aktivierungsfunktion an den Knoten jedoch nicht mehr linear, sondern eine allgemeine analytische Funktion. Die Netzeingabe net_j am Knoten i in der Ebene j ist :

$$net_j = \sum w_{ji} o_i \quad (5-4)$$

Die hieraus resultierende Ausgabe des Knotens j ist dann:

$$o_j = f(\text{net}_j) \quad , \text{ z.B.: } \quad o_j = \frac{1}{1 + e^{-(\text{net}_j + \theta_j)/\theta_0}} \quad (5-5)$$

wobei f die Aktivierungsfunktion darstellt. Für eine sigmoide Funktion f dient der Wert θ_j als Schwellwert, während θ_0 die Form der Kurve steuert. Diese Gleichungen gelten für alle Layers.

In der Trainingsphase gleicht das System die Gewichte und die Schwellwerte anhand eines vorgegebenen Musters an. Dies wird für alle Musterdaten durchgeführt. Sind alle Trainingsmuster klassifiziert, so ermittelt sich der Systemfehler als quadratische Abweichung aller Soll- und Ist-Ausgabewerte. Rumelhard & McClelland [1986] modifizieren die Gewichte in einer Weise, die den individuellen Fehler E_p am schnellsten reduziert. Die Gewichte einer bestimmten Ebene werden mit der Formel

$$\Delta \underline{w}_{ji} = \eta \delta_j \underline{a}_i \quad (5-6)$$

ermittelt. Das δ_j errechnet sich iterativ aus

$$\delta_j = f'(\text{net}_j) \sum_k \delta_k w_{kj} \quad (5-7)$$

Die Bestimmung der Gewichte muß dabei für alle Ebenen durchgeführt werden. Die Änderungen der obersten Ebene lassen sich direkt bestimmen und anschließend auf die jeweils nächste Ebene übertragen (propagieren).

Für jede Aufgabenstellung ist die Frage des Netzdesigns zu klären. Freie Parameter sind dabei insbesondere die Anzahl der Hidden Layers, sowie der Anzahl der Knoten dieser Zwischenschichten. Ein wesentlicher Vorteil dieser Algorithmen ist die Trennbarkeit von Mustern: Ein 3-Layer-Netz kann beliebige Mustercluster unterscheiden (auch solche, die räumlich getrennt sind). Wichtig ist, daß keine Annahmen über die zugrundeliegende Verteilung der Muster nötig sind.

5.5.2 Assoziativspeicher

Ein Assoziativspeicher ist derart organisiert, daß Ein- (\underline{x}) und Ausgabevektor \underline{y} ein dyadisches Produkt bilden.

$$M = \underline{y}_k \underline{x}_k^t \quad (5-8)$$

Sind der Assoziativspeicher M sowie ein Eingabemuster \underline{x} gegeben, so läßt sich \underline{y} gewinnen durch

$$\begin{aligned} M \underline{x}_k &= \underline{y}_k \underline{x}_k^t \underline{x}_k \\ &= \langle \underline{x}_k^t \underline{x}_k \rangle \underline{y}_k \end{aligned} \quad (5-9)$$

das Ergebnis ist also das gesuchte Ausgangsmuster \underline{y}_k , wenn das Skalarprodukt von x mit sich selbst gleich 1 ist, die Vektoren also normiert sind.

Ist der Eingabevektor nun nicht \underline{x} , sondern eine leicht veränderte Version davon \underline{x}' , so ergibt sich die Systemantwort zu

$$\begin{aligned} M \underline{x}'_k &= \underline{y}_k \underline{x}_k^t \underline{x}'_k \\ &= \langle \underline{x}_k^t \underline{x}'_k \rangle \underline{y}_k \end{aligned} \quad (5-10)$$

d.h. \underline{y}_k wird noch immer gefunden, allerdings mit geringerer Amplitude, wenn $\langle \underline{x}_k^t \underline{x}'_k \rangle < \langle \underline{x}_k^t \underline{x}_k \rangle$ ist.

Mehrere Muster-Ergebnis-Paare können in einer Matrix M durch einfaches Aufsummieren abgespeichert werden:

$$M = \sum_k \underline{y}_k \underline{x}_k^t \quad (5-11)$$

Zu beachten ist allerdings, daß eine unverfälschte Systemantwort \underline{y}_m allerdings nur dann erreicht wird, wenn kein sog. 'cross-talk', d.h. Abhängigkeiten zwischen den verschiedenen Musterpaaren bestehen: dies gilt nur für orthogonale Eingangsvektoren.

Setzt man folgende Funktion an

$$M = \sum y_k (\alpha_k \underline{x}_k^t) \quad (5-12)$$

so kann ein optimaler Speicher M durch einen kleinste-Quadrate-Ansatz ermittelt werden, bei dem die Parameter α_k derart variiert werden, daß der Systemfehler E minimal wird. Der Systemfehler E wird als Differenz zwischen Ist- und Soll-Ausgabe angesetzt

$$E = \left(\sum_m \| \underline{y}_m - \underline{y}'_m \|^2 \right)^{1/2} \quad (5-13)$$

Solch ein Speicher kann auch autoassoziativ genutzt werden. Dabei wird das Außenprodukt des Eingangsvektors mit sich selbst gebildet. Im entstehenden Netz sind alle Knoten miteinander in Verbindung (Hopfield-Netz) [Tank & Hopfield 1988].

Ein Assoziativspeicher zeichnet sich durch folgende Charakteristika aus:

- er kann viele assoziierte Muster (sog. Stimulus-Response-Paare) speichern, indem neue Muster (neues Wissen) addiert werden (gemäß Formel 5-11),
- diese Speicherung erfolgt selbstorganisiert,
- die Information wird verteilt, d.h. auf redundante Art und Weise gespeichert,
- auf ein Eingangsmuster erzeugt das Netz ein entsprechendes Ausgangssignal; diese Systemantwort ist auch dann noch richtig, wenn das Eingangssignal gestört oder nur unvollständig ist.

5.5.3 Beispiele für den Einsatz Neuronaler Netze in der Bildanalyse

Im folgenden werden einige Beispiele für die Nutzung Neuronaler Netze in der Bildanalyse exemplarisch vorgestellt. Allen Anwendungen ist gemeinsam, daß sie schnell und (in gewissen Grenzen robust) arbeiten, und auch - im Bezug auf die vorgegebenen Testdaten - gute Ergebnisse liefern. Probleme liegen sowohl in der Konfiguration des Netzes, als auch in der Interpretation des Ergebnisses. Freie Parameter für die Konfiguration sind dabei Art und Umfang der Merkmalsvektoren, sowie die Anzahl der Ebenen und Knoten im Netz. Das Ergebnis ist im trainierten Netz implizit codiert - es kann daher nur im Bezug auf konkrete Beispiele verifiziert werden. Somit ist keine allgemeine Beurteilung des Ergebnisses möglich.

Detektion von Paßpunkten

Die Detektion von Paßpunkten in Luftbildern wird von Kepuska & Mason [1991] beschrieben. Das neuronale Netz wird trainiert mit einer großen Anzahl von Bildausschnitten, in denen sich Paßpunkte befinden. Hierzu wurden typische Beispiele ausgewählt, d.h. verschiedenen Orientierungen und Auflösungen, unterschiedliche Radiometrie, etc. Als Gegenbeispiele wurden auch Bildausschnitte mit aufgenommen, die keine Paßpunkte enthalten. Die Autoren berichten von einer Erfolgsquote von 85 % für signalisierte Punkte in 34 Bildern. Sie vermuten, daß eine größere Menge an Trainingsdaten das Ergebnis noch verbessern kann.

Kantenextraktion

Spreeuwers [1992] nutzt den Back-Propagation-Algorithmus zur Kantendetektion in Grauwertbildern. Vorgegeben werden Bildausschnitte, in denen Kanten liegen. Diese müssen repräsentativ sein und alle möglichen Kantentypen erfassen. Er setzt ein 2-Ebenen-Netz an - auf der Basis der Überlegung, daß eine Kantendetektion üblicherweise ebenfalls ein zweistufiger Prozeß ist: Filterung mit anschließender Schwellwertbildung. Es zeigte sich allerdings, daß das sich herausbildende Netz dennoch nicht interpretierbar war, da die "Filterkerne" des Systems nicht mit gängigen Operatoren vergleichbar waren. Ein Vergleich mit anderen Kantenoperatoren (Sobel, Canny, Marr-Hildreth) wurde auf der Basis synthetischer (verrauschter) Bilder durchgeführt. Hierbei erreichte das Verfahren die besten Ergebnisse bezogen auf die Trainingsdatensätze.

Navigation

Thorpe, Amidi, Gowdy, Hebert & Pomerleau [1992] stellen ein System zur Fahrzeugsteuerung vor, welches auf der Basis eines zweischichtigen neuronalen Netzes arbeitet. Das Verfahren nutzt dabei als Eingabe einen Bildausschnitt, der die vor dem Auto befindliche Straße zeigt. Die Soll-Ausgabe ist die entsprechende Lenkbewegung des Fahrers. Schon nach kurzer Lernzeit (5 min) kann das System eigenständig lenken.

Erkennen von Gesichtern

Weng et al. [1993] stellen ein System *Cresceptron* vor, welches in der Lage ist, seine Konfiguration automatisch während des Lernens zu bestimmen. Ausgehend von Bildprimitiven (Zero-Crossings und gerichtete Gradienten), wird sukzessive ein Netz aufgebaut. Das Netz kann sich automatisch erweitern, wenn signifikante Konzepte entdeckt werden. Als Vorzüge des Verfahrens führen die Autoren an, daß es tolerant gegenüber Änderungen im Objekt ist (z.B. Erkennen von Gesichtern trotz unterschiedlicher Grimassen), und daß es sich für verschiedenste Aufgabenstellungen einsetzen läßt - was anhand von Szenen innerhalb und außerhalb von Räumen demonstriert wird.

5.6 Diskussion und allgemeine Kritik

Das Grundprinzip der Induktion, welches den Lernverfahren zugrunde liegt, bedingt, daß die Richtigkeit der ermittelten Ergebnissen nicht bewiesen werden kann. Hieraus ergeben sich einige Grundprinzipien für den Einsatz von Lernverfahren [vgl. Podiumsdiskussion über Neuronale Netze anlässlich der DAGM in München 1991]:

- Die Ergebnisse des Lernverfahrens sollten aussagekräftig und für den Menschen verständlich sein. Diese macht den Einsatz von Neuronalen Netzen problematisch: hier kann nicht nachvollzogen werden, warum das System auf welche Weise reagiert, da das Wissen aus einem unübersehbaren Netz von Knoten und Gewichten besteht. Der Vorteil struktureller Lernverfahren liegt darin, daß die Ergebnisse - aufgrund der gewählten Beschreibungsform - zumindest theoretisch für den Menschen les- und nachvollziehbar sind. Diese Verfahren beinhalten also eine Art von Erklärungskomponente zur Verifikation der Ergebnisse.
- Der Einsatz von Lernverfahren ist dort sinnvoll, wo eine algorithmische Lösung unbekannt bzw. zu aufwendig ist. In Fällen also, wo es zwar eine Fülle von Beispielen für eine bestimmte Tatsache gibt, die inneren Zusammenhänge zwischen allerdings unbekannt sind, ergeben sich viele Anwendungen. Ein klassisches Beispiel ist die Klassifikation von Sojabohnenkrankheiten. Pflanzen können als gesund oder krank klassifiziert werden. Welches jedoch die Gründe für die Krankheit sind, läßt sich rein aus dem Anblick einer Pflanze nicht ermitteln. Hier können nun die verschiedensten Beobachtungen zu der Pflanze zusammengetragen werden (Wuchs, Blattform, Farbe, Blüten, aber auch Standorteigenschaften), die schließlich in ihrer Kombination einen Zusammenhang zur Klassifikation in gesund oder krank erlauben. Weitere Beispiele sind: Schach, Diagnose von Krankheiten, Überprüfen der Kreditwürdigkeit von Bankkunden, Börse.
- Lernverfahren sollten nur dort unabhängig eingesetzt werden, wo es nicht auf 100 %-ige Sicherheit ankommt. So ist die Frage, ob man sich z.B. bei Waffenkontrollen in Flughäfen ausschließlich auf die Nutzung von automatisch gelerntem Wissen verlassen sollte. Auch sollte das Sicherheitssystem einer komplexen Anlage nicht allein auf der Basis solches Wissens arbeiten. Zumindest sollten die Ergebnisse eingehender Plausibilitätsprüfungen unterzogen werden, bevor sie Einsatz finden.
- Das Lernen sollte graduell, d.h. in kleinen Schritten erfolgen, um Über-Generalisierungen zu verhindern.

Zusammenfassend ist zu sagen, daß es für verschiedene Aufgabenstellungen auch die unterschiedlichsten Lösungsansätze gibt. Beim Entwurf eines Systems ist daher zu klären, welche Art von Wissen bereits vorliegt, und was daraus abgeleitet werden soll. Wenn auch die Sprache bestimmt, was überhaupt gelernt werden kann, ist doch zu sehen, daß die Repräsentation und der Algorithmus eine Einheit bilden, die eine gewisse Funktionalität bietet. Allerdings stellt diese immer ein Kompromiß dar: Neuronale Netze bieten Robustheit und Toleranz gegenüber unvollständigen Daten, auch arbeiten die Verfahren in gewissen Bereichen graduell, d.h. kleine Änderungen in den Ausgangsdaten führen auch nur zu kleinen Änderungen im Ergebnis. Der wesentliche Nachteil besteht

allerdings darin, daß der Lösungsweg nicht nachvollziehbar ist¹³. Zudem lassen sich für das Netzdesign keine allgemeinen Regeln angeben.

Das gelernte Wissen kann für die Interpretation unbekannter Information genutzt werden. Allerdings ist dieses Wissen nicht explizit gemacht - explizit in dem Sinne, daß es für den Menschen verständlich ist. Dies leisten - im Gegensatz hierzu - strukturelle Lernverfahren. Alle Verfahren setzen voraus, daß das Problem vorher in die entsprechende Form transformiert wird¹⁴. Die Wahl eines Verfahrens ist immer abhängig von Thema und Aufgabenziel. Im Falle des Wiederentdeckens der Kepler-Gesetze genügte die Attribut-Wert-Listen-Darstellung [Langley 1977]. Höhere und damit ausdrucksstärkere Beschreibungssprachen erfordern meist noch mehr zusätzliches Hintergrundwissen: z.B. sind im Programm Arch Generalisierungshierarchien von Objekten vorzugeben.

¹³Towell & Shavlik [1992] präsentieren ein Verfahren, welches die Vorteile von symbolischem und konnektionistischem Lernen kombiniert. Der eigentliche Lernschritt erfolgt subsymbolisch, während das Ergebnis - das trainierte Netz - anschließend in eine regelbasierte Form gebracht wird. Das Regelwerk wird allerdings erzeugt, indem für jeden Knoten im Netz eine eigene Regel gebildet wird - was sicher nicht unbedingt zur Verständlichkeit beiträgt.

¹⁴Ein ESPRIT Projekt - die sogenannte Machine Learning Toolbox (MLT) - hat zum Inhalt, viele Lernverfahren in einer einheitlichen Umgebung zu anzubieten. Insbesondere soll einen einheitliche Repräsentation angestrebt werden, um einen Datensatz mit mehreren Programmen bearbeiten zu können [Sleeman 1994].

6 Verwandte Arbeiten

Im Bereich des Bildverstehens gibt es zahlreiche Arbeiten, die sich mit dem Problem einer flexibleren Szeneninterpretation beschäftigen. Bislang ist offensichtlich der Mensch das einzige "System", welches eine schnelle, meist eindeutige Interpretation der ihn umgebenden Welt zu leisten vermag. Eine derartige Fähigkeit impliziert insbesondere das Erkennen unerwarteter und unbekannter Objekte.

Die Forschungsanstrengungen gehen auf der einen Seite dahin, generische Modelle – im Gegensatz zu spezifischen – bereitzustellen, um somit nicht nur Einzelobjekte, sondern ganze Klassen von Objekten erkennen zu können. Auf der anderen Seite wird versucht, die Flexibilität durch automatisches Lernen von Modellen zu erreichen.

6.1 Generische Modelle für die Objekterkennung

Besonders im Bereich der automatischen Interpretation natürlicher Szenen und Objekte unserer Umwelt finden sich zunehmend Ansätze, die auf generischen Modellen beruhen.

6.1.1 Generalisierung der geometrischen Beschreibungselemente

Das System PARVO [Bergevin & Levine 1993] erkennt 3D-Objekte aus 2D-Ansichten. Es basiert auf dem von Biederman [1985] postulierten Prinzip des "Erkennens aufgrund von Komponenten" (recognition by components). Dieses besagt, daß der Mensch komplexe Objekte durch Erkennen seiner Einzelteile wahrnimmt. Die elementaren Einzelobjekte sind 3D-Körper, sog. Geons (vgl. Abbildung 6.1), die in ihrer 2D-Struktur schon eindeutig bestimmt sind¹⁵.

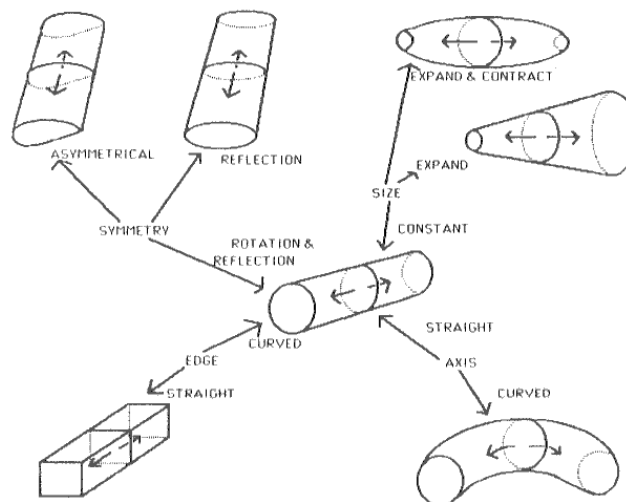


Abbildung 6.1: Generische Strukturen – Geons (aus [Biederman 1985])

PARVO verarbeitet Objekte, die in Form von Strichzeichnungen gegeben sind. Das System kennt eine Sammlung von 23 Objekten, deren strukturelle Beschreibung a priori vorgegeben sind. Diese qualitative Beschreibung der Objekte und ihrer Beziehungen wird manuell durch den Nutzer bestimmt und sind als Objektmodellgraph abgelegt. Zur Klassifizierung unbekannter Objekte werden diese zunächst in ihre Einzelteile (Geons) segmentiert; auch ihre räumlichen Beziehungen untereinander werden berechnet. Dieses komplexe Objekt kann anschließend durch Vergleich mit den Objekten der Modelldatenbank erkannt werden. Der generische Aspekt dieses Systems liegt in der Repräsentation der Einzelkomponenten als Geons. Voraussetzung ist die Erkennung dieser elementaren Bestandteile. Dies wird durch die starke Segmentierungskomponente des Systems erreicht. Sie erlaubt, auch verdeckte Objektteile als zusammenhängend zu erkennen. Nachteilig ist die notwendige manuelle Modelldefinition.

¹⁵Biederman begründet dies mit dem Prinzip der "Nicht-Zufälligkeit". Regelmäßigkeiten im Bild resultieren prinzipiell aus den entsprechenden Regelmäßigkeiten in der Realität - Unstimmigkeiten sind dann eher zufällig und auf spezielle Perspektiven beschränkt.

Ein ähnliches Verfahren wenden Lee, Pong, Esterline & Slagle [1992] an. Das KOR (Knowledge-Based Object Recognition) System erkennt 3D-Objekte in ihrer 2D-Projektion. Ein Objekt wird dabei über einen Graphen beschrieben. Zentralelemente sind hier allerdings keine Geons, sondern Regionen. Diese sind charakterisiert entsprechend ihrer Zusammensetzung aus Punktknoten bestimmten Typs. Diese Objektmodelle können automatisch erworben werden. Dabei wird allerdings angenommen, daß sich im Bild jeweils nur ein Objekt befindet.

Mit dem von Dickinson, Pentland & Rosenfeld [1992] vorgestellten System lassen sich polyedrische 3D-Objekte erkennen. Grundlage ist ein relationaler Graph, der angibt, wie eine hierarchische Objektbeschreibung aus elementaren Einheiten gebildet werden kann: ausgehend von Linienstücken werden Flächen und daraus komplexere 3D-Objekte gebildet. Das Zusammenfassen von Objektteilen zu Objekten der nächsten Stufe geschieht aufgrund bestimmter Relationen. Diese sind fix vorgegeben, allerdings generischer Natur: Parallelität, Symmetrie, Kollinearität. Sie ermöglichen damit eine qualitative, d.h. generische Beschreibung. Die Übergänge zwischen den einzelnen Darstellungsebenen werden über bedingte Wahrscheinlichkeiten gesteuert. Diese Wahrscheinlichkeiten für ein Zusammenfassen von Objektteilen wurden vorab anhand von 10 Beispielobjekten ermittelt. Abbildung 6.2 zeigt einen Ausschnitt aus den möglichen Übergängen von Ebene zu Ebene. Mit einer Wahrscheinlichkeit von 84 % gehört beispielsweise eine U-Form (rechts unten) zu einem Parallelogramm. Dieses ist wiederum Bestandteil von einem Quader - mit einer Wahrscheinlichkeit von 36 %; in bestimmten Projektionen kann es aber auch Teil eines Pyramidenstumpfs sein, allerdings mit einer sehr geringen Wahrscheinlichkeit von nur 4 %.

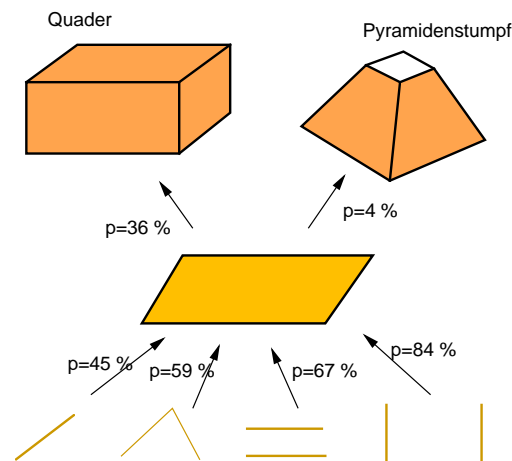


Abbildung 6.2: Wahrscheinlichkeiten der Gruppierung von Merkmalen

Der Ablauf der Erkennung gestaltet sich datengetrieben (bottom-up) als Gruppierung von Merkmalen zu höherwertigen Objektteilen, wobei jeweils die wahrscheinlichste Zusammenfassung durchgeführt wird. Der generische Aspekt dieses Verfahrens liegt in der statistischen Angabe der Zusammenhänge der Beschreibungen auf verschiedenen Hierarchiestufen. Auf diese Weise lassen sich komplexe Objekte aus sehr elementaren Bestandteilen (Kantenstücke) zusammensetzen.

Zur Lokalisierung und Rekonstruktion von Gebäuden in Luftbildern verwenden Mohan & Nevatia [1989] den Ansatz, Gebäude als Rechtecke zu beschreiben. Somit erfolgt bottom-up eine gezielte Suche nach geraden Kantenstücken, die sich (über Parallelitäts- und Orthogonalitätsbedingungen) zu Rechtecken zusammenfassen lassen. Durch zusätzliche Nutzung der Stereoinformation können Gebäude dreidimensional rekonstruiert werden. Fua & Hanson [1987] nutzen dabei noch die Möglichkeit, mit Kenntnis des Modells eine neuerliche, gezielte Segmentierung anzustoßen, und somit eine gemischte top-down – bottom-up Interpretation durchzuführen.

Stilla & Jurkiewicz [1991] lösen die Aufgabe der Erkennung von Siedlungsstrukturen in Luftbildern in einem Blackboard-orientierten Ansatz. Das Wissen über Siedlungen wird dabei in einfachen Grammatik-Regeln (Prozeduren) abgelegt. Ähnlich der Vorgehensweise bei Mohan & Nevatia [1989] erfolgt ein sukzessives Gruppieren einzelner Kantenstücke zu längeren Linien, dann zu Parallelogrammen. Zwei solcher Parallelogramme bilden ein Hausdach. Mehrere solcher Häuser stellen eine Hausreihe längs einer Straße dar. Mit einem auf Spezialhardware basierendem Blackboard wird die Zuordnung erreicht.

6.1.2 Nutzung nicht-geometrischer Merkmale

Die bisher vorgestellten Verfahren führen Generalisierungen auf der Ebene der geometrischen Beschreibung durch. Verallgemeinerungen lassen sich auch bezüglich weiterer Objektaspekte durchführen, z.B. ihrer Funktion oder physikalischen Gesetzmäßigkeiten.

Stark & Bowyer [1991] stellen ein System vor, welches Objekte aufgrund ihrer Funktionalität erkennt. So werden Stühle als "etwas, auf dem man sitzen kann" beschrieben. Diese Beschreibung impliziert, daß das Objekt eine Sitzfläche, eine gewisse Höhe, sowie Standfestigkeit aufweisen muß, um als Stuhl klassifiziert zu werden. Mit dieser allgemeinen Beschreibung lassen sich eine Reihe von Gegenständen als Sitzgelegenheit identifizieren.

Insbesondere in Realweltszenen lassen sich Objekte meist nicht über ihre Geometrie hinreichend beschreiben (Bäume, Wolken, Himmel, ...). Strat & Fischler [1991] machen sich daher den Kontext einer Szene zunutze, indem sie die Beziehungen, die zwischen diesen Objekten herrschen, modellieren und in elementaren Regeln fassen. Diese Hypothesen werden in einem speziellen Kontrollsystem (Core Knowledge System) abgearbeitet, um schließlich zu einer Szenenbeschreibung zu kommen.

Das Ziel des Verfahrens von Birnbaum, Brand & Cooper [1993] ist eine effiziente Segmentierung von Objekten einer Szene. Die Trennung von Objekten und Hintergrund wird erreicht, indem physikalische Gesetzmäßigkeiten zur Argumentation herangezogen werden: ein Objekt kann nicht "im luftleeren Raum" schweben, es benötigt eine Aufhängung bzw. Halterung. Diese physikalischen Begründungen steuern die Segmentierung, indem gezielt weitere Objekte gesucht werden, die solch eine Stützfunktion erfüllen. Abbildung 6.3 zeigt ein Beispiel für die Segmentierung vor sehr bewegtem Hintergrund. Die schwarzen Linien stellen dabei die Wege dar, die das Schlußfolgerungssystem während der Abarbeitung des Bildes zurücklegt.

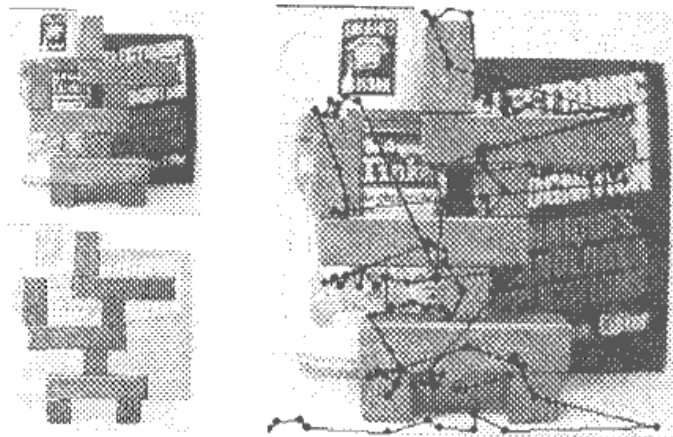


Abbildung 6.3: Beispiel für Segmentierung aufgrund physikalischer Aspekte

6.2 Beispiele für Lernverfahren im Bereich der Bildanalyse

Im folgenden werden einige Ansätze aufgezählt, bei denen Lernverfahren in der Bildanalyse eingesetzt werden. Diese Auflistung soll einen Überblick über das aktuelle Spektrum geben. Es zeigt sich, daß viele verschiedene Anwendungen existieren, die z.T. auf der Basis sehr diverser Methoden basieren. Die vorgestellten Arbeiten sind aus dem Bereich der 2D-Szenenanalyse. Die meisten Ansätze gehen von einer festen Anzahl an möglichen Beschreibungselementen aus, die dann mit Hilfe der Beispiele zu Modellen zusammengesetzt werden. Die Ansätze unterscheiden sich primär dadurch, welche Merkmale a priori festgelegt werden, bzw. ob neue Merkmale während des Lernprozesses hinzukommen können.

Hättich [1989] beschreibt ein System zum Erlernen von Modellen von 2D-Werkstücken. Seine Vorgehensweise besteht darin, verschiedene Bilder des Werkstücks zu segmentieren. Die extrahierten Merkmale, die in möglichst vielen Bildern ermittelt wurden, bilden dann das Objektmodell. Vorteil dieses Verfahrens ist, daß gut zu segmentierende Merkmale auf diese Weise ein hohes Gewicht in der Modellbeschreibung erhalten. Dies kann effizient für die Nutzung der Modelle zum Greifen des Werkstücks eingesetzt werden.

Kappenberger [1990] realisiert eine Wissenserwerbskomponente, die in ein Expertensystem integriert ist. Sein Anwendungsgebiet stammt aus dem Bereich des Verkehrswesens und dient der Beschreibung von Straßen und Kreuzungen. Aus einer vorgegebenen Sammlung von idealisierten Bildbeispielen ermittelt er durch Generalisierung eine strukturelle Beschreibung der gesuchten Objekte. Im Unterschied zu vorliegendem Ansatz geht er von einer festen Menge an Basisobjekten und Relationen aus, die a priori bekannt sind. Es ist keine Möglichkeit vorgesehen, diese zu erweitern.

Segen [1988] leitet klassifizierende Beschreibungen von 2D-Objekten aus einer Sammlung gegebener Beispiele ab. Im Unterschied zu den meisten anderen Programmen basiert sein Programm nicht auf einer fix vorgegebenen Menge an Merkmalen und Relationen, sondern ermittelt diese Deskriptoren während des Laufs selbst. Zunächst wird die 2D-Kontur der Objekte nach Punkten mit maximaler Krümmung segmentiert. Diese werden als Merkmale der ersten Stufe bezeichnet. Neue komplexere Merkmale werden erzeugt, indem die Deskriptoren der vorigen Stufe jeweils gruppiert werden. Welche Merkmale aggregiert werden, wird über ein informationstheoretisches Maß gesteuert. Das Ergebnis ist ein Netzwerk von Beschreibungen, das für die Klassifikation neuer Objekte eingesetzt werden kann. Nachteil des Verfahrens ist, daß die Repräsentation auf der Basis der selbst generierten Merkmalen nicht verständlich ist - sie ist vergleichbar der Repräsentation in einem neuronalen Netz.

Stier [1991] ermittelt Beschreibungen für einfache geometrische Objekte aus Bildbeispielen. Er nutzt dabei den Sprachformalismus von Ammon [1987]. Die Grundannahme des Ansatzes von Stier ist ähnlich dem hier in der Arbeit vorgestellten. Er geht davon aus, daß es unmöglich ist, alle möglichen Objekte und Relationen a priori vorzugeben, daher gibt er umgekehrt eine Methode vor, mit der flexibel auf neue Gegebenheiten reagiert werden kann. Das heißt er definiert elementare Grundstrukturen, aus denen sich komplexe Konzepte zusammensetzen (die Möglichkeiten des Zusammensetzens sind auch im Grundwissen vorgegeben). Stier interpretiert neue Objekte auf der Basis des vorhandenen Wissens. Die Beispiele dazu, die hypothetisierten Zusammenhänge zu bestätigen oder zu verwerfen. Die Auswertung geschieht in einem erschöpfenden Suchverfahren. Stier zeigt seine Strategie anhand des Lernens einfacher geometrischer Objekte (Viereck, Dreieck, Quadrat), welche aus elementaren Merkmalen und Relationen (z.B. Parallelität, Rechtwinkligkeit) abgeleitet werden.

Connell & Brady [1985] berichten von einem Experiment, Objektbeschreibungen aus realen Daten zu lernen. Das Programm lehnt sich an Winston's Arch an. Die Daten werden in einem semantischen Netz repräsentiert. Das Lernverfahren nutzt nur positive Beispiele und ermittelt neue Konzepte auf der Basis der 'dropping condition'-Heuristik: wenn zwei Dinge zur gleichen Klasse gehören, so müssen die Unterschiede zwischen ihnen irrelevant sein, und diese Unterschiede werden gelöscht. Die erzeugte Beschreibung ist zum einen stabil, aber auch sensitiv, d.h. detailliert. Dies wird durch eine hierarchische Beschreibung erreicht: grobe Levels garantieren die Stabilität, während feine die Bewahrung von Details gewährleisten.

Wong & You [1985] führen das Konzept des *random graphs* (Zufalls-Graphen) ein. Hiermit läßt sich die Variabilität einer strukturierten Beschreibung erfassen. Am Beispiel der Buchstabenerkennung wird das Clustering mit Hilfe solcher Graphen demonstriert. Vorgabe ist eine Sammlung von verrauschten Buchstaben. Die Beispiele werden jeweils als Graph dargestellt; im Lernschritt werden die Wahrscheinlichkeiten für die Attribute und Relationen ermittelt.

Teil III

Automatische Ableitung von Modellen aus Beispielen

7 Überblick über das realisierte Verfahren

Die Verknüpfung der Disziplinen *Maschinelles Lernen* (ML) und *Bildverstehen* ist noch sehr jung. Von verschiedensten Seiten wird sie allerdings als sehr vielversprechend angesehen - sozusagen als **die** Grundlage für automatische Interpretationssysteme [Strat 1994], [Bhanu & Poggio 1994]. Denn lernende Systeme verfügen über die Möglichkeit, ihr Wissen und Verhalten äußeren Gegebenheiten anzupassen. Hiermit wird erreicht, daß ein System bereits mit einem eingeschränkten Wissensvorrat operieren kann. Diesen kann es im Laufe der Interpretation unter Einfluß der Umgebung sukzessive erweitern. Noch gibt es sehr wenige Ansätze, Lernverfahren im Bildverarbeitungsbereich einzusetzen, daher sind die Erfolge bislang auch beschränkt. Momentane Ansätze zeichnen sich dadurch aus, daß verschiedenste Wege beschritten werden, während gleichzeitig viele noch unerforscht sind. Bhanu & Poggio [1994] formulieren grundsätzliche Voraussetzungen für den Einsatz solch neuer Techniken: so ist für eine gegebene Aufgabe des Bildverstehens das eigentliche Lernproblem herauszustellen; die Repräsentation für die Ein- und Ausgabedaten muß festgelegt werden, anschließend muß aus der Fülle von ML-Ansätzen ein für das Problem adäquater Algorithmus ausgewählt werden.

Auf der Basis der Grundlagen des zweiten Teils soll nun vertieft dargestellt werden, wie diese Kenntnisse genutzt werden können, das komplexe Problem der automatischen Modellgenerierung für die Bildanalyse anzugehen. Insbesondere soll geklärt werden, welcher Art die Vorgaben in diesem Problembereich sind, und welche der angesprochenen Hilfsmittel und Techniken der Repräsentation und des Lernens hierfür geeignet sind.

Vorliegende Arbeit orientiert sich an der menschlichen Informationsaufnahme durch Wahrnehmung und versucht dies in einem System zu simulieren. Von Objekten, welche visuell in Form von Beispielen präsentiert werden, sollen automatisch Modelle abgeleitet werden. Ziel ist ein Programm, welches Objekte "sieht" und diese in eine interne Repräsentation, ein Modell überführt. Die Modellierung berührt verschiedene Aspekte und nutzt daher verschiedene Techniken: Das System muß über "Wahrnehmungsfähigkeiten" verfügen, um bestimmte Objekte oder Objektteile unmittelbar zu erkennen. Visuell präsentierte Daten - welche lediglich Bilder, uninterpretiertes Wissen darstellen - können in echtes (internes) Wissen überführt werden, indem sie in die Sprache des Systems übersetzt werden. Darüber hinaus muß das System "lernfähig" sein, um aus Gesehenem in Verbindung mit einer angegebenen Bezeichnung neues Wissen abzuleiten. Die Begriffe sehen und lernfähig sind apostrophiert, da sie im Zusammenhang mit Maschinen verwendet werden, und bei weitem nicht die Qualität menschlicher Fähigkeiten erreichen - allerdings zumindest im Ansatz die damit vermittelten Fähigkeiten aufweisen sollen.

Wie aus der Betrachtung der menschlichen visuellen Wahrnehmung deutlich wurde (siehe Kapitel 2), ist diese grundsätzlich natürlich abhängig von der direkten äußeren Information, die es aufzunehmen gilt. Sie ist aber nicht möglich ohne zusätzliches Wissen im erkennenden System. Erkennen erfolgt also immer ausschließlich im Spiegel des vorhandenen Wissens. Die aus dem Kapitel über Wahrnehmung gewonnen Erkenntnisse legen zudem nahe, daß ein Modellierungssystem inkrementell arbeiten sollte. Hiermit kann der Tatsache Rechnung getragen werden, daß das System auf neue Gegebenheiten flexibel reagieren kann. Solche Gegebenheiten ergeben sich zum einen durch eine sich ändernde Umwelt, aber vor allem auch durch Änderungen im System (-wissen) selbst. Liegt ein reicheres Wissen vor, so können andere Dinge erschlossen werden. Wichtig ist, daß die Merkmale, die das jeweilige System für die Beschreibung benutzt, *beobachtbar* sein müssen. Auch müssen verschiedene Objekte bezüglich der zur Verfügung stehenden Merkmale unterscheidbar sein.

In dieser Arbeit sollen die Objektmodelle durch Lernverfahren bestimmt werden. Prinzipiell lassen sich die verschiedensten Methoden für dieses Problem heranziehen (vgl. Kapitel 5). So zeigen Konnektionistische Ansätze beachtliche Erfolge in der automatischen Bildinterpretation. Allerdings wird bei diesen Verfahren keine explizite Modellbestimmung erreicht, vielmehr bleibt diese "versteckt" im Netz. Insbesondere dieses Kriterium der Verständlichkeit führte zum Ausschluß neuronale Netze. Einige Lernverfahren nutzen höhere Beschreibungssprachen zur Formulierung der Ein- und Ausgabe. Diese große Ausdrucksfähigkeit des Systems wird allerdings

erkaufte durch eine höhere Komplexität der hierfür anwendbaren Algorithmen, die alle auf der Basis von Suchverfahren arbeiten. Verfahren wie Winston's Arch setzen zudem ein großes Quantum an Hintergrundwissen voraus, welches a priori zur Verfügung gestellt werden muß. Programme geringerer Komplexität wie ID3 oder Relax arbeiten mit einfacherer Beschreibungssprache und haben somit den Nachteil geringerer Ausdrucksfähigkeit. Allerdings zeichnen sie sich durch eine große Einsatzbreite und Effizienz aus.

Um diesen Konflikt zwischen Ausdrucksfähigkeit und Effizienz aufzulösen, wird das Lernproblem in vorliegender Arbeit in ein komplexeres Erkennungs- und Interpretationssystem integriert. Dadurch konnte eine Spezifizierung und damit eine Vereinfachung des Lernproblems erreicht werden.

Die Objekte, für die eine Beschreibung gesucht wird, werden in einem objektorientierten Ansatz repräsentiert und verwaltet. Sie werden in ihrer Struktur als fix angenommen. In dieser allgemeinen Struktur hat jedes Objekt Eigenschaften und es steht mit anderen Objekten in Beziehungen; eine vorgegebene Anzahl von Methoden garantiert seine Funktionalität. Einzelne Objektklassen unterscheiden sich durch ihre Eigenschaften bzw. in ihren Relationen und Methoden. Zur Bestimmung individueller Objektmodelle müssen somit lediglich die Objekteigenschaften gelernt werden, ebenso die Merkmale, die den Relationen zugrundeliegen. Damit reduziert sich das Problem der Objektmodellierung auf den (einfacheren) Fall der Merkmalsselektion. Dies muß dennoch keine Einschränkung bedeuten. Die Objekteigenschaften sind sehr allgemein gehalten und erweiterbar. Außerdem können Objekte beliebig viele Relationen zu anderen Objekten haben, die ebenfalls durch Lernen spezifiziert werden können. Die zur Verfügung stehenden Methoden dienen der Strukturierung der Objekte, nämlich der Erzeugung neuer Objektklassen und -instanzen, sowie der Aggregation von Objekten.

Durch das Einbetten des Lernens in ein Objektmodellierungssystem ergeben sich folgende Vorteile:

- Einfache Beschreibungssprache in Form von Attribut-Wert-Listen im Lernprozeß.
- Nutzen des Lernverfahrens ID3.

Das System versucht, in einem Wechselspiel zwischen Interpretation und Lernen nach und nach das Gesehene zu interpretieren. Auf diese Weise wird eine strukturierte Szenenbeschreibung erreicht: die visuelle Information wird mit Hilfe des vorhandenen Wissens klassifiziert. Stehen noch keine Objektmodelle zur Verfügung, so werden diese mittels Lernen ermittelt und können anschließend sofort für die weitere Interpretation genutzt werden.

Die Lernkomponente zur Erweiterung des Systemwissens kann in verschiedenen Situationen eingesetzt werden:

Erzeugung von Objektbeschreibungen: Die Beschreibung besteht aus den Merkmalen, in denen sich das Objekt gegen alle anderen abgrenzt. Zur Objektbeschreibung gehört zudem ein prototypisches Objekt, welches alle Eigenschaften und Relationen seiner Klasse umfaßt. Diese Beschreibung wird ermittelt, indem Beispiele und Gegenbeispiele für eine Objektklasse angezeigt werden. Das Lernverfahren selektiert aus den Merkmalen dieser Beispiele die für die Unterscheidung relevanten.

Erzeugung neuer Methoden: Jedes Objekt verfügt über Möglichkeiten zur **Strukturierung**, indem es neue Objektklassen ableiten, bzw. eine Gruppierung mit anderen Objekten durchführen kann. Jedoch liegt zunächst nicht fest, welche Eigenschaften für diese Strukturierung wichtig sind. Welche Objektteile beispielsweise ein komplexes Objekt bilden, kann erst im Laufe der Interpretation durch Analyse der entsprechenden Merkmale ermittelt werden.

Das System startet mit elementar beobachtbaren Objekten (siehe Kapitel 9.1.2), wobei es deren Eigenschaften und Relationen als Merkmale nutzt. Diese werden zur Bestimmung von neuen Objektbeschreibungen verwendet. Neu hinzukommende Objekte bzw. Eigenschaften oder Methoden von Objekten werden dabei direkt aufdatiert, so daß das System inkrementell sein Wissen erweitert. Die Implementierung erfolgte in der Sprache POP11 [Barrett & Sloman 1985]. Diese verfügt über die Möglichkeit, während des Programmablaufs Programmcode selbst zu erzeugen. Hierdurch wird die automatische Erzeugung neuer Objektklassen während des Programmablaufs möglich. Die unmittelbare Verfügbarkeit aller Erweiterungen wird durch das Interpreterprinzip der Sprache gewährleistet. Die während des Programmablaufs erzeugten neuen Funktionen sind durch einfaches Hinzuladen (kein Übersetzungsvorgang nötig) nutzbar.

Das System kennt somit (siehe Abbildung 7.1):

- Methoden zur Informationsaufnahme und Interpretation,
- Methoden zur Erweiterung des Wissens mittels Lernverfahren,

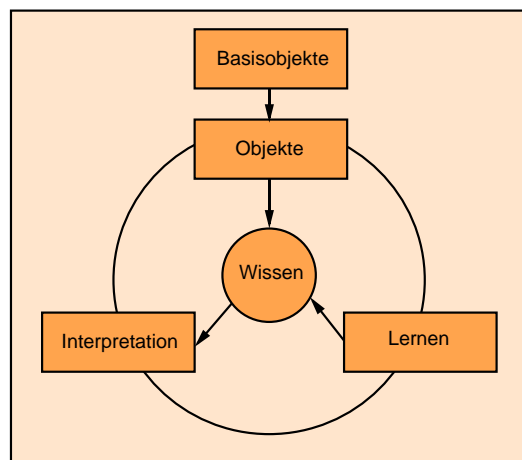


Abbildung 7.1: Objektmodellierungssystem mit Interpretations- und Lernkomponente

Zusammenfassend wird mit dem System zweierlei erreicht: zum einen Überführung der visuellen Information in eine interne Darstellung. Dieser Prozeß entspricht einer Übersetzung des Gesehenen in die System“sprache“. Andererseits wird aber auch eine inhaltliche (semantische) Änderung erzielt: während die präsentierten Beispiele lediglich ein einzelnes Objekt beschreiben, haben die abgeleiteten Modelle einen globalere Gültigkeit und können zur weiteren Interpretation genutzt werden.

Die Problematik lehnt sich an das menschlichen Vorbild an, erfordert jedoch - aufgrund ihrer Komplexität - starke Vereinfachungen.

- Im Gegensatz zum menschlichen Verhalten wird nur sichtbare (visuelle) Information genutzt. Es geht also kein globales Allgemeinwissen, wie es der Mensch ständig und unbewußt nutzt, mit ein - also insbesondere keine Erwartungen oder detailliertes Vorwissen. In vorliegendem Beispiel erfolgt darüber hinaus noch eine Konzentration auf reine Geometrie-Information. Das bedeutet, Farb- oder Texturhinweise werden ignoriert.
- Das Lernen erfolgt auf der Basis idealer Beispiele. Ideal in dem Sinne, daß sie die Objekte vollständig darstellen. Auch werden lediglich für diese Objektklasse typische Objekte verwendet. Dies bedeutet, daß für den Lernschritt keine Segmentierungsergebnisse verwendet werden (die ja meist unvollständig sind), sondern digitalisierte, vollständige Beschreibungen. Damit unterscheidet es sich auch vom menschlichen Lernen, welches in natürlicher Umgebung vor sich geht, wo Verdeckungen, Unvollständigkeit oder auch falsche Information vorkommen können. An dieser Stelle wird der ideale Regelkreis, welcher sich immer wieder an neue Situationen adaptiert, durch starke Vorinformation gestärkt. Unvollständige Information bedarf häufigerer Iterationen im Anpassungszyklus, sowie Verfahren, die es erlauben, frühere “Wahrheiten“ zu verwerfen.
- Die Objekte, an welchen das Verfahren demonstriert wird, sind einfach. Komplexe Objekte erfordern prinzipiell ähnliche Vorgehensweisen, unterscheiden sich jedoch im Detail stark.
- Es handelt sich um zwei-dimensionale Objekte.

Das Ergebnis des Interpretations- und Lernverfahrens bildet eine komplette Beschreibung der dargestellten Szene. Diese Beschreibung ist zum einen spezifisch, indem alle individuellen Objekte mit ihren jeweiligen Eigenschaften aufgezählt werden. Andererseits ist sie aber auch generisch. Denn sie enthält Beschreibungen von Objektklassen, sowie deren allgemeine Eigenschaften und Methoden. Darin liegt der große Vorteil des Verfahrens. Solche generischen Beschreibungen können sehr sinnvoll für die weitere Bildinterpretation eingesetzt werden (siehe Kap. 9.6).

Das System ID3 wird in vorliegender Arbeit als Basisverfahren benutzt, um iterativ neue Konzepte für die Bildinterpretation zu lernen. Daher wird dieses System zunächst ausführlich dargestellt und die implementierten Erweiterungen vorgestellt. Danach werden die Objekte sowie ihre Grundfunktionalitäten (Eigenschaften) definiert. Ihre Repräsentation und Verarbeitung erfolgt in einem objektorientierten Ansatz.

8 Lernverfahren ID3 mit Erweiterungen

ID3 geht von klassifizierten Beispieldaten aus, die in Form von Attribut-Wert-Listen gegeben sind. Jedes Beispiel besteht aus einer Liste von Merkmalen. Für jedes der Beispiele ist eine Klassenzugehörigkeit vorgegeben. Die Aufgabe besteht darin, eine Minimalkombination von Merkmalsausprägungen zu suchen, die ausreicht, um eine Klassifizierung durchzuführen. Oft genügen hierzu schon einige wenige Merkmale aus einer großen Merkmalsliste. Das Ergebnis ist ein Entscheidungsbaum. Dieser kann anschließend genutzt werden, neue, bisher unbekannte Beispiele zu klassifizieren. Die extensionale Beschreibung in Form der Beispiele gibt nur Information über den gegebenen Datensatz, während die abgeleitete intensionale Beschreibung danach idealerweise auf alle beliebigen Fälle anwendbar ist.

8.1 Strategie und Beispiel

In einem Top-Down-Verfahren wird der Entscheidungsbaum sukzessive aufgebaut. Auf jeder Ebene entscheidet eine Bewertungsfunktion, welches Merkmal die Beispielmenge am stärksten diskriminiert. Diese Unterteilung wird iterativ so lange durchgeführt, bis die einzelnen Knoten nur noch Beispiele einer Klasse enthalten. In jedem Knoten wird das Merkmal gewählt, welches die Beispielmenge so unterteilt, daß jede dieser Mengen möglichst viele Beispiele einer Klasse und möglichst wenige der anderen Klassen enthält. Die Bewertungsfunktion basiert auf einem informationstheoretischen Ansatz. Hintergrund für dieses Maß der Einfachheit ist die Vorstellung, daß ein einfacher Baum potentiell für mehr ungesehene Beispiele gültig ist, als ein stark verzweigter Baum.

Auf jeder Ebene wird also das Merkmal gesucht, welches den größten Informationsgewinn - bzw. die einfachste (kürzeste) Beschreibung - liefert. Diese wird über die Entropie (Einheit: bit) berechnet, welche ein Maß für die gewonnene Information, bzw. für die beseitigte Unsicherheit ist. Sie läßt sich aus der Selbstinformation I über die Wahrscheinlichkeit ermitteln. Die Entropie H stellt dabei den Erwartungswert der Selbstinformation, bzw. die gemittelte Selbstinformation dar ([Förstner 1989]).

$$I = -\log_2 p_a \quad (8-1)$$

$$H_i = -p_a \log_2 p_a \quad (8-2)$$

$$p_a = \frac{Anz_{a;i}}{Anz_{a;alle}} \quad (8-3)$$

mit:

H_i Entropie der Merkmalsausprägung i

p_a : Wahrscheinlichkeit, daß ein Beispiel zur Klasse a gehört

$Anz_{a;i}$: Anzahl der Beispiele die zur Klasse a gehören und bezüglich des betrachteten Attributes die Ausprägung i besitzen

$Anz_{a;alle}$ Anzahl aller Beispiele mit Ausprägung i .

Die Gesamtentropie eines Merkmals berechnet sich als gewichtetes Mittel aller Einzelentropien der jeweiligen Merkmalsausprägungen über folgende Beziehung:

$$\begin{aligned} H_{Merkmal} &= \sum_{i=1}^m p_i * H_i \\ &= \sum_{i=1}^m \frac{Anz_i}{Anz_{alle}} * H_i \end{aligned} \quad (8-4)$$

mit:

m : Anzahl der Ausprägungen eines Merkmals

p_i : Wahrscheinlichkeit, daß ein Beispiel die i -te Ausprägung des betrachteten Merkmals annimmt

Anz_i : Anzahl aller Beispiele mit Merkmalsausprägung i

Anz_{alle} : Anzahl aller Beispiele

Damit ergibt sich der Informationsgewinn (IG) für jedes Merkmal zu:

$$IG = H_{apriori} - H_{Merkmal} \quad (8-5)$$

An folgendem Beispiel (vgl. [Pao 1989]) soll ID3 beschrieben werden. Gegeben ist eine Menge von Beispielen, die eine Klassifizierung von Personen in zwei Klassen a und o angeben.

Größe	Haarfarbe	Augenfarbe	Klasse
groß	dunkel	blau	a
klein	dunkel	blau	a
groß	blond	blau	o
groß	rot	blau	o
groß	blond	braun	a
klein	blond	blau	o
klein	blond	braun	a
groß	dunkel	braun	a

Zunächst wird die Beschreibungslänge der Beispiele ermittelt. Zur Berechnung dieser a priori Beschreibungslänge benötigt man die Wahrscheinlichkeiten der Klassen. Sind beide Klassen gleich wahrscheinlich, führt dies zu einer Beschreibungslänge von 1 bit:

$$\begin{aligned}
 p_a &= 0.5 \\
 p_o &= 0.5 \\
 -0.5 \log_2(0.5) - 0.5 \log_2(0.5) &= 1 \text{ bit}
 \end{aligned}$$

Ist allerdings - wie hier - durch die Beispiele schon Vorinformation gegeben, so kann eine Klasse wahrscheinlicher sein, als eine andere: von den 8 Beispielpersonen gehören 5 zur Klasse a , 3 zur Klasse o , daher ist die Wahrscheinlichkeit, daß eine unbekannte Person zur Klasse a gehört, etwas höher.

$$\begin{aligned}
 H &= -p_a \log_2 p_a - p_o \log_2 p_o \\
 H &= -\frac{5}{8} \log_2 \left(\frac{5}{8}\right) - \frac{3}{8} \log_2 \left(\frac{3}{8}\right) \\
 H &= 0.954 \text{ bit}
 \end{aligned}$$

Für alle Ausprägungen der Merkmale Augenfarbe (blau, braun, grün), Haarfarbe (braun, rot, blond) und Größe (groß, klein) wird dann die Beschreibungslänge ermittelt, die sich ergibt, wenn zunächst nach diesem Merkmal klassifiziert wird. Eine kürzere Beschreibungslänge impliziert, daß eine Klasseneinteilung mit diesem Merkmal schneller möglich ist.

Eine Klassifikation nach dem Merkmal "Größe" teilt die Beispielmenge in zwei Untermengen (groß und klein). Würde hiermit bereits eine Klassifizierung in die Klassen a und o erreicht, so ergäbe sich ein maximaler Informationsgewinn für dieses Merkmal.

Die Klassifikation nach dem Merkmal "Größe" ergibt allerdings folgendes Bild:

groß	blond	braun	a
groß	rot	blau	o
groß	dunkel	blau	a
groß	blond	blau	o
groß	dunkel	braun	a

Entropie des Merkmals "groß":

$$\begin{aligned}
 H_{\text{groß}} &= -\frac{3}{5} \log_2 \left(\frac{3}{5}\right) - \frac{2}{5} \log_2 \left(\frac{2}{5}\right) \\
 &= 0.971 \text{ bit}
 \end{aligned}$$

klein	blond	blau	o
klein	dunkel	blau	a
klein	blond	braun	a

Entropie des Merkmals "klein":

$$\begin{aligned}
 H_{\text{klein}} &= -\frac{2}{3} \log_2 \left(\frac{2}{3}\right) - \frac{1}{3} \log_2 \left(\frac{1}{3}\right) \\
 &= 0.918 \text{ bit}
 \end{aligned}$$

Mit Gleichung 8-4 ergibt sich die Entropie der Entscheidung "Größe" als gewichtetes Mittel der Einzelentropien zu:

$$H_{\text{größe}} = \frac{5}{8} 0.971 \text{ bit} + \frac{3}{8} 0.918 \text{ bit} = 0.951 \text{ bit}$$

Der Informationsgewinn gegenüber der ursprünglichen Beschreibung

$$H_{apriori} - H_{größe} = 0.954 \text{ bit} - 0.951 \text{ bit} = 0.003 \text{ bit}$$

ist damit gering. Dies kann auch schon an der Einteilung der Beispielmenge durch dieses Merkmal abgelesen werden: hierdurch entstanden keine einheitlichen Klassen.

Für die übrigen Merkmale wird dieser Prozeß ebenfalls durchgeführt.

Haarfarbe:

klein	dunkel	blau	a
groß	dunkel	blau	a
groß	dunkel	braun	a

groß	rot	blau	o
------	------------	------	---

groß	blond	blau	o
klein	blond	blau	o
groß	blond	braun	a
klein	blond	braun	a

Zwei Werte des Merkmals Haarfarbe klassifizieren die Beispielmenge bereits eindeutig: Dunkelhaarige gehören zur Klasse *a*, Rothaarige zur Klasse *o*, lediglich für Blonde muß eine weitere Entscheidung getroffen werden.

Damit ergibt sich die Entropie der Entscheidung "Haarfarbe" zu:

$$H_{dunkel} = -\frac{3}{3} \log_2\left(\frac{3}{3}\right) - 0 = 0 \text{ bit}$$

$$H_{rot} = 0 \text{ bit}$$

$$H_{blond} = -\frac{2}{4} \log_2\left(\frac{2}{4}\right) - \frac{2}{4} \log_2\left(\frac{2}{4}\right) = 1 \text{ bit}$$

$$H_{haarfabe} = \frac{3}{8} * 0 \text{ bit} + \frac{1}{8} * 0 \text{ bit} + \frac{4}{8} * 1 \text{ bit} = 0.5 \text{ bit}$$

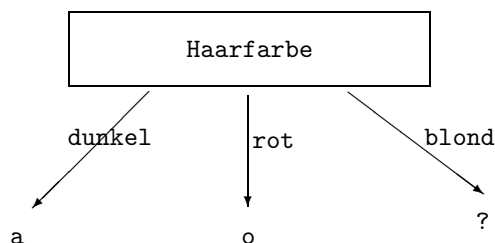
Der Informationsgewinn gegenüber der ursprünglichen Beschreibung ergibt sich somit zu

$$H_{apriori} - H_{haarfabe} = 0.954 \text{ bit} - 0.5 \text{ bit} = 0.454 \text{ bit}$$

Analog wird schließlich noch der Informationsgewinn für das Merkmal "Augenfarbe" berechnet. Er beträgt 0.347 bit. Damit liegt maximaler Informationsgewinn für das Merkmal Haarfarbe vor. Nach diesem muß also zunächst unterschieden werden, es bildet einen ersten Knoten im Entscheidungsbaum.

```

if (Haarfabe = dunkel) then
    Klasse = a
elseif (Haarfabe = rot) then
    Klasse = o
elseif (Haarfabe = blond) then
    .....
endif
    
```



Auf der nächsten Ebene müssen die entstandenen Beispielmengen jeweils gesondert nach dem gleichen Prinzip weiteruntersucht werden¹⁶. Dies wird so lange durchgeführt, bis alle Beispiele eindeutig klassifiziert sind und die Untermengen nur noch Beispiele einer Klasse enthalten. In obigem Beispiel betrifft dies lediglich die Menge (4 Beispiele), die durch eine Abfrage nach blonder Haarfarbe gewonnen wurde. Nun sind nur noch die Merkmale Augenfarbe und Größe verwendbar.

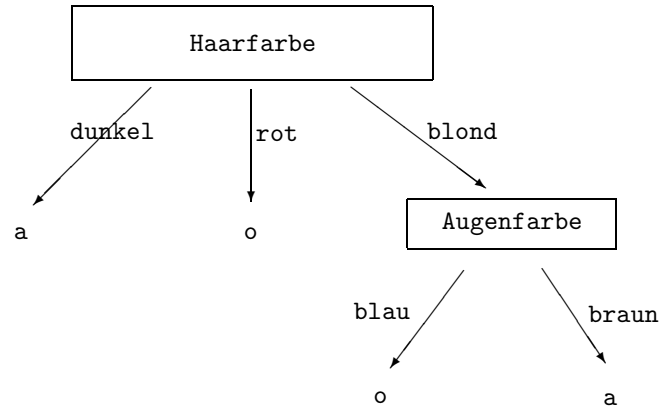
Als Ergebnis erhält man folgenden Entscheidungsbaum:

¹⁶Dabei ist zu beachten, daß das erste Merkmal (die Haarfarbe) jetzt nicht mehr zur Verfügung steht.

```

if (Haarfarbe = dunkel) then
  Klasse = a
elseif (Haarfarbe = rot) then
  Klasse = o
elseif (Haarfarbe = blond) then
  if (Augenfarbe = blau) then
    Klasse = o
  elseif (Augenfarbe = braun) then
    Klasse = a
  endif
endif
endif

```



Offensichtlich ist das Merkmal “Größe” unerheblich für die Klassifizierung der Personen in die Klassen *a* und *o*. Lediglich Abfragen nach Haar- und Augenfarbe sind nötig. Eindeutig ist auch, daß diese neue Repräsentationsform für eine Klassifizierung in die beiden Klassen kürzer ist als die ursprüngliche extensionale Beschreibung durch die Beispiele. Hinzu kommt, daß dieser Entscheidungsbaum verwendet werden kann, um neue Beispiele zu klassifizieren - eine Möglichkeit, die mit der ursprünglichen, extensionalen Beschreibung nicht direkt gegeben war. Diese stellte lediglich eine implizite Beschreibung in Form einer Beispielsammlung dar, und nicht in Form von klassifizierenden Merkmalen.

8.2 Defizite und Erweiterungen

Die Vorteile von Klassifikatoren, die auf der Basis von Entscheidungsbäumen arbeiten, sind folgende (vgl. u.a. [Walker & Moore 1988], [Kodratoff 1994])

- Der Entscheidungsbaum gibt einen guten Einblick in die innere Struktur der Daten. Zudem wird eine gewisse Hierarchisierung durchgeführt, bei der ähnliche Objekte in den gleichen Teilbäumen angesiedelt sind. Ein Zusammenfassen von Blättern im Baum entspricht damit einer Generalisierung.
- Das Ergebnis ist effizient, indem im Entscheidungsbaum nur diejenigen Objektmerkmale genutzt werden, die für eine eindeutige Klassifizierung nötig sind. Alle anderen müssen nicht berechnet werden.
- Das zugrundeliegende Bewertungsmaß der Entropie ist anschaulich und überzeugend. Es läßt sich für sehr viele Fragestellungen einsetzen (vgl. z.B. [Förstner 1991], [Vosselman 1992], [Leclerc 1988]).
- Leichte Lesbarkeit des ermittelten Ergebnisses - solange der Baum nicht zu verzweigt ist.

Das Verfahren basiert darauf, daß auf jeder Ebene das beste Merkmal ausgewählt wird. Damit entspricht die Abarbeitung einer Best-First-Suche und erkaufte sich somit das Risiko, eine optimale Lösung zu übersehen. Damit wird eine einfache Komplexität des Algorithmus erreicht: Für *k* Merkmale mit je *j* Ausprägungen und einen Baum mit *l* Ebenen errechnet sich die Komplexität zu kj^l . Sie wächst also exponentiell mit den Ebenen im Baum, allerdings nur linear mit der Anzahl der Merkmale und deren Ausprägungen. Die Größe *l* ist meist klein, zudem reduzieren sich auf jeder Ebene die Anzahl der Merkmale, was zu einem günstigeren Verhältnis führt [Pao 1989].

Das hier in seiner ursprünglichen Form beschriebene Verfahren hat jedoch auch Nachteile. Dies führte dazu, daß das System einige Erweiterungen erfahren hat:

- Die Sprache läßt nur Vergleiche mit Konstanten oder mit Merkmalen zu, allerdings keine Vergleiche unter den Merkmalen, wie z.B. “Augenfarbe = Haarfarbe”. Ebenfalls sind keine logischen Operationen folgender Art möglich: “Haar = braun und Größe = 188”. Der Nutzer muß daher die Daten so geschickt aufbereiten, damit sie von ID3 verarbeitet werden können. Das bedeutet die Transformation der Daten in die Attribut-Wert-Liste und eventuelle logische oder mathematische Vorberechnungen.

- Das System beruht auf konsistenten und rauschfreien Daten. Jedes Beispiel muß für jedes Merkmal eine bestimmte Ausprägung aufweisen. Mit falschen oder widersprüchlichen Daten kann ID3 nicht umgehen. Es existieren jedoch auch hier Erweiterungen, die die fehlenden Werte über Wahrscheinlichkeiten des entsprechenden Werts des Attribut für eine Klasse bestimmen.
- ID3 in seiner ursprünglichen Version kennt nur Symbole, es kann also lediglich diskrete Werte verarbeiten. Im Falle von numerischen Werten führte dies dazu, daß - falls nötig, nach jedem einzelnen Wert unterschieden wird, und keine Abfragen der Art “Größe ≥ 1.77 cm” möglich sind. Hier sind allerdings Verfahren entwickelt worden, die Daten durch Mittelung oder Clusteranalyse vorzuverarbeiten. Diese Erweiterung ist auch in vorliegender Arbeit realisiert worden (siehe Kapitel 8.2.1).
- Neben der Erweiterung auf kontinuierliche Attribute existieren laut Quinlan [1994, pers. Mitteilung] auch Versionen, die noch weitere Datentypen verarbeiten können z.B. Relationen oder auch Bäume. Das Programm FOIL [Quinlan 1990] kann relationale Strukturen verarbeiten und erzeugt als Ergebnis eine Regel in Form von Horn-Klauseln. Gegeben ist ein Zielkonzept, Beispiele hierfür, sowie Hintergrundwissen in Form von weiteren Regeln und Beziehungen, die zwischen den Beispielen gelten. Die gesuchte Regel wird dadurch erzeugt, daß nacheinander Klauseln hinzugenommen werden (basierend auf dem Hintergrundwissen), die für alle positiven Beispiele gelten, und für die negativen nicht. Als Maß für die Qualität der hinzugenommenen Klausel wird ebenfalls die Entropie genutzt. Dieses Maß ist allerdings die einzige Verwandtschaft zwischen den beiden Programmen. Bezüglich der Abarbeitung wird deutlich, daß alle möglichen Kombinationen von Klauseln in einem Suchverfahren abgearbeitet werden müssen - was geeignete Heuristiken erfordert. Die lineare Berechnungszeit ist hiermit allerdings nicht mehr gegeben.

Der Hinweis auf die Verwendung von Bäumen als Merkmal bezog sich auf eine Arbeit von Shapiro [1987]. Er beschreibt ein inkrementelles Verfahren zur Erzeugung von Schach-Regeln. Hierbei werden bestimmte Schachstellungen einer Klasse zugeordnet, und mit ID3 bewertet. Das Ergebnis (ein Entscheidungsbaum) wird wiederum als Eingabe für eine Bewertung auf einer nächsten Stufe verwendet. In diesem Sinne entsteht ein hierarchisches System, welches auf jeder höheren Stufe Ergebnisse der unteren verwendet. So gesehen stellt die Eingabe für ID3 jedoch nicht einen Baum im eigentlichen Sinne dar, sondern bleibt ein Attribut.

- Das Ergebnis ist immer ein Entscheidungsbaum, der unter Umständen recht verzweigt werden kann, da für jedes Beispiel eine richtige Klassifikation gesucht wird. Im Extremfall kann es dazu führen, daß für jedes Beispiel eine separate Verzweigung im Baum geschaffen werden muß. Solch ein Baum garantiert zwar eine richtige Klassifizierung der Beispieldaten, ist jedoch - aufgrund der sehr geringen Stichprobe in den einzelnen Unterteilungen - nicht sehr zuverlässig. Eine Erweiterung des Programms geht dahin, daß der Entscheidungsbaum an bestimmten Stellen abgeschnitten wird und an dieser Stelle eine prozentuale Angabe über die Klassifizierung an diesem Knotens gemacht wird. An einem Knoten, der 3 Beispiele für Klasse *a* und 1 Beispiel für Klasse *o* enthält, kann beispielsweise folgende Aussage stehen: zu 75 % Klassifizierung nach Klasse *a*, zu 25 % Klassifizierung in Klasse *o* [Quinlan 1987].

Meist ist allerdings ein buschiger Baum ein Zeichen dafür, daß in den Daten keine einheitliche Struktur zu finden war, daß es sich offensichtlich im einen inkohärenten Datensatz handelte. Es kann aber auch dafür sprechen, daß die Merkmale für die Charakterisierung der Objekte nicht günstig gewählt wurden, bzw. daß mit der gegebenen Sprache nicht ausdrückbar ist, was erzielt werden sollte.

- Bei sehr großen Datenmengen besteht die Möglichkeit, zunächst nur mit einem Bruchteil dieser Daten zu lernen, und mit dem Entscheidungsbaum die übrigen Daten zu klassifizieren. Beispiele die falsch klassifiziert werden, werden in einem zweiten Iterationsschritt in die Beispielmenge aufgenommen und es erfolgt ein erneutes Lernen. Üblicherweise sind nur 3-4 Iterationen nötig.

8.2.1 Erweiterung auf kontinuierliche Attribute

Die ursprüngliche Version von ID3 arbeitet lediglich auf diskreten Merkmalen. Jedes Element der Merkmalsliste wird dabei als potentielles Klassifikationskriterium angenommen. Für kontinuierliche Merkmale ist dieser Ansatz nicht sinnvoll, da verschiedene ähnliche Zahlen als Objektmerkmale auftreten können. Werden in obigem Beispiel anstelle der Attribute “groß” und “klein” die Körpergröße in cm angegeben, so ergäbe sich beispielsweise folgende Liste für die 8 Beispielpersonen:

Größe	Haarfarbe	Augenfarbe	Klasse
1.82	dunkel	blau	a
1.60	dunkel	blau	a
1.80	blond	blau	o
1.88	rot	blau	o
1.90	blond	braun	a
1.55	blond	blau	o
1.61	blond	braun	a
1.90	dunkel	braun	a

Im Extremfall würde ID3 dieses Größenmerkmal als Selektionskriterium heranziehen, da eine Klassifizierung mit ihm ja eindeutig ist! Dies reflektiert allerdings in keiner Weise die Realität. Es liegt daher nahe, die Größenattribute in einem Vorverarbeitungsschritt auf bestimmte Intervalle vorzuskalieren. Diese Intervalle werden anschließend als neue Merkmale benutzt.

Es wurden verschiedene Ansätze implementiert:

- Unterteilung am Mittelwert in zwei Wertebereiche. Damit entsteht ein Intervall, das Werte kleiner als der Mittelwert enthält, und eines dessen Werte größer als der Mittelwert sind.

[\leq Mittelwert] [$>$ Mittelwert]

- Das gleiche Prinzip kann auch angewandt werden auf eine Trennung am Median.
- Eine Trennung erfolgt in n gleichgroße Klassen um den Mittelwert. Die Anzahl der Klassen n wird vorher festgelegt (z.B. $n = 3, 4, \dots$).

Alle diese Ansätze werden für jedes kontinuierliche Merkmal durchgerechnet. Die Bewertung der Ansätze erfolgt im Hinblick auf das anschließende Lernverfahren. Hierzu wird das gleiche Bewertungsmaß - die Entropie - herangezogen. Dasjenige Cluster, welches die Klassen am besten unterteilt, wird ausgewählt. Die Unterteilung erfolgt also im Hinblick auf eine optimale Klassenunterteilung durch dieses Merkmal - und nicht auf eine optimale Clusterung der Merkmale selbst. Damit orientiert die Auswahl des besten Merkmals rein an dem sich anschließenden Lernprozeß und nicht an dem vorliegenden Datensatz.

Für obiges Beispiel ergeben sich folgende Unterteilungen:

Unterteilung am Mittelwert (zwei Merkmale):

$$\begin{aligned} [\leq 1.76][> 1.76] & : H : 0.95 \\ IG & = 0.954 - 0.95 = 0.004 \end{aligned}$$

Unterteilung am Median (zwei Merkmale):

$$\begin{aligned} [\leq 1.81][> 1.81] & : H : 0.90 \\ IG & = 0.954 - 0.90 = 0.054 \end{aligned}$$

Unterteilung in 3 Klassen (drei Merkmale):

$$\begin{aligned} [\leq 1.61][1.61 < x \leq 1.90][> 1.90] & : H : 0.95 \\ IG & = 0.954 - 0.95 = 0.004 \end{aligned}$$

Den maximalen Informationsgewinn liefert in diesem Fall die Unterteilung der Merkmale am Median. Die neu aufbereitete Beispieldatensatz lautet damit folgendermaßen:

Größe	Haarfarbe	Augenfarbe	Klasse
[$>$ 1.81]	dunkel	blau	a
[\leq 1.81]	dunkel	blau	a
[\leq 1.81]	blond	blau	o
[$>$ 1.81]	rot	blau	o
[$>$ 1.81]	blond	braun	a
[\leq 1.81]	blond	blau	o
[\leq 1.81]	blond	braun	a
[$>$ 1.81]	dunkel	braun	a

Wichtig für den Ansatz ist, daß die ursprünglichen Merkmale dennoch weiterhin beibehalten werden. Es kann der Fall auftreten, daß zunächst eine Unterteilung nach den Intervallen sinnvoll ist, später aber Beispielmengen entstehen, die sich nicht trennen lassen. Hier ist es nun möglich, durch Rückgriff auf die ursprünglichen Zahlenwerte immer noch eine Unterteilung zu erreichen.

8.2.2 Bewertung der Verzweigungen des Baums

Quinlans ID3 betrachtet nur den Informationsgewinn bezüglich einer Klassifizierung der Daten, nicht aber bezüglich des entstehenden Entscheidungsbaums selbst. Allein die Klassifizierungsfähigkeit eines Merkmals wird berücksichtigt, nicht aber der zur Beschreibung der Merkmale nötige Aufwand. Wäre beispielsweise für jedes Beispiel der jeweils zugehörige Eigenname als zusätzliches Merkmal spezifiziert, so würde dieses die Beispiele eindeutig klassifizieren - allerdings erforderte dies einen Entscheidungsbaum mit n (=Anzahl der Beispiele) Verzweigungen. Im ursprünglichen Programm gibt also keine "Bestrafung" von Knoten im Baum, welche viele Äste erzeugen (Merkmale mit vielen Ausprägungen). Quinlan [1986] schlägt daher vor, diesen negativen Effekt dadurch zu umgehen, indem nicht allein der Informationsgewinn der Merkmale bewertet wird, sondern auch noch die Information des Merkmals selbst. Diese Beschreibungslänge IV eines Merkmals errechnet sich aus:

$$\begin{aligned} IV(\text{Merkmal}) &= \sum_{i=1}^m H(i) \\ &= - \sum_{i=1}^m \frac{Anz_i}{Anz_{alle}} \log_2 \frac{Anz_i}{Anz_{alle}} \end{aligned} \quad (8-6)$$

Dieser Wert beträgt 0 für ein Merkmal mit nur einer Ausprägung und wächst mit der Anzahl der Merkmale an; d.h. stärkere Verzweigungen führen zu größeren Werten der Beschreibungslänge der Merkmale IV . Die Beschreibungslänge IV wird als eine Art von Gewicht eingeführt, und als neues Bewertungsmaß $Gain$ der Quotient aus IG und IV angesetzt¹⁷:

$$Gain = \frac{IG}{IV} \quad (8-7)$$

In obigem Beispiel betragen die Werte IG für die 3 Merkmale:

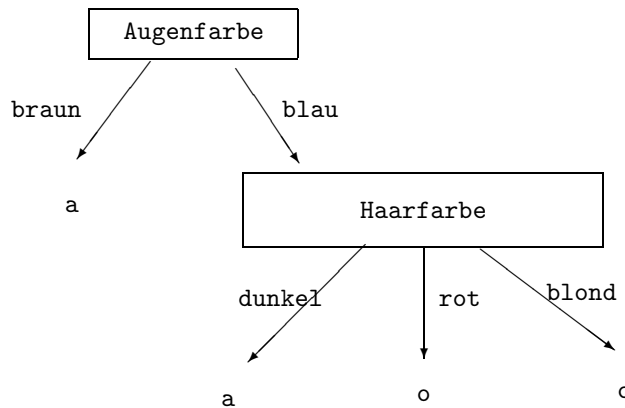
$$\begin{aligned} IG_{größe} &= 0.003 \text{ bit} \\ IG_{haarfärbe} &= 0.454 \text{ bit} \\ IG_{augenfärbe} &= 0.347 \text{ bit} \end{aligned}$$

Die Information der Merkmale und das neue Maß $Gain$ errechnen sich zu:

$$\begin{aligned} IV_{größe} &= -\frac{5}{8} \log_2 \left(\frac{5}{8}\right) - \frac{3}{8} \log_2 \left(\frac{3}{8}\right) = 0.954 \text{ bit} \Rightarrow \frac{0.003}{0.954} = 0.0031 \\ IV_{haarfärbe} &= -\frac{3}{8} \log_2 \left(\frac{1}{8}\right) - \frac{1}{8} \log_2 \left(\frac{3}{8}\right) - \frac{4}{8} \log_2 \left(\frac{4}{8}\right) = 1.406 \text{ bit} \Rightarrow \frac{0.454}{1.406} = 0.3230 \\ IV_{augenfärbe} &= -\frac{5}{8} \log_2 \left(\frac{5}{8}\right) - \frac{3}{8} \log_2 \left(\frac{3}{8}\right) = 0.954 \text{ bit} \Rightarrow \frac{0.347}{0.954} = 0.3637 \end{aligned}$$

Durch Berücksichtigung der Anzahl der Merkmalsausprägungen (und damit die Anzahl der nötigen Verzweigungen im Baum) ergibt sich eine Veränderung im Bewertungsmaß: offensichtlich hat hiermit das Merkmal "Augenfärbe" den größeren Informationsgewinn und wird somit als erstes Selektionskriterium herangezogen:

¹⁷Wobei der Fall daß $IV = 0$ wird natürlich abgefangen werden muß.



8.2.3 Erweiterung auf ungenaue Merkmale

Im Folgenden werden kurz mögliche Erweiterungen von ID3 bezüglich der Auswertung ungenauer bzw. fehlerbehafteter Merkmale angesprochen. Ungenaue Attribute können in zwei Fällen auftreten, die hier gesondert diskutiert werden: zum einen können sie im Trainingsdatensatz auftreten (Lernphase), aber auch erst im neu zu klassifizierenden Beispiel (Klassifizierungsphase).

Ungenaue Trainingsdaten

Diese Erweiterung behandelt den Fall, daß bereits das Lernverfahren mit ungenauen Merkmalen arbeiten muß. Für die Merkmale kann lediglich die Zugehörigkeit zu einem bestimmten Merkmal angegeben werden. So ist beispielsweise für die Diagnose einer Krankheit die Farbe der Rötung der Haut zu bestimmen. Diese kann in manchen Fällen als rot (70 %) oder aber auch als blau (30 %) angesehen werden. Weber & Zimmermann [1991] schlagen ein sogenanntes Fuzzy ID3 vor, um diesen Fall zu bearbeiten. Die Idee besteht darin, die Beispielmenge derart zu erweitern, daß ein Beispiel entsprechend seiner Merkmalsausprägung mehrfach auftreten kann. Es ist allerdings sicherzustellen, daß mit der Einführung neuer Beispiele keine Überschneidungen mit bereits vorhandenen Beispielen auftreten. Dies könnte der Fall sein, wenn die identischen Merkmale für zwei verschiedene Klassen sprechen. Zur Prozessierung muß der ursprüngliche Algorithmus nur geringfügig modifiziert werden, indem der Faktor p_i bei der Berechnung der Gesamtentropie geändert wird (vgl. Gleichung 8-4). Dieser Faktor berücksichtigt die Wahrscheinlichkeit einer Merkmalsausprägung. Anstelle der Anzahl der Beispiele mit dem jeweiligen Merkmal wird jetzt die jeweilige Zugehörigkeit zu einem Merkmal für die Berechnung des Gewichtsfaktors angesetzt.

Als Beispiel sei für obigen Fall angenommen, die Haarfarbe "rot" lasse sich gelegentlich mit "blond" verwechseln:

$$p(\text{rot}|\text{rot}) = 90\%$$

$$p(\text{blond}|\text{rot}) = 10\%$$

Die ursprüngliche Beispielmenge erweitert sich somit um eins (nur eine Versuchsperson hatte rote Haare):

Zugehörigkeit	Größe	Haarfarbe	Augenfarbe	Klasse
...
1.0	groß	blond	blau	o
0.9	groß	rot	blau	o
0.1	groß	blond	blau	o
1.0	groß	blond	braun	a
...

Im Programmablauf muß nun lediglich bei der Berechnung der Gesamtentropie die Wahrscheinlichkeit der Zugehörigkeit zur jeweiligen Merkmalsausprägung berücksichtigt werden.

Haarfarbe:

1.0	klein	dunkel	blau	a
1.0	groß	dunkel	blau	a
1.0	groß	dunkel	braun	a
3.0				

0.9	groß	rot	blau	o
0.9				

1.0	groß	blond	blau	o
1.0	klein	blond	blau	o
1.0	groß	blond	braun	a
1.0	klein	blond	braun	a
0.1	groß	blond	blau	o
4.1				

Damit ergibt sich die Entropie der Entscheidung "Haarfarbe" zu:

$$H_{dunkel} = -\frac{3}{3} \log_2\left(\frac{3}{3}\right) - 0 = 0 \text{ bit}$$

$$H_{rot} = 0 \text{ bit}$$

$$H_{blond} = -\frac{2}{5} \log_2\left(\frac{2}{5}\right) - \frac{3}{5} \log_2\left(\frac{3}{5}\right) = 0.971 \text{ bit}$$

$$H_{haarf Farbe} = \frac{3}{8} * 0 + \frac{0.9}{8} * 0 + \frac{4.1}{8} * 0.971 = 0.498 \text{ bit}$$

Der Informationsgewinn gegenüber der ursprünglichen Beschreibung beträgt somit

$$H_{apriori} - H_{haarf Farbe} = 0.954 - 0.498 = 0.456 \text{ bit}$$

Hiermit ergibt sich eine Entropie, die mit 0.456 bit etwas über der ursprünglichen (0.454) liegt. Dies liegt daran, daß die Klassifizierung der durch das Attribut blond gebildeten Menge nicht mehr gleichverteilt ist in *a* und *o*, sondern die Klasse *o* leicht bevorzugt.

In analoger Weise wird der gesamte Entscheidungsbaum erzeugt.

Ungenaue Testdaten

Diese Erweiterung bezieht sich auf die Nutzung des Entscheidungsbaums für die Klassifikation. Es wird angenommen, daß bei der Merkmalsbestimmung unbekannter Beispiele Fehler auftreten können [Quinlan 1987]. Gegeben ist das zu klassifizierende Beispiel. Für die Merkmale sind bestimmte Zugehörigkeiten bekannt:

Groesse: gross 100 %
 Haarfarbe: blond 100 %
 Augenfarbe: blau 30 % braun 70 %

Für den Kandidaten steht also fest, daß er groß ist und blondes Haar hat, allerdings kann seine Augenfarbe nur mit einer gewissen Unsicherheit festgestellt werden: zu 30 % hat er blaue, zu 70 % braune Augen. Der Entscheidungsbaum muß nun für alle Kombinationen der Attribute durchlaufen werden, wobei die Unsicherheiten mitgeführt werden müssen. Es ergibt sich schließlich eine Klassifizierung, die ebenfalls mit einer gewissen Unsicherheit behaftet ist. Aufgrund des Entscheidungsbaums gehört der Kandidat somit mit 70 % zur Klasse *a* und mit 30 % zur Klasse *o*.

Diese Erweiterungen wurden allerdings in der Arbeit nicht verwendet, da von idealen Daten ausgegangen wurde. Reale Daten erfordern allerdings ein "Robustifizieren" in diesem Sinne. Hiermit kann die Qualität bzw. Unsicherheit der Merkmalsextraktion berücksichtigt werden.

9 Objektorientierte Darstellung und Verarbeitung

Das entwickelte Programm stellt eine interaktive, inkrementelle Methode zur Wissensgewinnung zur Verfügung. Das System hat ein geometrisches Grundwissen über eine einfache 2D-Welt. Diese umfaßt Objekte vom Typ **Punkt**, **Linie** und **Fläche**, zu deren Manipulation und Analyse Routinen zur Verfügung stehen. Dieses vorgegebene Basiswissen bestimmt die Ausdrucksfähigkeit des Systems. Es kann daher mit einer Sprache verglichen werden, da es sowohl das Vokabular zur Beschreibung von Objekten und Relationen umfaßt, als auch die Grammatik, die dies geeignet und in sinnvoller Weise verknüpft. Die Sprache besteht aus den elementaren geometrischen Grundobjekten, sowie geometrischen und topologischen Beziehungen zwischen ihnen.

Die Sprache ist objektorientiert aufgebaut. Das System arbeitet dabei auf zwei Ebenen: die Objekte Punkt und Linie werden sehr effizient in einer Art relationaler Datenbankstruktur verwaltet, während Flächen (d.h. Polygone) **echt** objektorientiert dargestellt werden¹⁸. Diese Trennung ist insofern sinnvoll, als Punkte und Linien in vorliegender Arbeit lediglich als "statische" Bestandteile der Objekte angesehen werden, während alle "semantischen" Objekte vom Typ Fläche sind¹⁹. Mit dem objektorientierten Prinzip können diese somit als Unterklassen von Flächen ermittelt und somit effizient hierarchisch verarbeitet werden. Die Objektklasse **Polygon** verfügt über sie kennzeichnende Eigenschaften, sowie einen Satz an Methoden, welche an ihre Unterklassen vererbt werden.

Diese Objekte und ihre Methoden werden für die Szeneninterpretation und für den inkrementellen Wissenserwerb genutzt. Der Ansatz geht dabei davon aus, daß dem System eine Reihe von Beispielen präsentiert werden. Die Steuerung der Interpretation der Beispiele wird durch den sog. Inferenzmechanismus gesteuert. Diese Abarbeitungsstrategie ist in einer stark vereinfachten Form eines Blackboards (vgl. z.B. [Engelmore & Morgan 1988]) realisiert. Für diesen Ablauf der Interpretation ist die Vorstellung eines runden Tisches hilfreich, auf dem verschiedene Aufgaben bzw. Objekte verteilt sind. Um diesen sitzen Agenten, von denen jeder über bestimmte Fähigkeiten verfügt, mit denen er versucht, die Aufgaben zu lösen. Findet ein Agent ein passendes Objekt, so nimmt er es vom Tisch, führt die Aktion durch und legt das Ergebnis zurück. Dieses wiederum kann dann von einem weiteren Agenten benutzt werden (vgl. Abbildung 9.1).

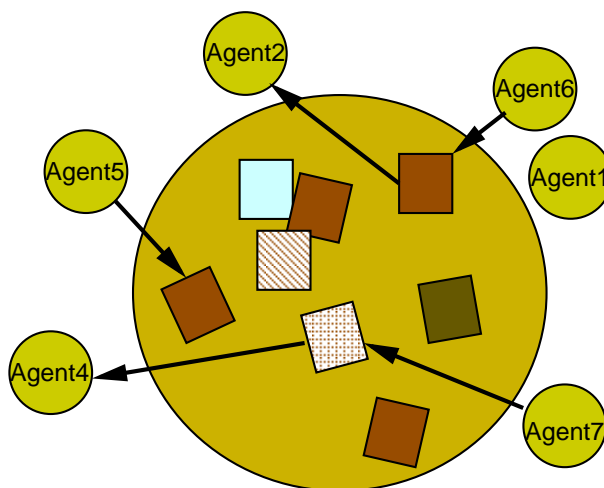


Abbildung 9.1: Blackboard, visualisiert anhand von Agenten um einen runden Tisch

Die flächenhaften Objekte in vorliegendem Fall stellen zum einen die Agenten dar, zum andern sind sie allerdings auch die Objekte, mit denen etwas geschieht - sie sind somit aktiv und passiv zugleich. Jedes Objekt hat bestimmte Möglichkeiten zur Manipulation anderer Objekte (und ist damit ein Agent) - auf der anderen Seite kann es von einem anderen Objekt genutzt werden. Jedes Objekt führt somit seine möglichen Aktionen aus und stellt das Ergebnis den anderen wieder zur Verfügung. Der Vorgang wird so lange durchgeführt, bis keine Veränderungen mehr erfolgen, d.h. kein Agent mehr etwas mit den Tischobjekten anfangen kann. Damit ist die

¹⁸Hierzu wurde die POP11-Erweiterung **Flavours** eingesetzt, welche sich an das am MIT entwickelte System Flavors [Canon 1980] anlehnt.

¹⁹Eine Erweiterung des Systems auf punkt- und linienhafte kartographische Objekte ist denkbar und dann sinnvoll, wenn Punkte und Linien ebenfalls unterschiedliche Bedeutung haben (z.B. Böschungslinien, Grenzlinien). Dann sollten diese ebenfalls objektorientiert abgelegt und verwaltet werden.

Interpretation durchgeführt: die nun noch vorhandenen Objekte stellen die gesuchte Szenenbeschreibung dar. Wichtig ist, daß diese Ablaufsteuerung von den Objekten selbst ausgeht. Diese führen ihre Möglichkeiten aus und verändern somit die Gegebenheiten, auf die die anderen Objekte ihrerseits wieder reagieren können. Somit läuft der Prozeß datengesteuert ab.

Allerdings hat der Nutzer Möglichkeiten der Interaktion. Zum einen spezifiziert er, was gelernt wird und gibt die Beispiele vor. Zum anderen kann er eingreifen, falls Inkonsistenzen im abgeleiteten Wissen auftreten. Somit steuert er die Revision des Wissens.

9.1 Basisobjekte, Objekteigenschaften und Methoden

Das Programm kennt die elementaren geometrischen Elemente Punkt, Linie und Fläche. Zu deren Definition und Erkennung stehen Routinen zur Verfügung (Kapitel 9.1.1). Auch für die Bestimmung und Manipulation der Relationen zwischen diesen Elementen sind Funktionen implementiert (Kapitel 9.1.2), wobei prinzipiell alle Kombinationen zwischen den Grundobjekten möglich sind. Einige dieser Relationen sind bereits implizit in den Datenbankkonstrukten abgelegt, beispielsweise die Inzidenz eines Punkts mit einer Linie. Während die meisten der Relationen als Funktionen im Programm realisiert sind, wurden die Relationen zwischen den flächenhaften Objekten in einer erweiterbaren Liste der Datenbank abgelegt (Kap. 9.1.3). Hiermit kann die Funktionalität des Systems einfach erweitert werden.

9.1.1 Definition der Basisobjekte

Die Objekttypen Punkt, Linie und Fläche werden in einer flexiblen POP11-spezifischen Datenbank verwaltet. Darunter ist nicht eine Datenbank im herkömmlichen Sinne zu verstehen²⁰. Sie ist eine besondere Speicherstruktur, auf die effizient zugegriffen werden kann. Die Datenbank ist als geschachtelte Liste angelegt. Jeder Eintrag stellt eine Unterliste dar, auf die mit einer bestimmten Kennung jeweils zugegriffen werden kann.

Punkte werden in relationaler Form mit ihren x/y-Koordinaten und der Punktnummer als Schlüssel abgelegt. Die Linien bestehen aus Anfangs- und Endpunkt, sowie ihrer Liniennummer als Identifikator. Der Zugriff auf Einzelobjekt erfolgt primär über den Identifikator, kann aber auch über alle anderen Listenelemente erfolgen. So können beispielsweise alle Punkte angesprochen werden, die die x-Koordinate 10 besitzen, bzw. alle Linien, die im Punkt **poi4** enden.

Punkte: Punkte werden in der Datenbank folgendermaßen dargestellt: **poi** definiert die Liste als Punkteintrag, **poi1** bezeichnet den Punktidentifikator, die folgende Liste [**10 100**] kennzeichnet x- und y-Koordinate des Punktes:

```
[poi poi1 [10 100]]
[poi poi2 [50 100]]
....
```

Während der Verarbeitung entstehen Zwischenobjekte, die ebenfalls in der Datenbank abgelegt werden. Besonders wichtig ist hier die Typisierung der Punkte, die separat von der Punktdefinition verwaltet wird. Diese Klassifizierung erfolgt aufgrund der Anzahl der Linien, die in einem Punkt zusammenkommen, sowie deren Richtungen.

Besondere Bedeutung haben dabei die TEE-Verbindungen, da sie Hinweise auf Verdeckungen geben: die waagerechte Linie des TEE's gehört zu einem Objekt, welches räumlich gesehen vor dem Objekt, das durch die vertikale Linie TEE's (Fuß) begrenzt wird, liegt.

Im Beispiel der Abbildung 9.3 treten u.a. folgende Knotentypen auf:

²⁰Insbesondere existiert keine Standardzugriffssprache wie SQL. Komplexe Abfragen sind jedoch mit den Konstrukten der Sprache POP11 realisiert.

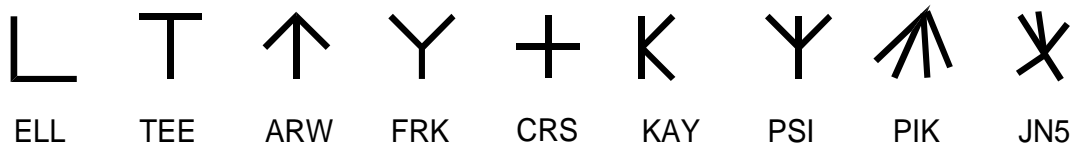


Abbildung 9.2: Verschiedene Knotentypen

```
[line1 poi1 poi2]
[line2 poi2 poi3]
[line5 poi2 poi5]
.....
[poi poi1 10 100]
[poi poi2 40 100]
.....
[junc poi1 ell [lin1 lin4]]
[junc poi2 tee [lin1 lin5 lin2]]
[junc poi3 tee [lin2 lin6 lin3]]
[junc poi5 tee [lin9 lin10 lin5]]
...
```

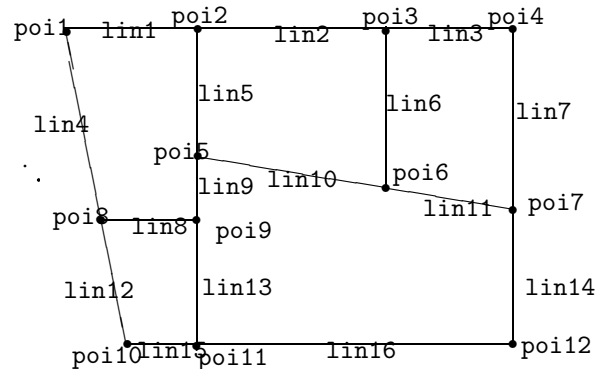


Abbildung 9.3: Punkte, Linien und Knoten

junc kennzeichnet den Datenbankeintrag als Spezifizierung des Knotentyps. Der Punkt **poi1** beispielsweise ist ein ELL-Knoten, in dem die Linien **lin1** und **lin4** zusammenkommen.

Linien: Eine Linie wird beschrieben durch den Bezeichner **line**, ihren eindeutigen Identifikator **lin1**, sowie die beteiligten Punkte **poi1**, **poi2**. Linie **lin1** verbindet also die Punkte **poi1** und **poi2**.

```
[line lin1 poi1 poi2]
[line lin2 poi2 poi3]
[line lin3 poi4 poi3]
[line lin4 poi8 poi1]
.....
```

Flächen: Die Repräsentation der Flächen erfolgt objektorientiert. Jede Fläche bildet ein Objekt der Klasse Polygon. Es enthält als wesentliche Beschreibungselemente die Liste der Punkte, die das Objekt begrenzen. Weitere Eigenschaften der Klasse Polygon werden im Kapitel 9.1.4 dargestellt.

Das System geht davon aus, daß die Eingangsinformation in Form von Linienstücken mit ihren Endpunkten gegeben ist. Für die Bestimmung aller weiteren Objekte und Eigenschaften liegen Routinen vor. Dies sind u.a. Funktionen zur Ermittlung geschlossener Flächen aus der Sammlung von Linienstücken, Bestimmung des Flächeninhalts, des Flächentyps, der Form, des minimal umschreibenden Rechtecks (bezogen auf die Hauptachsen des Objekts), der Länglichkeit, der Kompaktheit.

9.1.2 Definition der Basisrelationen

Zwischen den drei Grundobjekten Punkt, Linie und Fläche existieren prinzipiell 6 mögliche Relationen, die die verschiedensten Ausprägungen haben können [Molenaar 1989], [Bill & Fritsch 1991]. Einige dieser Relationen sind in den Daten implizit gegeben. So ist die Relation einer Fläche mit einer sie begrenzenden Linie in der Datenstruktur der Fläche gegeben, auch die Inzidenz des Endpunkts einer Linie mit der Linie wird implizit angenommen. Einige Beispiele für Relationen, die als Grundfunktionen implementiert sind, sind im folgenden aufgeführt:

Punktrelationen:

- Abstand zweier Punkte
- Abstand Punkt-Linie
- Punkt in Polygon
- ...

Linienrelationen:

- Abstand zweier Linien
- Winkel zwischen zwei Linien
- Parallelität zweier Linien
- Orthogonalität zweier Linien
- ...

Flächenrelationen:

- gemeinsame Punkte zweier Flächen Anzahl der Punkte zweier Flächen, die in beiden enthalten sind
- gemeinsame Knotentypen: TEE-, ELL-, ...Knoten: Anzahl der Punkte zweier Flächen, die in beiden enthalten sind und zusätzlich noch vom Typ TEE, ELL, etc. sind.
- Größenunterschied
- Formunterschied
- gemeinsame Linien sowie Länge derselben
- Abstand zweier Flächen
- Enthaltensein zweier Flächen
- ...

Alle oben genannten Relationen sind als Funktionen realisiert. Während die Flächen-Flächen-Relationen **explizit** in Form eines Funktions-Pools in der Datenbank realisiert sind, dienen alle anderen Relationen der Ableitung von Zwischenergebnissen.

9.1.3 Erweiterbarer Funktionen-Pool

Für flächenhafte Objekte stehen eine Reihe von Funktionen zur Verfügung, mit denen sich ihre Merkmale, sowie ihre Relationen zu anderen Objekten bestimmen lassen. Diese Funktionen werden in der Datenbank in einer offenen Liste gesammelt. Neu hinzukommende Funktionen können einfach in die Liste mit aufgenommen werden, womit eine einfache Erweiterbarkeit des Systems gewährleistet ist. Eine Funktion wird in Form einer "Regel" spezifiziert, wobei angegeben ist, für welches Konzept (d.i. ein Objekt oder eine Relation) sie gilt. Z.B. kann für ein Objekt "Fläche" die Größe oder auch die Form bestimmt werden; zwischen zwei Flächen kann, wie oben angegeben, beispielsweise die Relation "Größenunterschied" berechnet werden.

```
[rule objekt object_form]
[rule objekt object_size]
[rule relation size_diff]
```

Diese Art der Wissensverwaltung ist ein Schritt hin zur Trennung von Abarbeitungsstrategien und Daten und erhöht die Flexibilität des Systems: neue "Regeln" können auf diese Weise sehr einfach hinzugenommen werden, um die Funktionalität zu erweitern. Steht beispielsweise ein Farbbild zur Verfügung, wo kann neben den bisher rein geometrischen Merkmalen auch die Farbe als zusätzliches Attribut mit aufgenommen werden, indem eine Regel zur Ermittlung der Farbe mit in die Datenbank integriert wird.

Diese Regeln lesen sich folgendermaßen: die Funktionen **object_form** und **object_size** können nur für Objekte vom Typ **Objekt** angewandt werden; gleichermaßen läßt sich die Funktion **size_diff** nur für Relationen zwischen zwei Polygonen bestimmen. Eine konventionelle Art und Weise, solch eine Funktionalität zu gewährleisten, sieht folgendermaßen aus:

```

if concept = objekt then
    object_size;
    object_form;
    .....
elseif concept = relation then
    size_diff;
    ....
endif;

```

Werden die Funktionen auf diese Weise explizit kodiert, so erfordert eine Erweiterung des Wissens eine Modifikation des Programms selbst: neue Funktionen oder Konzepte lassen sich nur einfügen, indem der Programmcode selbst verändert wird, was zudem ein neuerliches Übersetzen mit sich bringt. Im Falle der regelbasierten Realisierung hingegen genügt einfach eine Erweiterung der Wissensbasis um die neue Regel²¹. Das Zusatzwissen, daß Flächen auch das Attribut Farbe beinhalten können, wird auf folgende Weise realisiert:

```
[rule objekt object_color]
```

Diese Regel steht dem System dann sofort zur Verfügung. Hier ist natürlich das Interpreterkonzept, wie es hinter der Programmiersprache POP11 steht, von größter Bedeutung. Zur Bestimmung der Eigenschaften und Relationen flächenhafter Objekte sind also immer lediglich alle Elemente der Datenbank mit dem Schlüssel "rule objekt" resp. "rule relation" zu selektieren und die entsprechenden Funktionen auszuwerten.

Folgende Liste von Funktionen sind für Konzepte (d.h. Objekte und Relationen) bekannt. In Listen wird der Name der Funktion definiert, sowie für welche Art von Konzept die Regel gilt.

```

[rule objekt contains]
[rule objekt contained]
[rule objekt object_size]
....
[rule relation connection]
[rule relation inside]
[rule relation encloses]
....

```

Die Funktionen berechnen im Einzelnen:

Objektfunktionen: Funktionen zur Bestimmung von Eigenschaften von Polygonen

contains: stellt fest, ob das Objekt andere Objekte enthält.

contained: stellt fest, ob das Objekt in einem anderen Objekt enthalten ist.

object_elongation: charakterisiert die Kompaktheit des Objekts
($Umfang^2 / (4 * \pi * \text{Flächeninhalt})$).

²¹Voraussetzung ist natürlich, daß auf die Datenbank dynamisch zugegriffen werden kann, d.h. daß eine Erweiterung bzw. Veränderung des Inhalts der Datenbank immer sofort aktuell verfügbar ist.

object_lq: bestimmt das Verhältnis von Länge zu Breite des minimal umschließenden Rechtecks.

object_size: ermittelt Größe des Objekts.

object_form: ermittelt Form des Objekts.

num_points, num_right_angles, num_parallel, num_same_length: diese Funktionen berechnen die Anzahl der Punkte, der Rechten Winkel, der parallelen Seiten, sowie der gleich langen Seiten.

junction_ell, junction_tee, junction_frk, junction_arw, junction_jn5: diese Funktionen bestimmen die Anzahl der entsprechenden Knotentypen eines Polygons.

Relationen-Funktionen: Funktionen zur Bestimmung von Relationen zwischen zwei Polygonen

connection: bestimmt, ob zwei Polygone verbunden sind, d.h. gemeinsame Kanten haben.

inside, encloses: ermittelt, ob erstes Objekt innerhalb (außerhalb) des zweiten liegt.

are_parallel, are_orthogonal: spezifiziert, ob die Hauptachsen der beiden Objekte parallel oder orthogonal zueinander stehen.

common_ell, common_tee, common_frk, common_arw, common_jn5, common_stem: ermittelt die Anzahl der gemeinsamen Knoten verschiedener Typen (siehe Abbildung 9.2); **common_stem** steht für einen TEE-Knoten, wobei der Fuß des TEE's beiden Flächen jeweils gemeinsam ist.

size_diff: ermittelt den Größenunterschied der beiden Polygone.

distance: bestimmt Abstand der beiden Polygone (bezogen auf Mittelpunkte der Flächen).

left_position, top_position: ermittelt, ob das erste Objekt links (oberhalb) des zweiten liegt.

common_sides: bestimmt die Länge der gemeinsamen Seiten.

same_polytype: stellt fest, ob die Objekte gleichen Polygontyps sind.

Die Objekteigenschaften und -relationen werden somit in Form einzelner Berechnungsfunktionen vorgehalten. Die Eigenschaften bilden unmittelbar die Merkmale polygonaler Objekte, und werden somit in der Frame-Struktur übernommen.

9.1.4 Objektklassen

Geometrische Objekte in ihrer allgemeinsten Form setzen sich aus Geometrie (incl. Topologie) und Semantik zusammen. In vorliegendem Zusammenhang werden die Objekte primär über ihre Geometrie definiert: zunächst können polygonale Objekte jede beliebige Semantik besitzen - diese wird später im Rahmen einer Bedeutungs-zuweisung geklärt. Das Polygon bildet daher die allgemeinste Objektklasse, welche sich dann in die einzelnen semantischen Objekte aufspaltet. Die Klassen sind jeweils über Frames beschrieben. Um eine bessere Strukturierung zu gewährleisten, wurde eine Oberklasse von **polygon** namens **gen_object** angelegt, in welcher einige Methoden und Eigenschaften polygonaler Objekte definiert sind. Das Polygon-Objekt selbst kennt lediglich strukturbildende Methoden, während alle Berechnungsfunktionen bzw. Hilfsroutinen als Methoden der Klasse **gen_object** abgelegt sind. Diese Trennung hat allerdings lediglich programmtechnische Gründe und bleibt für den Nutzer versteckt - er arbeitet mit den polygonalen Objekten. Alle weiteren Objektklassen, d.h. *semantische Objekte* wie Straßen, Häuser, . . . , die im Laufe der Interpretation entstehen, werden als Unterklasse von **polygon** angelegt (siehe Abbildung 9.4).

In Abbildung 9.5 sind die möglichen Beziehungen der Objekte untereinander angegeben. Die Aggregation (part_of-Beziehung) erlaubt ein Zusammenfassen von Objekten zu einem neuen Objekt. Dieses kann entweder vom Typ **polygon** sein, oder ein neues - komplexes - Objekt bilden. Assoziationen stehen für beliebige Relationen zwischen Objekten (Nachbarschaft, Ähnlichkeit, . . .). Sie charakterisieren damit den Kontext zu anderen Objekten.

Anhand der Objektklassen **gen_object** und **polygon** sollen die Merkmale und Vorteile objektorientierter Verarbeitung dargestellt werden.

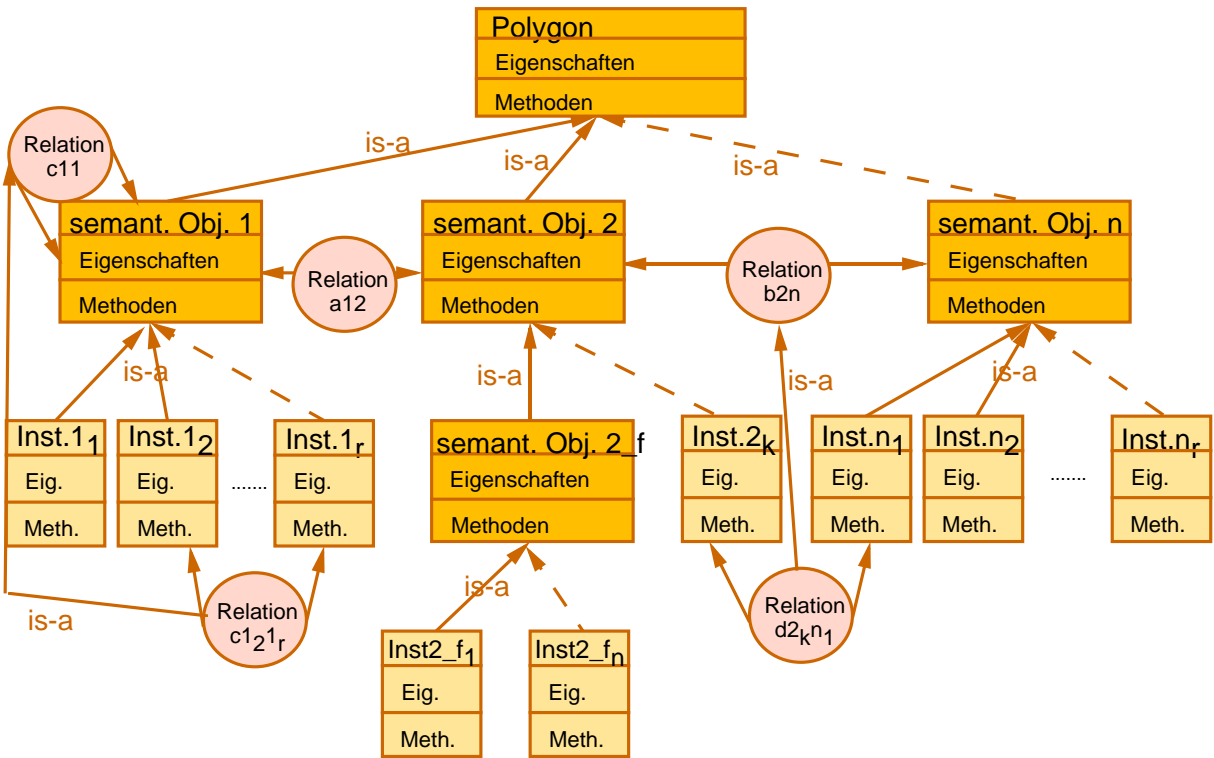


Abbildung 9.4: Übersicht über verwendete Objektklassen: semantische Objekte sind Unterklassen von Polygonen; alle Objektklassen können weitere Unterklassen, sowie individuelle Objektinstanzen besitzen.

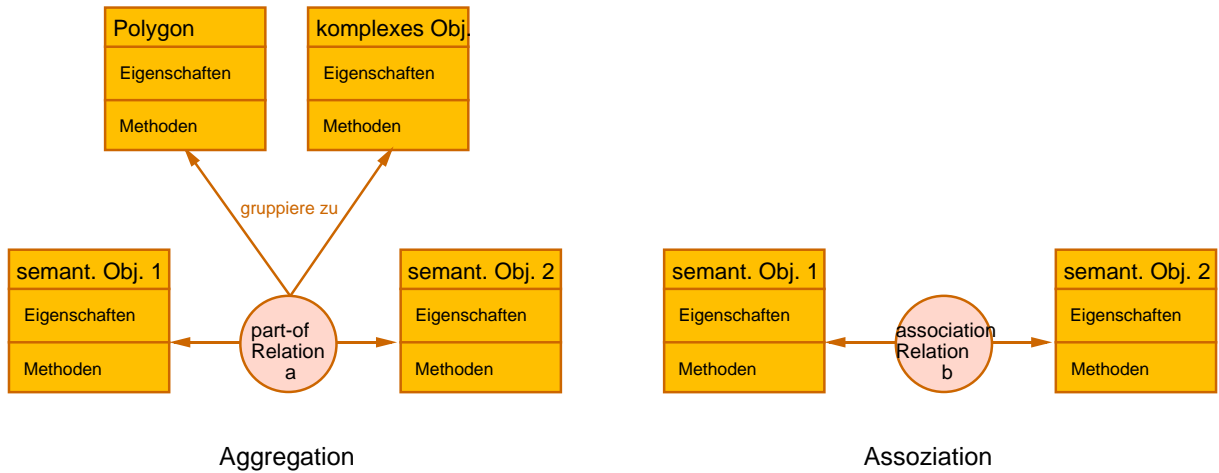


Abbildung 9.5: Übersicht über die Relationen der Objekte: Aggregation und Assoziation

Objektklasse gen_object:

Das generische Objekt verfügt über die charakterisierenden Eigenschaften und Methoden, welche in Abbildung 9.6 aufgelistet sind. In dieser Objektklasse sind primär die geometrischen (**all_points**, **all_lines**, ...) und strukturbildenden Merkmale polygonaler Objekte verzeichnet (**part_of_relation**, **part_of_objects**, **association**, **assoc_objects**); letztere sind besonders für die Aggregation von Bedeutung (siehe Kapitel 9.1.5).

Objektklasse polygon:

Die Klasse **polygon** ist eine Spezialisierung der Klasse **gen_object**. Abbildung 9.7 listet die wichtigsten Eigenschaften und Methoden dieser Klasse auf. Um eine dynamische Erweiterung der Objektmerkmale zu gewährleisten, werden diese erst zur Laufzeit aus dem Funktionen-Pool entnommen (vgl. Kapitel 9.1.3). All diese Funktionen werden dann entsprechend für das polygonale Objekt berechnet. Das Polygon-Objekt verfügt also - neben den von **gen_object** ererbten - noch über die Objekteigenschaften des Funktionenpools. Außerdem kennt

Eigenschaften / Slots	Methoden
name	am_i(unknown_object) stellt fest, ob das unbekannte Objekt <code>unknown_object</code> von gleichem Typ ist, wie das aktuelle Objekt selbst
all_points	part_of stellt fest, ob Objekt mit weiteren Objekten zusammengefaßt werden kann und führt diese Aggregation durch
all_lines	unite_with(object) vereinigt aktuelles Objekt mit Objekt <code>object</code>
...	search_neighbor sucht Relationen (vom Typ <code>part_of_relation</code>) und schreibt diese in den Slot <code>obj_relations</code>
part_of_relation	search_assoc sucht nach weiteren Relationen (<code>associations</code>) schreibt diese in den Slot <code>obj_assoc</code>
part_of_objects	kill_me löscht Objekt und dessen Relationen zu anderen Objekten
association	rc_show_me objekt graphisch darstellen
assoc_objects	after_initialize berechne verschiedene Objekteigenschaften
...	
status	
composed_of	
belongs_to	

Abbildung 9.6: Methoden der Objektklasse Generisches Objekt

Eigenschaften	Methoden
<i>Objektmerkmale</i> <i>aus Funktionenpool</i>	recognize_object ermittelt, ob das unbekannte Polygon einer bekannten Objektklasse zugeordnet werden kann und erzeugt im positiven Fall eine neue Instanz dieser Klasse
	after_initialize sammelt Objekte in einer Liste

Abbildung 9.7: Methoden der Objektklasse Polygon

es zusätzliche Methoden, welche der Strukturierung dienen, nämlich der Bildung neuer Objektinstanzen.

9.1.5 Wichtige Methoden flächenhafter Objekte

Die im folgenden angeführten Methoden sind von allgemeiner Art, welche allen (flächenhaften, d.h. zur Klasse `polygon` gehörenden) Objekten zur Verfügung stehen. Diese strukturbildenden Methoden dienen primär der Beschreibung und Erzeugung einer hierarchischen Darstellung. Diese Hierarchie läßt sich auf zweierlei Arten fassen: als Klassifikations- und als Aggregationshierarchie [Janssen 1993].

Klassifikationshierarchie: Objekte mit teilweise gemeinsamen Eigenschaften lassen sich zu einer Objektklasse zusammenfassen; diese wiederum kann die Unterklasse einer weiteren Klasse bilden. Der entstehende Hierarchiebaum kann auf diese Weise viele Ebenen enthalten. Der Übergang von einer Oberklasse zur nächst tieferen wird mit Spezialisierung, der umgekehrte mit Generalisierung bezeichnet. Ein Haus beispielsweise gehört zur Oberklasse Gebäude, diese wiederum kann zu den Landschaftsobjekten gezählt werden. Diese Form der Hierarchie ist in einer objektorientierten Darstellung unmittelbar gegeben. Üblicherweise spricht man von einer sog. **instance-of** oder **is-a**-Beziehung.

Aggregationshierarchie: Die zweite Grundfunktionalität verbindet verschiedene Objekte bzw. Objektteile zu neuen Objekten. Solch eine Beziehung besteht beispielsweise zwischen einem Haus und seinen Teilen, aus denen es sich zusammensetzt: Fenster, Tür, Wand, Dach, ... Diese Beziehung wird auch **part-of**-Beziehung genannt.

Mit dem Beispiel des Hauses wird bereits deutlich, daß diese beiden Formen auch gekoppelt werden können. Sowohl die elementaren Objekte (z.B. Fenster, Tür, Haustür, Garagentür, Zimmertür), als auch die zusammengesetzten (Haus, Gebäude, ...) können in eine Klassifikationshierarchie eingebettet sein (vgl. Abbildung 9.8).

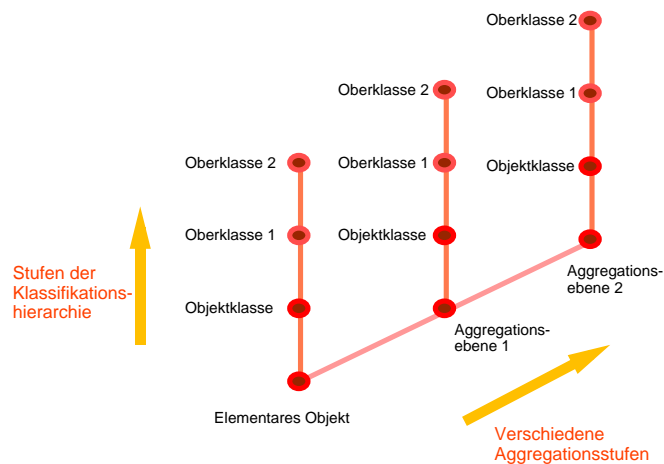


Abbildung 9.8: Verschiedene Hierarchietypen ([Janssen 1993] nach [Husing 1993])

Die Klassifikationshierarchie ist durch den objektorientierten Ansatz bereits gegeben. Neue Objektklassen können gebildet werden, wobei diese alle Eigenschaften und Methoden ihrer Oberklasse erben. Dies wird im Lernverfahren durchgeführt, indem Spezialisierungen der jeweils angegebenen Objekte ermittelt werden. Ebenfalls können Instanzen beliebiger Klassen erzeugt werden. Die Erzeugung neuer Objekte erfolgt nach Anwendung der Methode `recognize_object`. Im Gegensatz hierzu wird in der Aggregationshierarchie das mögliche Zusammenfassen von verschiedenen Objekten zu neuen Objekten vorgegeben. Sie wurde als weitere Methode (`part_of`) implementiert. Abbildung 9.9 zeigt die verwendeten Objektklassen und Methoden im Überblick.

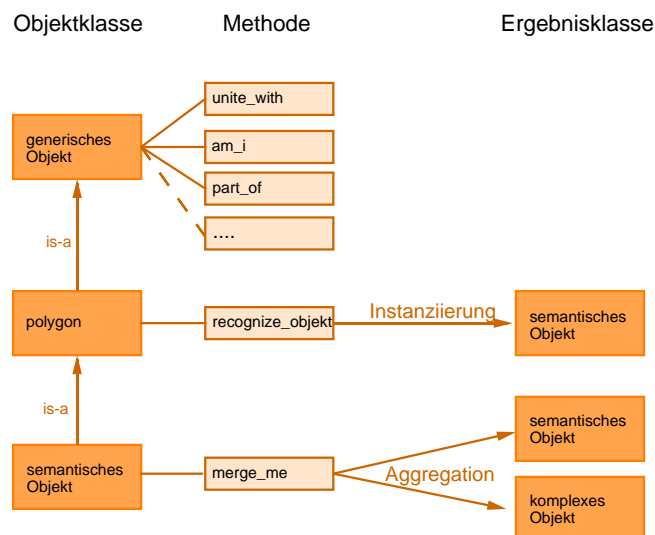


Abbildung 9.9: Übersicht über die Objektklassen und Methoden

Methode `recognize_object`:

Die Methode `recognize_object` dient dazu, Objekte vom Typ `polygon` weiter zu spezifizieren in die Unterklassen der Klasse `polygon`, d.h. die semantischen Objekte `feld`, `strasse`, `haus`, ..., welche im Laufe der

Analyse entstehen können. Diese Methode überprüft, ob sich das unbekannte Objekt mit der diskriminierenden Objektfunktion²² bestätigen läßt. Das Erkennen basiert üblicherweise allein auf der diskriminierenden Funktion; weiter können jedoch auch noch die Objekteigenschaften mit denen des Prototyps verglichen werden. Darüberhinaus läßt sich prüfen, ob auch die Relationen des zu neuen Objekts mit den Soll-Relationen übereinstimmen. Ist dies der Fall, so erfolgt eine Zuweisung dieses Typs an das polygonale Objekt und schließlich die Erzeugung einer neuen Objektinstanz.

In POP11 lautet diese Methode²³:

```
defmethod recognize_object -> success;
  vars success;

  ;;; Bestimmung der Objektklasse des unbekanntes Objekts

  self <- myflavour -> polyflav;
  pr('ich bin ein objekt der klasse : ');polyflav==>

  ;;; Bestimmung der Unterklassen dieser Objektklasse

  polyflav <- subflavours -> all_subclasses;
  pr('ich teste alle subclasses: \n');all_subclasses=>

  ;;; Teste fuer alle Unterklassen, ob das unbekannte Objekt diesem
  ;;; Typ entspricht

  for class in all_subclasses do
    consword( (class<-name)><"_proto" ) -> object;
    pr('vergleich mit : ');object==>

    ;;; Die eigentliche Ueberpruefung uebernimmt die Methode am_i

    object <- am_i(self);

    ;;; entspricht der so ermittelte Objekttyp dem Prototyp der Klasse,
    ;;; so wird neue Objektinstanz erzeugt.

    if type == object<-name then
      create_instance(self) -> success;
      pr('erfolgreich \n');
    else
      pr('nicht erfolgreich \n');
    endif;
  endfor;
enddefmethod;
```

Für ein polygonales Objekt wird also überprüft, welchen semantischen Objekten es zugeordnet werden könnte. Letztere sind als Unterklassen von `polygon` abgelegt. Daher wird es mit allen bisher bekannten Unterklassen von `polygon` verglichen und schließlich der entsprechenden Klasse zugewiesen.

Methoden `part_of`:

Diese Methode steht für den Aufbau eines neuen Konzepts aus anderen Konzepten. Beispielsweise besteht eine solche `part-of`-Beziehung zwischen den Bestandteilen eines Hauses ("Tür, Fenster, Wand, ...") und dem "Haus" selbst. Die Möglichkeit der Aggregation wird im Slot `part_of_relation` spezifiziert. Dort ist festgelegt, welche

²²Vergleiche Kapitel 9.3.1 zur Ermittlung der diskriminierenden Objektfunktion.

²³In POP11-Syntax bedeutet der Pfeil `->` eine Wertzuweisung; der umgekehrte Pfeil `<-` steht für die Anwendung einer Methode.

Objekte aufgrund welcher Bedingungen zusammengefaßt werden können. Auch der Typ des neuen Objekts ist hier angegeben. Diese Bedingung für eine Aggregation ist immer objektspezifisch, kann also nicht allgemein a priori festgelegt werden. Daher wird diese Relationsfunktion automatisch durch Lernen aus vorgegebenen Beispielen abgeleitet (siehe Kap. 9.3.2). Im Slot `part_of_objects` werden die Objekte gesammelt, mit denen es in der vorgegebenen Relation steht. Mit diesen kann es aggregiert werden. Danach wird dies in allen beteiligten Objekten nachgeführt: im Slot `composed_of` werden die einzelnen Objekte beschrieben, aus denen das neue sich zusammensetzt; die gruppierten Objekte belegen umgekehrt im Slot `belongs_to`, zu welchen Objekten sie gehören.

Die Erzeugung neuer Objektklassen und Objektinstanzen ist somit an verschiedenen Stellen im Programm realisiert. Die Objektklassen werden zunächst im Lernverfahren durch Lernen aus Beispielen festgelegt. Mit der Methode `recognize_object` werden Objektinstanzen abgeleitet. Eine weitere Möglichkeit der Generierung neuer Objektklassen besteht über die Methode `merge_me`. Die Aggregation zweier Objekte resultiert in einem neuen Objekt. Dieses kann von einem bekannten Typ sein, kann allerdings auch ein neues Objekt darstellen, welches dann der Klasse `complex` (ebenfalls eine Unterklasse von `Polygon`) zugeordnet wird.

9.2 Ablauf des Interpretationszyklus

Die Aufgabe des System ist es, die gegebene, visuelle Information zu interpretieren. Zunächst werden die bekannten Basisobjekte identifiziert und die Polygone bestimmt, danach werden diese klassifiziert. Hierzu dient die Erkennungsfunktion, die zu den Methoden der Objektklasse `polygon` gehören. Die Grundidee des vorliegenden Systems ist, daß die Ablaufsteuerung dabei direkt von den Objekten selbst ausgeht: jedes Objekt prüft, welche Methoden ihm zur Verfügung stehen, und ob es sie auf die gegebenen Daten anwenden kann. In POP11-Code lautet diese Grundstrategie folgendermaßen:

```
;;; nimmt nacheinander alle Objekte aus der Liste aller Objekte ALLobjects
;;; und fuehrt deren moegliche Aktionen aus

while ALLobjects matches [?testobjekt ??ALLobjects] do
  ;;; solange noch Objekte in der Liste aller Objekte (ALLobjects)
  ;;; vorhanden sind,
  ;;; selektiere ein Objekt TESTOBJEKT und fuehre alle dessen
  ;;; moegliche Aktionen durch

  testobjekt <- methods -> all_methods;
  pr('START: alle meine methoden: ');all_methods==>

  ;;; Alle Methoden ALL_METHODS werden nun der Reihe nach auf das Objekt
  ;;; TESTOBJEKT angewandt;
  ;;; Das Ergebnis steht im Wert SUCCESS.
  ;;; Das Ergebnisobjekt wird anschliessend in die Liste ALLobjects mit
  ;;; aufgenommen.

  for do_me in all_methods do
    testobjekt(do_me) -> success;
    if success then
      [^^ALLobjects ^success] -> ALLobjects;
    else
      pr('hiermit kann ich noch nichts anfangen, es bleibt in der liste');
      [^^ALLobjects ^testobjekt] -> ALLobjects;
    endif;
  endfor;
endwhile;
```

Die einzelnen Objekte werden der Reihe nach abgearbeitet. Für jedes TESTOBJECT wird geprüft, welcher Objektklasse es zugehört, und über welche Methoden es somit verfügt (ALL_METHODS). Diese Methoden

werden anschließend nacheinander durchgeführt. Das Ergebnis der Aktion ist ein neues Objekt, welches in die Liste aller Objekte aufgenommen wird, und somit in einer weiteren Schleife verarbeitet werden kann. Es kann selbst wieder über eigene Methoden verfügen, die es auf die übrigen Daten anwendet. Falls die Methode nicht erfolgreich war, so wird das ursprüngliche Objekt wieder in die Liste mit aufgenommen, um gegebenenfalls in einem späteren Zyklus weiterverarbeitet werden zu können.

9.3 Lernverfahren

Lernen kann an verschiedenen Stellen im Prozeß angestoßen werden, wobei immer aus vorhandener Information durch Vorgabe einer Klassifizierung neues Wissen ermittelt wird. Dazu werden Beispiele und Gegenbeispiele für das zu lernende Konzept angezeigt²⁴. Das Konzept wird allein aus diesen Beispielen abgeleitet, daher hängt der Lernerfolg von der sorgfältigen Auswahl der Beispiele ab. Die Lernmöglichkeiten des Systems werden zunächst im einzelnen beschrieben, bevor sie in einem gesonderten Kapitel in Form eines beispielhaften Programmablaufs im Zusammenhang dargestellt werden.

9.3.1 Erzeugung von Objektklassen, ihren diskriminierenden Objektfunktionen und Prototypen

Ein Objekt kann erkannt werden, indem es anhand seiner Merkmale einer Objektklasse zugeordnet wird. Diese Zuordnung erfolgt aufgrund derjenigen Merkmale, mit denen es sich am stärksten von allen anderen bekannten Objekten unterscheidet. Welche Merkmale dies sind, wird mittels ID3 bestimmt und in einer Unterscheidungsfunktion abgespeichert. Dazu werden Objekte der zu diskriminierenden Klasse als positive Beispiele genommen, die allen bislang bekannten Objekten - als negative Beispiele gewertet - gegenüberstehen. ID3 selektiert die wichtigsten Merkmale und generiert damit eine Objektfunktion. Jede Objektklasse verfügt dann über eine solche Funktion, mit der seine Erkennung erfolgen kann. Neue Objektklassen werden immer als Spezialisierung bereits bestehender Klassen erzeugt. Dies wird zunächst am Beispiel der Klasse `polygon` verdeutlicht, welche sich in die semantischen Objekte spezialisiert, später dann aber auch anhand weiterer Objektklassen gezeigt.

Im folgendem sei beispielsweise davon ausgegangen, daß Polygonobjekte gezeigt und als Feld bzw. Haus bezeichnet wurden. Automatisch werden die Objektklassen `haus` und `feld` als Unterklasse von `polygon` erzeugt. Die Klasse `haus` wird in Flavours beispielsweise folgendermaßen beschrieben:

```
flavour haus isa polygon ;
  defmethod printself;
    pr('OBJECT: haus\n');
    printf('attr: name : %p\n', [^name]);
    printf('attr: composed_of      : %p\n', [^composed_of]);
    printf('attr: belongs_to      : %p\n', [^belongs_to]);
    printf('attr: part_of_relation : %p\n', [^part_of_relation]);
    printf('attr: obj_relations   : %p\n', [^obj_relations]);
    printf('attr: association    : %p\n', [^association]);
    printf('attr: obj_assoc      : %p\n', [^obj_assoc]);
  enddefmethod;
  defmethod after initialise;
    [^name ^^DBhaus] -> DBhaus;
  enddefmethod;
endflavour;
```

Die Klasse wird als Unterklasse von `polygon` definiert. Alle Instanzen dieser Objektklasse werden in einer Liste `DB<<objektklasse>>` gespeichert, die jeweils bei der Erzeugung eines neuen Objektes aufdatiert wird (`after initialise`). Hiermit ist es möglich, alle existierenden Objekte zu sammeln. Die Objektklasse verfügt über eine spezielle Methode der Darstellung (`printself`).

²⁴Lernen mit ID3 erfolgt immer auf der Basis von Beispielen und Gegenbeispielen. Hintergrund ist, daß - solange keine anderen Gegenbeispiele bekannt sind - alle Objekte in einer Klasse zusammengefaßt werden können; sobald ein Gegenbeispiel bekannt ist, ist eine Unterscheidung möglich.

Zur Bestimmung der diskriminierenden Funktion werden die Eigenschaften der Beispielobjekte, nämlich die "Regeln" für Objekte des Funktionenpools (contains, contained, object_elongation, ..., junction_njn5), für die gezeigten Polygone berechnet und in einer Datei aufgelistet:

```
% polygon_object contains contained object_elongation object_lq object_size
polytyp num_points num_right_angles num_parallel num_same_length object_form
junction_ell junction_tee junction_frk junction_arw junction_njn5 %
[haus no yes 1.27 1.00 1190.00 quadrat 4 4 2 3 SQ 4 0 0 0 0 ] ;;; poly7
[haus no yes 1.30 0.76 480.02 rechteck 4 4 2 2 H 2 2 0 0 0 ] ;;; poly6
[feld no no 1.88 0.28 10231.34 rechteck 4 4 2 2 V 1 3 0 0 0 ] ;;; poly3
[feld yes no 1.74 0.32 11667.89 rechteck 4 4 2 2 V 0 5 0 0 0 ] ;;; poly2
```

Es werden nun für alle neuen Objektklassen die diskriminierenden Funktionen bestimmt, in diesem Falle die Funktionen **feld** und **haus**. Am Beispiel des Lernens der Hausfunktion sei dies veranschaulicht: alle Objekte vom Typ **haus** werden als positive Beispiele angenommen, alle anderen (d.h. die Feldobjekte) gelten als negative Beispiele. ID3 muß somit zwei Klassen trennen: die Klasse **haus** und die Klasse aller Nicht-Häuser (**not_haus**). Die entstehende Funktion lautet dann:

```
vars haus ;
define haus ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if ( contained (area) ->> val) == " no " then
    'not_haus' -> klasse;
  elseif ( contained (area) ->> val) == " yes " then
    'haus' -> klasse;
  endif;
enddefine;
```

Ein Haus kann also dadurch unterschieden werden, daß es in einer anderen Fläche enthalten ist. Neben dieser individuellen Klassifizierungsfunktion wird zusätzlich eine Funktion abgeleitet, die **alle** bisher bekannten Objekte unterscheiden kann. Diese kann insbesondere im Konfliktfall eingesetzt werden. Dazu wird die ursprüngliche Liste an ID3 übergeben, welches eine Klassifizierungsfunktion für die Unterscheidung von Feldern und Häusern ermittelt²⁵. Grundsätzlich werden daher immer so viele Funktionen erzeugt, wie Objektklassen vorhanden sind, sowie eine zusätzliche Funktion zur Unterscheidung aller Klassen.

Als weiteres Kennzeichen jeder Objektklasse wird ein sog. Objektprototyp ermittelt. Für dessen Bestimmung werden alle Eigenschaften aller Objektinstanzen einer Klasse jeweils gemittelt und einem neu zu instanzierenden Objekt (**feld_proto** und **haus_proto**) zugewiesen. Der Hausprototyp enthält folgende Werte:

```
make_instance([haus
  name haus_proto
  association []
  obj_assoc []
  relation []
  obj_relations []
  contains_val [no]
  contained_val [yes]
  object_elongation_val [1.28 +-0.017321]
  object_lq_val [0.92 +-0.138564]
  polytyp_val [rechteck quadrat]
  ....
]) -> haus_proto ;
```

²⁵Diese Funktion muß nicht mit den diskriminierenden Funktionen übereinstimmen. Bei mehr als zwei zu unterscheidenden Klassen kann es vorkommen, daß sich alle Objekte untereinander durch andere Merkmale abgrenzen lassen, als im zwei-Klassen Fall (Objekt - not_Objekt).

Das Lernen von Objektklassen wurde somit für die Spezialisierung der Klasse `polygon` dargestellt. Sie läßt sich jedoch auch auf weiteren Stufen ansetzen. Die Objektklasse `haus` könnte beispielsweise in die Klassen `einfamilienhaus`, `garage`, `bungalow`, ... unterteilt werden. Dazu sind dann entsprechend die Hausobjekte anzugeben und zu klassifizieren. Der einzige Unterschied zur eben beschriebenen Vorgehensweise ist, daß die diskriminierende Funktion jetzt direkt von ihrer Oberklasse abhängt, im obigen Beispiel also von der Klasse `haus`²⁶. Eine Funktion zur Spezialisierung von Garagen könnte daher lauten:

```
vars garage ;
define garage ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if haus(area) then          ;;; Ueberpruefung der Oberklasse
    ;;; mit ID3 ermittelte diskriminierende Funktion fuer Garage
    ...
  endif;
enddefine;
```

Durch diese sukzessive Verfeinerung der Objektklassen ist eine sehr gute Strukturierung der gegebenen Daten möglich.

9.3.2 Lernen von Relationen und Erzeugung neuer Methoden

Für neu erzeugte Objekte können neue Methoden bestimmt werden. Diese dienen der Strukturierung der Objekte, d.h. insbesondere der Aggregation von Objekten (Objektteilen) zu neuen Objekten (Objektteilen). Die Methoden werden nachträglich an Objektklassen hinzugefügt; diese Erweiterung von Objektklassen wird durch Flavours unterstützt, indem die neue Methode einfach hinzugeladen werden kann. Diese inkrementelle Erzeugung von Objektbeschreibungen ist in sofern sinnvoll, als das System erst mit Kenntnis bestimmter Objekte bestimmte Aktionen durchführen kann. Die Aggregation kann nicht a priori definiert werden, da sie stark objektabhängig ist. So ist beispielsweise ein Zusammenfassen auf der Basis der Nachbarschaft möglich - wobei allerdings spezifiziert werden muß, was Nachbarschaft bedeutet (räumliche Nachbarschaft, Ähnlichkeit, Enthaltensein, ...). Aufgrund dieser Objektabhängigkeit werden die Bedingungen für eine Aggregation zweier Objekte ebenfalls durch Lernen ermittelt. Für die Bestimmung dieser sogenannten **part_of_relation** zeigt der Lehrer Beispiele und Gegenbeispiele. Die neu erzeugte Funktion wird anschließend als Methode **merge_me** an das jeweils beteiligte Objekt gegeben und erlaubt diesem in der weiteren Interpretation, eine Gruppierung zu einem neuen Objekt durchzuführen.

Dieser Formalismus wird beispielhaft an der Relation zwischen einem Haus und einer Garage gezeigt. Diese können unter bestimmten Bedingungen zu einem neuen Objekt (Bauwerk) zusammengefaßt werden. Diese Bedingung ist im Lernverfahren anhand vorgegebener Beispiele zu ermitteln. Der Lehrer zeigt daher Beispiele und Gegenbeispiele für diese Relation. Automatisch werden dann alle gemeinsamen Merkmale der zwei gezeigten Flächen berechnet. Dazu werden alle „Regeln“, die für Relationen gelten (z.B. `connection`, `inside`, ...), dem Funktionenpool der Datenbank entnommen und auf die beiden beteiligten Objekte angewandt. Es entsteht eine Sammlung von Merkmalen, welche mit ID3 dann bearbeitet wird.

```
% haus_garage connection inside encloses are_parallel are_orthogonal common_ell
common_tee common_stem common_frk common_arw common_njn5 size_diff
distance_2 left_position top_position common_sides same_polytyp %
[positiv yes no no no yes 0 2 1 0 0 0 762.98 24.84 left_of top_of 0.7 diff_poly ]
[negativ no no no yes no 0 0 0 0 0 0 709.98 87.57 left_of under 0.0 same_poly ]
[negativ no no yes no yes 0 0 0 0 0 0 8784.26 10.65 left_of top_of 0.0 same_poly ]
```

Das Lernprogramm ermittelt hieraus die gesuchte Relationen-Funktion.

²⁶Diese Abhängigkeit wird im Falle der Polygonklasse implizit angenommen.

```

vars haus_garage ;
define haus_garage ( area1,area2 ) -> klasse ;
  vars klasse , area1,area2 ;
  undef -> klasse;
  if ( connection (area1,area2) ->> val) == " yes " then
    'positiv' -> klasse;
  elseif ( connection (area1,area2) ->> val) == " no " then
    'negativ' -> klasse;
  endif;
enddefine;

```

Die `part_of`-Beziehung impliziert eine Aggregation, daher kann die ursprüngliche Objektdefinition nun automatisch um die neu gelernte Relationsfunktion und die Aggregationsmethode (`merge_me`) erweitert werden:

```

flavour haus isa polygon ;
  ivars part_of_relation = [haus_garage positiv garage bauwerk];

  defmethod merge_me -> success;
    vars success;
    self <- part_of -> success;
  enddefmethod;
endflavour;

```

Im Slot `part_of_relation` ist dabei folgendes verzeichnet:

```

ivars part-of-relation [
  haus_garage   : Name der Relation
  positiv       : Positiver Ausgang der Funktion
  garage        : Partnerobjekt
  bauwerk       : Typ des aggregierten Objekts]

```

Diese Relation spezifiziert nun eine Eigenschaft von Häusern, nämlich mit Garagen benachbart und aggregierbar zu sein. Die abzurufende Relation ist die Funktion **haus_garage**. Diese Relation kann auf Objekte vom Typ **haus** angewandt werden. Das aggregierte neue komplexe Objekt ist vom Typ **bauwerk**. Die Objektklasse **haus** verfügt über die Methode **merge_me**, mit der sie diese Aggregation durchführen kann.

Assoziationen lernen:

Neben den zur Aggregation nötigen Methoden (`part_of`, bzw. `merge_me`) können auch weitere Beziehungen zwischen Objekten gelernt werden. Diese allgemeinen Beziehungen sind hier mit Assoziationen bezeichnet. Diese beinhalten keine Methoden, sie dienen lediglich der weiteren Charakterisierung eines Objekts, indem sie den Kontext zu anderen spezifizieren. Das Lernen dieser Assoziationsfunktionen erfolgt in gleicher Weise wie das Lernen der `part_of`-Beziehung. Die Objekte erhalten nun die Einträge in den Slot `association`, in welchem die Assoziation - analog wie die `part_of`-Beziehung - näher spezifiziert wird. Alle Objekte, mit denen es die Assoziation teilt, werden im Slot `assoc_objects` aufgelistet.

9.4 Konfliktlösungen

Die beschriebenen Lernverfahren stellen elementaren Komponenten des Systems dar. In der Anwendung kann es allerdings zu einigen Problemen kommen. So ist es möglich, daß durch eine Objektdiskriminierungsfunktion nicht alle Objekte richtig klassifiziert werden. Es können Fehlklassifizierung vorkommen, oder einzelne Objekte werden überhaupt nicht erkannt. Diese Fehler können auch beim Lernen von Relationen auftreten. Für diesen Fall ist die Möglichkeit einer Korrektur vorgesehen. Zum einen können die Interpretationen des Systems korrigiert werden, zum anderen können weitere, aussagekräftige Konzepte als Beispiele in den Lernschritt aufgenommen werden. Auf diese Weise kann der Lehrer noch immer in das System eingreifen, um ein erneutes Lernen anzustoßen.

9.5 Beispielhafter Programmablauf

Anhand des Kartenausschnitts von Abbildung 9.10 (rechts) soll der Programmablauf verdeutlicht werden. Beim Programmstart kennt das System lediglich die Basisobjekt Punkt und Linie, sowie Funktionen zur Ableitung von Polygonen. Zusätzlich verfügt es über Verfahren, mit der es sein Wissen (Objektwissen) im Laufe des Interpretations- und Lernzyklus sukzessive erweitern kann.

Zunächst werden aus den Bildbeispielen, die in Form digitalisierter Einzellinien gegeben sind, Objekte vom Typ Punkt und Linie erkannt, und flächenhafte Objekte erzeugt. Diese bilden Instanzen der Objektklasse `polygon` (`poly1`, \dots , `polyN`, siehe Abbildung 9.10, rechts). Diese sehr allgemeine Objektbestimmung wird im Laufe der Interpretation verfeinert. Dazu wird in einem Erkennungsschritt versucht, alle Methoden der jeweiligen Objektklasse auf die zu interpretierenden Objekte anzuwenden. Dies ist insbesondere die Methode `recognize_object`. Diese greift auf alle Unterklassen der Objektklasse zu, und versucht, das jeweilige Objekt als zu einer dieser Klassen gehörig zu erkennen.

Im Regelfall liegen allerdings zu Programmstart noch keine Unterklassen vor, so werden diese im Lernschritt zunächst bestimmt (Option "specialization"). Das Lernprogramm fordert den Lehrer dazu auf, Polygon-Objekte, die im Bild erkennbar sind, zu benennen. Als Auswahl wird die Liste aller Unterklassen der Objektklasse `polygon` angeboten. Diese ist zu Beginn leer. Daher fällt die Auswahl beim ersten Objekt auf `new_class`. Nach der Eingabe eines Namens wird eine Objektklasse dieses Namens als Unterklasse von `polygon` erzeugt. Das gezeigte Objekt bildet die erste Instanz der neuen Objektklasse. Nun können weitere Objekte benannt werden, die entweder zur eben erzeugten Klasse gehören, oder auch nochmals eine neue Klasse bilden. Abbildung 9.10 veranschaulicht dies. Dem System wurde bereits ein Feld-Objekt (`poly16`) gezeigt, daher erscheint diese Objektklasse bereits neben `new_class` in der Auswahlliste. Nachdem der Lehrer das Polygon `poly1` als Verkehr spezifiziert hat, wird eine neue Klasse `verkehr` erzeugt, welche von nun an ebenfalls im Menü erscheint.

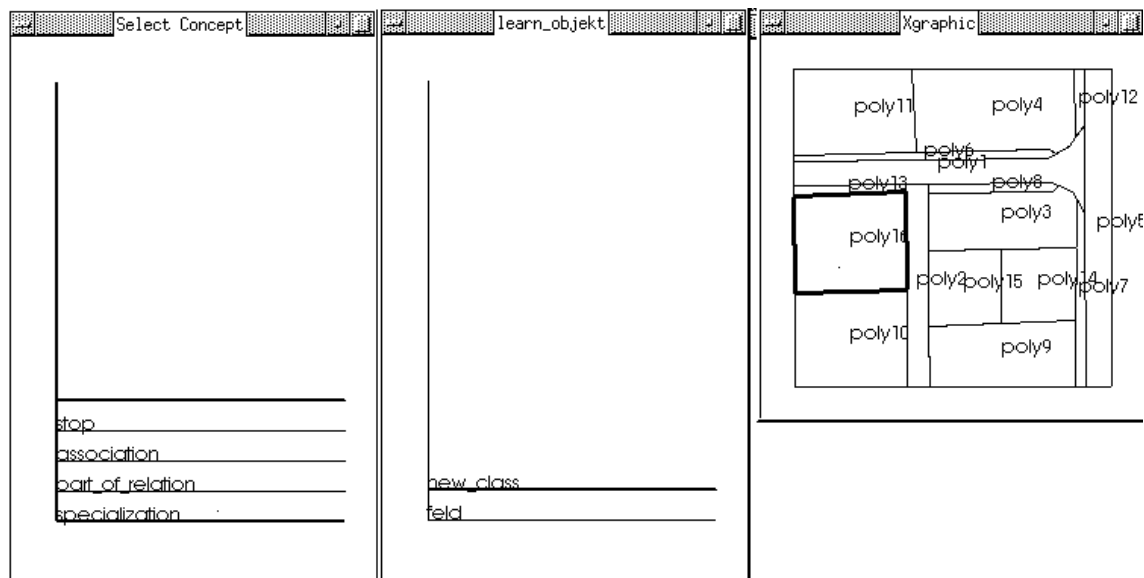


Abbildung 9.10: Lernen von den Objekt-Konzepten Feld und Verkehr

Sind genügend Beispiele gezeigt, so werden diese ausgewertet. Wie in Kapitel 9.3.1 angegeben, wird für jede Objektklasse "object" eine diskriminierende Funktion gleichen Namens, d.h. "object", bestimmt, mit deren Hilfe sie von allen anderen bisher bekannten Objekten unterschieden werden kann. Ebenfalls werden die strukturellen Eigenschaften der Objektinstanzen ermittelt und im Prototyp "object"-proto festgehalten.

Anhand des Beispiels soll die eben beschriebene Vorgehensweise nochmals veranschaulicht werden: der Lehrer zeigt auf die Objekte `poly16` und `poly3` und benennt sie als "Feld" und auf `poly1`, `poly12` und `poly6` und gibt ihnen die Klassifikation "Verkehr". Automatisch werden die Objektklassen `feld` und `verkehr` als Unterklasse von `polygon` erzeugt. Zur Ermittlung der diskriminierenden Funktionen `feld` und `verkehr` werden - wie in Kapitel 9.3.1 beschrieben - alle Merkmale für die gezeigten Objekte berechnet und an ID3 übergeben. In diesem Fall zeigt es sich, daß sich die beiden Objektklassen durch das Kompaktheitsmaß unterscheiden lassen.

Der Prototyp der Klasse `feld` enthält die Eigenschaften der zwei Feld-Beispiele und wird `feld_proto` genannt. Ebenso wird eine Instanz `verkehr_proto` erzeugt. Diese neuen Objektklassen und ihre zugehörigen Methoden können unmittelbar nach ihrer Erzeugung für die weitere Interpretation eingesetzt werden. Das heißt, bislang unbekannte Objekte können nun auf ihre Zugehörigkeit zu den beiden Objektklassen `verkehr` und `feld` geprüft werden. Auf diese Weise können im Beispiel alle Polygonobjekte korrekt klassifiziert werden.

Die Objektklasse `Verkehr` läßt sich nun noch weiter spezialisieren in die Klassen `radweg` und `strasse`. Dazu wird das Lernverfahren erneut mit der Option "specialization" aufgerufen. In analoger Weise wie oben werden vom Lehrer nun Objekte angezeigt und klassifiziert. Das System ermittelt ihre Klasse und legt die neuen Objekte als deren Unterklasse an. Aus den gegebenen Beispielen werden wiederum die jeweiligen Objektfunktionen (hier `radweg` und `strasse`) ermittelt. Diese Objektklassen hängen von ihrer Oberklasse `verkehr` ab, was sich auch in der diskriminierenden Funktion widerspiegelt: z.B. wird für die Bestimmung der Klasse `radweg` jetzt automatisch die Überprüfung der Oberklasse (`verkehr`) vorgeschaltet, d.h. nur wenn ein zu interpretierendes Objekt zur Klasse `Verkehr` gehört, kann es sich möglicherweise um einen Radweg handeln.

```
vars radweg;
define radweg ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if verkehr(area) then          ;; Ueberpruefung der Oberklasse
    if <= ( ( object_lq (area) ->> val) , 0.14 ) then
      'radweg' -> klasse;
    elseif > ( ( object_lq (area) ->> val) , 0.14 ) then
      'not_radweg' -> klasse;
    endif;
  endif;
enddefine;
```

Abbildung 9.11 zeigt das Ergebnis der Interpretation mit diesen neuen Spezialisierungen der Verkehrsobjekte. In diesem Falle wurden 3 Radwege irrtümlich als Straße klassifiziert. Hier kann nun die Möglichkeit der Korrektur genutzt werden, um diese Objekte richtig zu klassifizieren. Die fehlerhaften Objekte werden mit ihrer richtigen Klassifikation in die Liste der Beispielobjekte integriert, worauf die Objektfunktionen erneut abgeleitet wird. Nach der Korrektur zeigt sich in der berechneten Funktion, daß offensichtlich die Flächengröße ausschlagge-

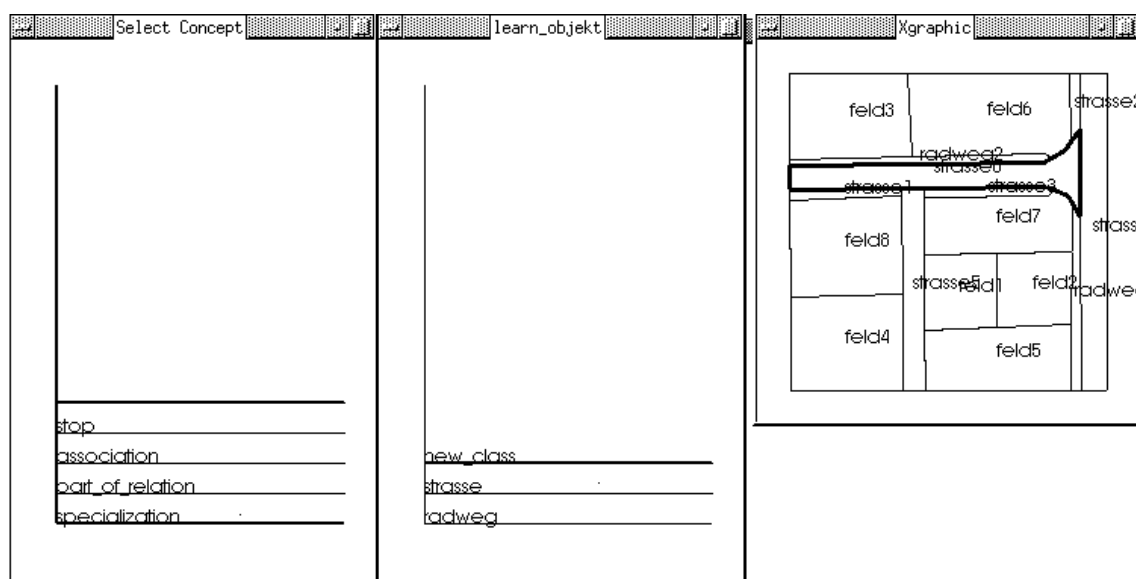


Abbildung 9.11: Fehlerhafte Interpretation dreier Radwege als Straße

bend ist für eine Unterscheidung zwischen Straße und Radweg, und nicht - wie aufgrund der ersten Beispiele angenommen - die Länglichkeit der Flächen.

```

vars radweg;
define radweg ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if verkehr(area) then          ;;; Ueberpruefung der Oberklasse
    if <= ( ( object_size (area) ->> val) , 1380.66 ) then
      'radweg' -> klasse;
    elseif > ( ( object_size (area) ->> val) , 1380.66 ) then
      'not_radweg' -> klasse;
    endif;
  endif;
enddefine;

```

Kommen im Laufe der Interpretation neue Instanzen hinzu, so kann die jeweilige Objektbeschreibung weiter spezifiziert werden, indem sie um die Merkmale der neue Objekte erweitert wird. Hier wird die Bedeutung des inkrementellen Lernens besonders klar. Aus einer "rohen" Initialbeschreibung kann nach und nach ein reiches Objektmodell abgeleitet werden. Wurde die Ausgangsbeschreibung beispielsweise lediglich aus einem oder zwei Objekten ermittelt, ist es sehr wahrscheinlich, daß ihre Merkmale nicht für alle Objekte gültig sind. Besonders die strukturellen Eigenschaften, die ja durch Mittelung einiger weniger Beispiele entstanden sind, sind sicher nur eingeschränkt gültig. Nach jedem Interpretationszyklus kann daher die Merkmalsbestimmung auf der Basis der dann bekannten Objekte wiederholt und somit verbessert werden.

Neben den Objektklassen können auch Relationen gelernt werden. Über das Menü wird als zu lernendes Konzept z.B. **association** angegeben. Als solche kann die Beziehung zwischen Straßen und Radwegen beschrieben werden. Für die Bestimmung der charakteristischen Eigenschaften dieser Funktion werden Beispiele hierfür gezeigt: strasse3 und radweg1 als positives Beispiel, die Paare (strasse1,radweg3), (strasse2, radweg4) sowie (strasse3, radweg2) als negative. Nach der Eingabe der Beispiele, sowie den Angaben Relationenname und Nachbarkonzept wird die Relationsfunktion mit ID3 ermittelt. Diese wird nun jedem Objekt dieser Klasse **strasse** als zusätzliche Eigenschaft mitgegeben. Damit wird folgende Funktion für die Nachbarschaft von Straßen und Radwegen bestimmt:

```

vars strasse_radweg ;
define strasse_radweg ( area1,area2 ) -> klasse ;
  vars klasse , area1,area2 ;
  undef -> klasse;
  if ( are_parallel (area1,area2) ->> val) == " yes " then
    if ( connection (area1,area2) ->> val) == " yes " then
      'positiv' -> klasse;
    elseif ( connection (area1,area2) ->> val) == " no " then
      'negativ' -> klasse;
    endif;
  elseif ( are_parallel (area1,area2) ->> val) == " no " then
    'negativ' -> klasse;
  endif;
enddefine;

```

Das heißt, die zu einer Straße gehörenden Radwege laufen parallel zu ihr und sind mit ihr verbunden (siehe auch Abbildung 9.12, links).

Über das Lernen von **part_of_relation** können die Methoden zur Aggregation von Objekten ermittelt werden. Nach der Eingabe der Beispiele, sowie den Angaben Relationenname, Nachbarkonzept und Zielkonzept wird die Relationsfunktion mit ID3 ermittelt und dem Objekt als neue Methode hinzugefügt. Im Beispiel wird die Relation zweier Straßen bestimmt, in der sich zu einer neuen Straße zusammenfassen lassen. Die Relation zwischen strasse3 und strasse1 wird als positives Beispiel angegeben, die zwischen strasse1 und strasse2 als negativ. Aus den gesammelten Beispieldaten wird mit ID3 die Relationsfunktion **strasse_strasse** bestimmt und diese in den Slot **part_of_relation** der erweiterten Objektklasse **strasse** eingetragen. Durch diese Erweiterung

verfügt die Klasse jetzt über die Möglichkeit der Aggregation (merge_me) mit anderen Straßenobjekten - sofern sie die geforderte Relation teilen.

Eine entsprechende Funktion kann für Feldobjekte abgeleitet werden. Somit können im nächsten Interpretationsschritt alle Feldobjekte aggregiert werden, welche die gegebene Relation mit einem weiteren Feld teilen. Zunächst werden die Felder feld8 und feld4 zusammengefaßt. Auch feld1 und feld2 lassen sich gemäß dem gelernten Kriterium aggregieren. Das hieraus entstandene Feld teilt nun die geforderte Relation mit den Feldern feld7 und feld5 und kann daher mit ihnen zusammengefaßt werden.

Haben alle Objekte ihre verfügbaren Methoden auf die Daten angewandt, so ist das Ergebnis die gesuchte Szenenbeschreibung (Abbildung 9.12, rechts). Die im Laufe der Interpretation erzeugten spezifischen Objekte,

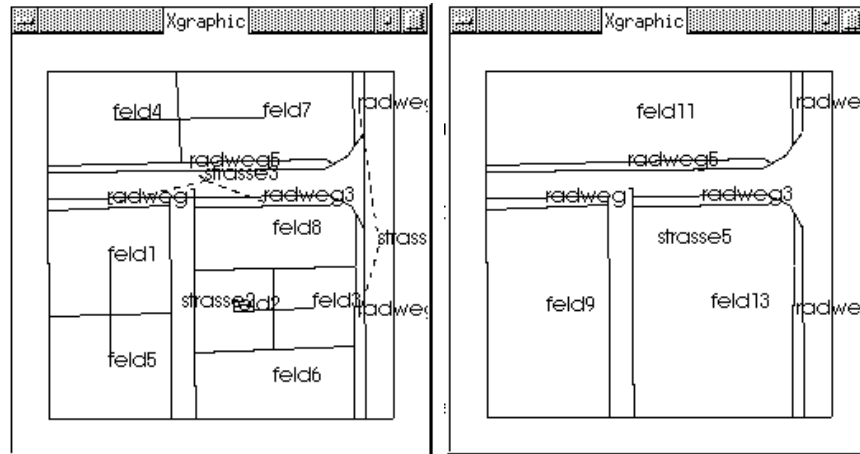


Abbildung 9.12: Darstellung der Assoziationen strasse_radweg (gestrichelt) sowie der Relation feld_feld (durchgezogen) (links); Ergebnis: Beschreibung der Szene auf Basis der ermittelten Konzepte (rechts)

sowie die generische Beschreibung der Objektklassen **feld** und **verkehr**, sowie deren Unterklassen **radweg** und **strasse** bilden das gesuchte Modell der Objekte (Abbildung 9.13). Die abgeleiteten Objektklassen und ihre

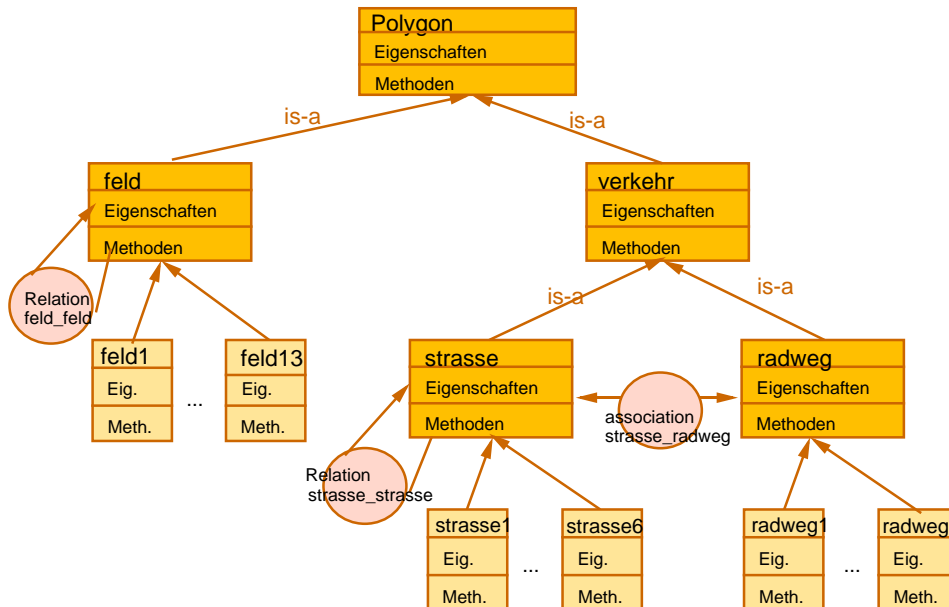


Abbildung 9.13: Abgeleitete Objekte und Relationen

Funktionen sind hier auszugsweise angegeben:

```

** <flavour strasse>
methoden : [recognize_object merge_me]
relationen : [[strstr positiv strasse strasse]]
    
```



```

association : [[strasse_radweg positiv radweg]]
-----
es existieren bisher folgende instanzen:
** [strasse5 strasse4 strasse3 strasse2 strasse1 strasse_proto]

strasse_proto:
contains_val [no]
contained_val [no]
object_elongation_val [4.445 1.110158]
object_lq_val [0.205 0.13435]
object_size_val [2673.84 1081.434969]
polytyp_val [rechteck]
num_points_val [6 2.828427]
num_right_angles_val [3 1.414214]
num_parallel_val [2 0.1]
num_same_length_val [6.5 6.363961]
object_form_val [H V]
junction_ell_val [2 2.828427]
junction_tee_val [9.5 0.707107]
junction_frk_val [0 0.1]
junction_arw_val [0 0.1]
junction_njn5_val [0 0.1]

** <flavour feld>
PRESENT_RESULTS: methoden      : [recognize_object merge_me]
PRESENT_RESULTS: relationen    : [[feld_feld positiv feld feld]]
PRESENT_RESULTS: association   : []
-----
es existieren bisher folgende instanzen:
** [feld13 feld12 feld11 feld10 feld9 feld8 feld7 feld6 feld5 feld4 feld3
    feld2 feld1 feld_proto]

feld_proto :
contains_val [no]
contained_val [no]
object_elongation_val [1.43 0.212132]
object_lq_val [0.635 0.318198]
object_size_val [4223.34 742.518689]
polytyp_val [rechteck]
num_points_val [5.5 2.12132]
num_right_angles_val [3.5 0.707107]
num_parallel_val [2 0.1]
num_same_length_val [1.5 0.707107]
object_form_val [H]
junction_ell_val [1 1.414214]
junction_tee_val [5 1.414214]
junction_frk_val [0 0.1]
junction_arw_val [0 0.1]
junction_njn5_val [0 0.1]

```

9.6 Automatische Erzeugung (Synthese) von Programmen für die Objekterkennung

Mit dem beschriebenen Programm wurde ein Modul zur automatischen Ableitung von Beschreibungen graphischer Bildobjekte geschaffen. Diese Ableitung erfolgt inkrementell aus elementaren geometrischen Objekten und somit datengetrieben (bottom-up). Die gegebenen graphischen Beispiele werden zunächst gruppiert, dann klassifiziert, d.h. ihnen wird eine Bedeutung zugewiesen. Die so erzeugten semantischen Objekte werden über eine Klassenbeschreibung spezifiziert. Diese abstrakte Klasse enthält Eigenschaften und Relationen der Objekte. Mit der Modellierung wurde also ein Generalisierungs- und Gruppierungsprozeß von elementaren geometrischen - hin zu komplexen semantischen Objekten durchgeführt.

Diese Modelle lassen sich nun umgekehrt in einem top-down-Ansatz für die Erzeugung von Programmen zur Objekterkennung in Bildern einsetzen. Solche Programme gehen von einem bestimmten Aufgabenziel aus und erzeugen die dafür nötigen Programmschritte (vgl. z.B. [Messer 1992], [Draper, Hanson & Riseman 1991], [Hansen & Henderson 1989]). Im Unterschied zur realisierten Aufgabe geht es bei der Programmsynthese darum, aus der Kenntnis einer bestimmten Aufgabenstellung eine gezielte Strategie zu deren Lösung abzuleiten. Im Falle der Objekterkennung ist dies beispielsweise die gezielte Suche nach bestimmten vorgegebenen Objekteigenschaften. Prinzipiell handelt es sich dabei um ein Problem der Suche nach einer geeigneten Strategie (vgl. Kap. Prozedurale Modelle 3.4). Dazu muß u.a. festgelegt werden, nach welchem Merkmal zunächst gesucht werden soll, bzw. welche Routinen zu dessen Extraktion bzw. Erkennung einzusetzen sind.

Für die Programmsynthese ist die in der Modellierung erfolgte Abstraktion wieder rückgängig zu machen, bzw. aufzulösen. Abbildung 9.14 soll diesen Zusammenhang erläutern. Während auf der linken Seite eine inkrementelle bottom-up Objektmodellierung stattfindet, geht es auf der rechten Seite um die modellgetriebene Interpretation von Bildern; in der Mitte ist die Abfolge zur Erzeugung der Objekterkennungsstrategie dargestellt. Sie stellt das Interpretationsmodell [Braun et al. 1994] dar, welches den Ablauf der Bildererkennung festlegt. Ausgangspunkt

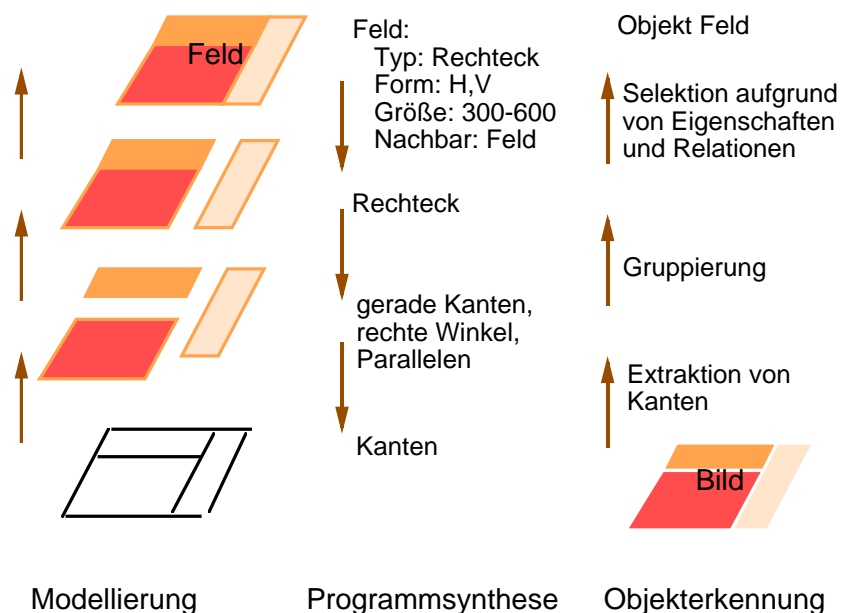


Abbildung 9.14: Modellierung, Programmsynthese und Interpretation

ist eine Analyse der Eigenschaften und Relationen der jeweiligen Objektklasse (d.i. des Modells). Die im Modell beschriebenen Objekte müssen in ihre erkennbaren Einzelteile unterteilt werden. Für ihre jeweilige Detektion sind Routinen bereitzustellen (vgl. z.B. [Förstner 1994], [Fuchs & Förstner 1995], [Etemadi, Schmidt, Matas, Illingworth & Kittler 1991]). Der Vorteil der hier erzeugten Modelle besteht darin, daß die verwendeten Merkmale auch in den Bildern extrahiert werden können: Kanten, gerade Kantenstücke, gruppierte Objekte, Relationen, ... Auf der Basis dieser extrahierten symbolischen Bildbeschreibung kann die weitere Verarbeitung erfolgen.

Aus der Objektbeschreibung können die Art der Objektteile, sowie deren Defaultwerte entnommen werden (z.B. Polygontyp, -form, -größe). Die Angabe, daß es sich um ein polygonales Objekt handelt, kann eine Routine zur Extraktion gerader Kantenstücke triggern. Anschließend muß eine weitergehende Organisation dieser

Kantenstücke (Kollinearität, Parallelität, Rechtwinkligkeit) erfolgen. Hierbei werden gezielt solche Polygone gruppiert, die die gegebenen Bedingungen erfüllen. Das Ergebnis dieses Schrittes sind alle Regionen im Bild, welche Kandidaten für das gesuchte Objekt darstellen, da sie alle Eigenschaften des Modells erfüllen. Zusätzlich können die Relationen, die für die Objekte gelten, ausgenutzt werden, um schließlich eine weitere Einschränkung möglicher Objekte durchzuführen. Auf diese Weise ist erreicht, daß eine konsequente Top-Down Prozessierung ausgehend von den Objektmodellen die Segmentierung gezielt steuert²⁷.

Beispielhaft seien die Schritte für die Erkennung von Straßen mit dem eben bestimmten Modell hier angedeutet: Zur Detektion von Straßen wird zunächst nach Objekten des geforderten Polygontyps - nämlich Rechteck - gesucht. Dazu müssen gerade Kantenstücke extrahiert werden (z.B. mit dem Algorithmus von Burns & Riseman [1986]). Diese werden zu Rechtecken gruppiert, die eine bestimmte Größe besitzen. Diese Einschränkung kann schon für die Gruppierung genutzt werden (indem nur Kantenstücke verwendet werden, die kürzer sind, als die geforderte Kantenlänge). Die Association zwischen einer Straße und einem Radweg, beschrieben in der Funktion `strasse_radweg`, kann zur weiteren Interpretation eingesetzt werden. Sie besagt, daß Straßen parallel zu Radwegen verlaufen. Hierzu wird vorausgesetzt, daß bereits in ähnlicher Weise mit Hilfe der Objektklasse `radweg`, Radwege segmentiert worden sind. Eine Straßenhypothese kann daher auf der Basis dieser Relation zu Radwegen bekräftigt werden.

Hiermit wurde lediglich ein möglicher Weg der Nutzung der gewonnenen Modelle für die Bildinterpretation skizziert. Eine konkrete Realisierung bedarf weiterer Untersuchungen. So sind insbesondere die Unsicherheiten bei der Merkmalsextraktion zu berücksichtigen. Wenngleich die Merkmale grundsätzlich beobachtbar sind (z.B. Linienlänge, Parallelität, Anzahl der TEE-Knoten), stellt sich doch die Frage, ob diese Merkmale in realen, segmentierten Bildern auch in dieser "reinen" Form vorkommen, wie sie aus den idealen Beispielen gelernt worden sind, bzw. wie ev. Unterschiede zu verarbeiten sind.

Wissensbasierte Bildanalyzesysteme wie z.B. KBVision bieten prinzipiell eine Verknüpfung von high- und low-Level-Verarbeitungsschritten. Auf einer abstrakten Ebene können Objekte formal über sog. Schemata beschrieben werden. In einem Blackboard-Ansatz werden anschließend die möglichen Hypothesen in konkurrierender Weise abgearbeitet. Idealerweise könnte die gelernte Modellbeschreibung direkt in ein Schema übertragen werden, welches dann für die Objekterkennung genutzt wird [Eggleston 1989], [Adam 1994].

²⁷Ideal wären Programme, die bereits komplexe Objekte in Bildern detektieren können, und somit direkt von der Modellbeschreibung aus angestoßen werden können, z.B.: "Suche Rechtecke mit einem Länge-zu-Breite-Verhältnis von 1:1.5, deren Flächeninhalt zwischen 100 und 500 liegt", "Suche alle geraden Kanten mit Richtung 43 Grad".

10 Beispiele für inkrementelle, interaktive Objektmodellierung

Anhand folgender Beispiele soll veranschaulicht werden, wie das vorgestellte Verfahren zur Modellierung eingesetzt werden kann. Es handelt sich bei den zu modellierenden Konzepten zwar um einfache geometrische Objekte und ihre Relationen, allerdings läßt sich nicht a priori festlegen, welche Merkmale die Objekte im jeweiligen Fall besonders kennzeichnen. Insbesondere trifft dies für die Bestimmung der Relationen zwischen den Objekten zu. Diese sind - wie gezeigt werden wird - stark objekt- und kontextabhängig, und lassen sich daher nicht allgemein angeben. Auch für die Ableitung rekursiver Strukturen wird diese Art der Verarbeitung genutzt.

10.1 Feldmodelle zur Landnutzungsklassifizierung

Die Landnutzungsklassifizierung zielt darauf ab, eine Einteilung des Geländes in verschiedene Nutzungsklassen zu erhalten. Solche Erhebungen sind für die Landwirtschaftsämter wichtig. Hiermit können Statistiken über die landwirtschaftlich genutzten Flächen oder auch die Versiegelung der Erdoberfläche aufgestellt werden.

Durch die Fernerkundung - meist mittels Satellitendaten - erfuh die Methode einen starken Auftrieb, da es hiermit möglich ist, solche Ergebnisse großflächig zu ermitteln. Herkömmliche Ansätze basieren auf einer Multi-spektralklassifizierung, das heißt, eine Einteilung in verschiedene Nutzungsklassen wird auf der Basis von Grauwerten (meist 3-er Farbwerte) durchgeführt. Das zugrundeliegende Modell der Objekte besteht also lediglich aus diesen drei Werten. Mit dieser Information allein wird über die Zugehörigkeit zu bestimmten Objektklassen (Weizen, Grünland, Bebauung, ...) entschieden. Ein großer Nachteil dieser Verfahren ist das Vernachlässigen der Nachbarschaftsinformation und weiterer Objekteigenschaften, wie z.B. der Objektgeometrie. Diese kann zu Fehlklassifikationen führen, speziell an Feldrändern, wo häufig aufgrund des unterschiedlichen Spektralverhaltens Mischsignaturen auftreten. Versuche, diesem abzuweichen, gehen in die Richtung, die Geometrie der Feldstrukturen, also Information über die Feldgrenzen mit einzuführen. Janssen [1993] berichtet über ein Verfahren, welches die Information über rechtliche Grenzen einem GIS entnimmt und diese nutzt, um die Klassifikation zu stützen. Dieses Verfahren setzt jedoch voraus, daß Geometriedaten vorhanden sind, was selbst in gut vermessenen Gebieten nicht immer gegeben ist. Eine weitere Einschränkung stellt dar, daß die Information im GIS den rechtlichen Bestand widerspiegelt, welcher nicht immer den (sichtbaren) Nutzungsgrenzen entspricht. Daher entstand die Idee, ein allgemeines Modell für Felder bzw. Feldstrukturen zu entwickeln, welches nicht nur für eine bestimmte Feldeinheit Gültigkeit besitzt, sondern in einem größeren Bereich anwendbar ist.

Felder sind Objekte generischen Typs. Der Begriff 'Feld' bezeichnet eine ganze Klasse von Individualobjekten. Ein Einzelfeld ist zwar einfach zu beschreiben, die individuelle Vielfalt verhindert jedoch, daß explizit alle vorkommenden Objekte beschrieben werden können. Zunächst mag die Feldmodellierung simpel oder gar überflüssig erscheinen: Felder lassen sich im allgemeinen als Rechtecke beschreiben. Diese Beschreibung ist in der Tat einfach, sie läßt sich allerdings auf sehr viele von Menschenhand geschaffene bzw. künstliche Objekte unserer Umgebung anwenden, da diese meist regelmäßige, einfache Formen besitzen. Im Falle der Feldform ist die Einfachheit bedingt durch ihre Funktion: geht es darum, mit Maschinen ein Feld zu bearbeiten, so ist die rechteckige Form die einzig sinnvolle. Jede zusätzliche Ecke bringt erhöhten Arbeitsaufwand mit sich. Die Betrachtung eines einzelnen rechteckigen Objekts läßt mehrere Interpretationen offen. Was aber ein Feld-Objekt besonders kennzeichnet, und auch als solches auszeichnet, ist seine Umgebung. Erst durch Analyse des Kontexts wird es möglich, ein Feld von einer Dachfläche zu unterscheiden, bzw. von einer Straße. Diese Kontextinformation in das Modell mit zu integrieren, ist Teil vorliegender Arbeit.

Betrachtet man eine größere Feldstruktur, so wird deutlich, daß sie nicht aus einer willkürlich angeordneten Sammlung von Einzelfeldern (Rechtecken oder Polygonen) besteht, sondern eine starke Regelmäßigkeit aufweist. Diese gleichförmige Struktur ist auf das planerische Prinzip zurückzuführen, welches bei der Entstehung der Felder waltete. Die Frage ist allerdings, wie sich diese offensichtliche Struktur in ein Modell überführen läßt. Eine diffuse Beschreibung würde lauten: "mehrere gleichgroße Rechtecke, die nebeneinander angeordnet" sind. Da sich diese Beschreibung offensichtlich nicht einfach angeben läßt, wird hierfür das Lernverfahren genutzt.

Die Tatsache, daß Felder durch eine planerische Handlung entstanden sind, läßt sich nutzen, um ihre Form und ihre Beziehungen zu anderen Feldern charakterisieren. In vorliegendem Ansatz wird versucht, die zugrundeliegende Planung automatisch (aus gegebenen Beispielen) zu rekonstruieren, um so die innere Struktur - die sich ja nach außen widerspiegelt - offenzulegen. Das bedeutet, die Entstehungsgeschichte eines Feldes ist aus

der sichtbaren Information wiederherzustellen. Es wird also versucht, den Kontext, in den das Objekt 'Feld' eingebettet ist, über seine 'Historie' zu ermitteln. Wie gezeigt werden wird, ist diese direkt aus rein geometrischen Bedingungen ableitbar. Mit der Ableitung der Entstehung einer Feldstruktur ist der Kontext, in den ein Einzelfeld eingebettet ist, festgelegt. Damit liegen auch die Relationen zwischen Einzel(feld)objekten fest.

Das Beispiel der Abbildung 10.1 ist ein Ausschnitt einer Szene, wie sie für einen Test der ISPRS Working Group III/3 zur Verfügung gestellt worden ist [Fritsch, Sester & Schenk 1994]. Die Flurstücksgrenzen wurden aus einer Flurkarte digitalisiert. Im Lern- und Interpretationsschritt werden zunächst die Polygone ermittelt.



Abbildung 10.1: Zu diesem Bild existieren digitalisierte Flurstücke, Straßen und Wegraine

Anschließend wird das Lernverfahren angestoßen. Typische Feldbeispiele werden markiert; auch Straßen und Wegraine werden gezeigt. Nun können hieraus die diskriminierenden Objektfunktionen für Flurstücke, Straßen und Raine bestimmt werden. Es zeigt sich, daß mit diesen Funktionen bereits alle der dargestellten Polygone klassifiziert werden können (Abbildung 10.2). Die allgemeine Polygonfunktion lautet:

```
vars polygon_object ;
define polygon_object ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if ( contains (area) ->> val) == " yes " then
    'strasse' -> klasse;
  elseif ( contains (area) ->> val) == " no " then
    if <= ( ( object_elongation (area) ->> val) , 6.175 ) then
      'flurstueck' -> klasse;
    elseif > ( ( object_elongation (area) ->> val) , 6.175 ) then
      'rain' -> klasse;
    endif;
  endif;
enddefine;
```

Die ermittelte Funktion besagt, daß Objekte, die weitere Objekte enthalten, als Straßen klassifiziert werden. Ansonsten sind es - entsprechend dem Wert ihrer Kompaktheit entweder Raine oder Flurstücke. Die etwas sonderbare Straßendefinition ergibt sich aus der speziellen Darstellung der Straßen in dieser Karte: durch die Ausbuchtung der Straßen in der Mündung, entsteht ein konkaves Polygon. Die Funktion "contains" allerdings überprüft, ob sich weitere Objekt in der konvexen Hülle des gegebenen Objekts befinden. Dies ist im Falle der Straßen so: dort liegen nämlich die Wegraine.

Diese neue Darstellung besteht nun aus einer Aufzählung der vorkommenden Felder und Straßen. Somit wurde eine objektbezogene Beschreibung gewonnen. Eine weitere Strukturierung kann erreicht werden, indem einzelne Objekte als Teile einer größeren Struktur definiert und zu neuen Objekten zusammengefaßt werden. Diese



Abbildung 10.2: Straßen, Flurstücke und Wegraine wurden korrekt klassifiziert

Aggregation einzelner Objekte kann aufgrund bestimmter Relationen vollzogen werden. Im Falle der Feldstrukturen lassen sich solche Felder zusammenfassen, die zur gleichen Zeit durch Unterteilung einer übergeordneten Fläche entstanden sind. Die Frage stellt sich, an welchen (geometrischen oder physikalischen, also sichtbaren) Eigenschaften der Einzelfelder sich diese Historie der Unterteilung ableiten läßt. Diese Eigenschaften lassen sich nicht ohne weiteres von vornherein bestimmen, andererseits können aber genügend Beispielobjekte angegeben werden, die diese erfüllen. Mithilfe eines Lernverfahrens werden die Eigenschaften der Objekte, die für das Zusammenfassen sprechen, aus den Beispielen ermittelt - und müssen daher nicht a priori vorgegeben werden. Die oben angesprochene Relation, die sich aufgrund der Historie der Feldunterteilung ergibt, wird als part-of-Beziehung gelernt. Dazu werden benachbarte Felder markiert, und deren Beziehung als positives Beispiel gewertet, Felder, die nicht unmittelbar aggregierbar sind, gelten als negatives Beispiel. Aus den Merkmalen der Beispieldaten werden im Lernprozeß automatisch diejenigen ermittelt, die anzeigen, daß diese Felder zusammengehören. Das Lernverfahren erzeugt somit eine Relationsfunktion, die es den Feldern als zusätzliche Eigenschaft beibringt (part-of-relations). Aufgrund dieser verfügen die Felder nun über die Methode der Aggregation **merge-me**. Die Relation, die aus den gegebenen Beispielen ermittelt wurde besagt, daß Flurstücke dann zusammengefaßt werden dürfen, wenn sie miteinander verbunden sind.

```

vars fefe ;
define fefe ( area1,area2 ) -> klasse ;
  vars klasse , area1,area2 ;
  undef -> klasse;
  if ( connection (area1,area2) ->> val) == " no " then
    'negativ' -> klasse;
  elseif ( connection (area1,area2) ->> val) == " yes " then
    'positiv' -> klasse;
  endif;
enddefine;

```

In Abbildung 10.3 (links) sind alle Flurstücke angegeben, die die Relation miteinander teilen (Flurstücke sind durch Linien verbunden). Wie sich dies neu erworbene Methode auf die Daten auswirkt kam im nachfolgenden Interpretationsschritt beobachtet werden, bei dem alle Feldobjekte nun versuchen, ihre neue Methode anzuwen-

den. Zunächst suchen die Felder nach Nachbarn im Sinne der eben gelernten Relation. Mit diesen kann dann ein Zusammenschluß erfolgen. Das Ergebnis der Interpretation ist also folgende Darstellung, bei der alle Felder zu größeren Einheiten gruppiert worden sind (Abbildung 10.3 (rechts)). Damit ist eine strukturierte Beschrei-

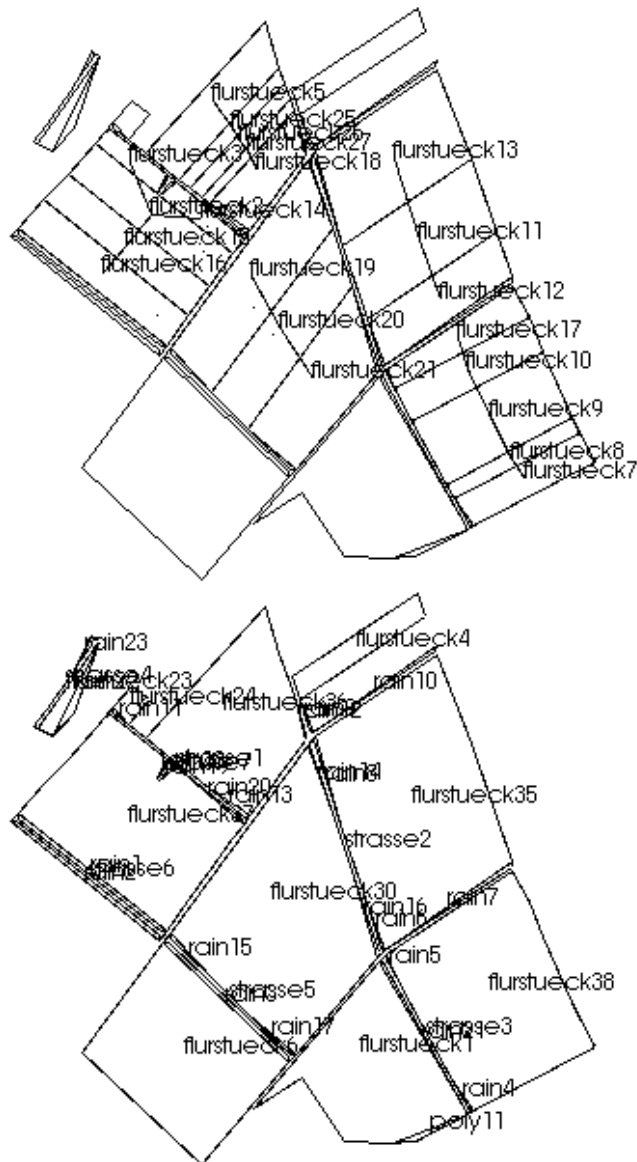


Abbildung 10.3: Aggregierbare Flurstücke (links); Ergebnis der Interpretation (rechts)

bung dieser Szene erreicht. Diese besteht zunächst aus der spezifischen Darstellung der einzelnen Objekte Felder und Straßen. Von Bedeutung ist allerdings die generische Klassenbeschreibung, welche in der Beschreibung des Objektprototyps zum Ausdruck kommt: hier sind die Eigenschaften und Relationen aller Objektinstanzen zusammengefaßt. Analog zum Beispiel von Seite 88 sind auch hier die jeweiligen Klassenbeschreibungen und Prototypen ermittelt worden.

In Sester [1992] wird vorgeschlagen, über eine statistische Analyse die Regelmäßigkeiten in der Generalisierung der Objekte festzustellen. Diese werden anschließend in Form einer stochastischen Grammatik repräsentiert. Hiermit kann insbesondere die rekursive Struktur der Felder modelliert werden.

10.2 Modelle für kartographische Objekte

Geo-Informationssysteme dienen der Erfassung, Verarbeitung, Analyse und Präsentation raumbezogener Daten. Mit dieser Funktionalität öffnet sich ein breites Anwendungsspektrum für solche Systeme, welche von Kataster, Umweltorganisationen, Energieversorgungsunternehmen, u.a. gleichermaßen genutzt werden. Elementare

Grundlage ist allerdings die Verfügbarkeit der digitalen Daten. Die Erfassungsmethoden der Geodäsie und der Photogrammetrie werden dafür genutzt. Eine naheliegende Möglichkeit, sehr schnell digitale Daten bereitzustellen, ist die Nutzung bestehender Kartenwerke. Diese können manuell digitalisiert werden - was allerdings einen enormen personellen Aufwand darstellt. Daher gibt es eine Fülle von Ansätzen, diesen Vorgang mit Methoden der Mustererkennung und Bildanalyse zu automatisieren (vgl. z.B. [Illert 1990]). Mayer [1994] stellt einen Ansatz vor, der auf explizit vorgegebenen Modellen der Szene basiert. Das Wissen umfaßt die Einzelobjekte, und deren Relationen untereinander. Diese sind a priori in einem Modell festgeschrieben, welches manuell erstellt wurde. Der Autor führt denn auch als Nachteil an, daß der Modellierungsaufwand für die Interpretation sehr hoch ist.

Das vorliegende Verfahren kann für die Modellierung sehr sinnvoll eingesetzt werden. Dies wird am Beispiel einer DGK 1:5000 gezeigt. Abbildung 10.4 gibt einen geschnittenen Ausschnitt einer solchen Karte an. Im Bild lassen sich die Objekte Straßen, Flurstücke, und Häuser unterscheiden. Aus dem Gebiet wurden zwei Ausschnitte

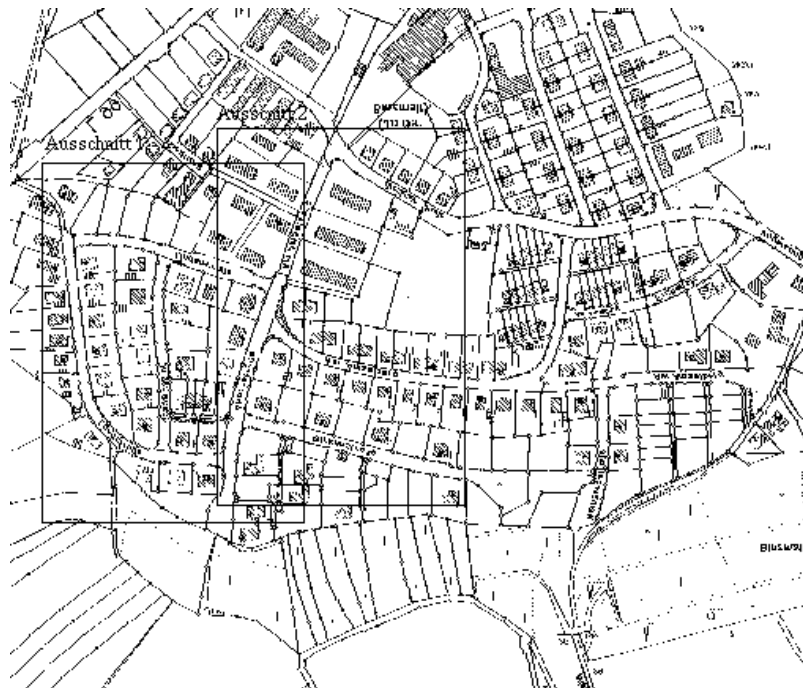


Abbildung 10.4: Ausschnitt aus einer DGK5 - Gebiet Gengenbach

herausgegriffen und digitalisiert. Mit dem ersten Ausschnitt wurde das System trainiert, während der zweite Ausschnitt als Testgebiet für das Verfahren gilt.

Trainingsphase

Zunächst wird die Analyse und Modellierung im ersten Gebiet durchgeführt. Nachdem aus der Liniendarstellung Polygone abgeleitet wurden, kann der Lernschritt beginnen. Es werden 3 Klassen von Objekten spezifiziert, indem einzelne Beispiele angegeben werden. Dies sind zunächst die in Abbildung 10.5 (links) hervorgehobenen Objekte. ID3 ermittelt aus den Merkmalen dieser Objekte eine Unterscheidungsfunktion, die es im anschließenden Interpretationsschritt auf die Polygone anwendet. Abbildung 10.5 (rechts) zeigt einen Ausschnitt während des Interpretationslaufs, bei dem nach und nach alle polygonalen Objekte eine in die 3 Klassen eingeteilt werden. Mit den gegebenen Funktionen lassen sich fast alle Objekte eindeutig klassifizieren. Es zeigt sich jedoch, daß einige Flurstücke nicht klassifiziert werden konnten (siehe Abbildung 10.6). Der Blick auf die ermittelte Funktion und die vorgegebenen Beispiele erklärt dies allerdings. Zufälligerweise wurden für das Lernen nur solche Flurstücke gezeigt, auf welchen ein Haus steht.

```
vars polygon_object ;
define polygon_object ( area ) -> klasse ;
vars klasse , area ;
undef -> klasse;
```

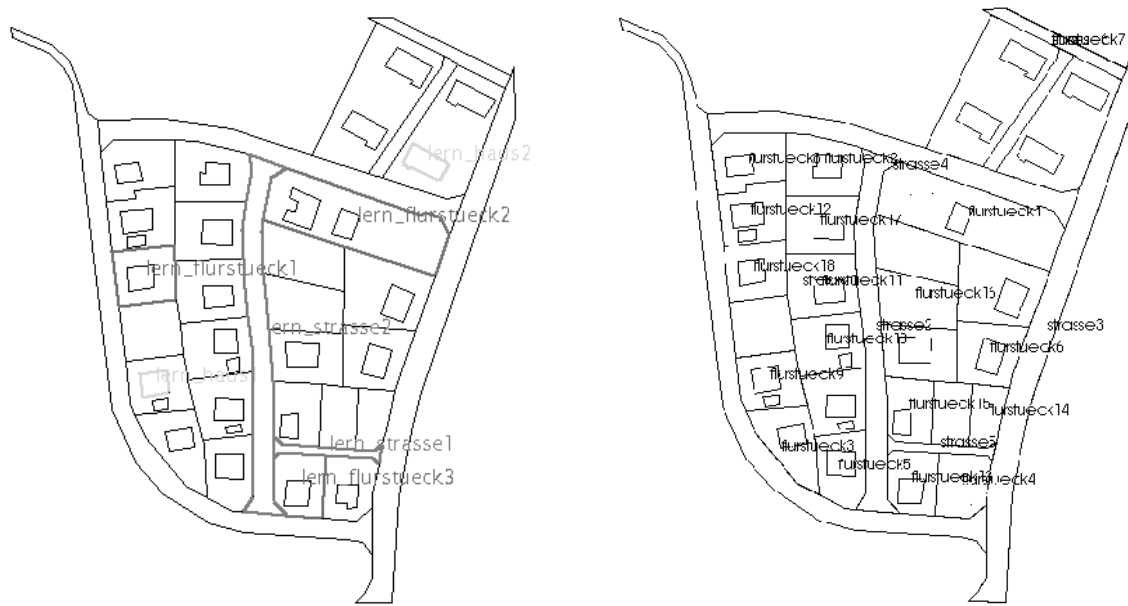



Abbildung 10.5: Vorgegebene Objekte: Straßen, Flurstücke und Häuser (Die einzelnen Objektklassen sind durch unterschiedliche Grauwerte visualisiert) (links); Ablauf der Interpretation (rechts)

```

if ( contains (area) ->> val) == " yes " then
  'flurstueck' -> klasse;
elseif ( contains (area) ->> val) == " no " then
  if ( contained (area) ->> val) == " yes " then
    'haus' -> klasse;
  elseif ( contained (area) ->> val) == " no " then
    'strasse' -> klasse;
  endif;
endif;
enddefine;

```

Daher konnte dies als besonderes Unterscheidungsmerkmal von ID3 genutzt werden. Die Funktion sagt aus, daß ein Objekt ein Flurstück ist, wenn es eine andere Fläche enthält; andernfalls muß unterschieden werden, ob das Objekt selbst in einer Fläche enthalten ist - dann ist es ein Haus - oder nicht - dann wird es der Klasse Straße zugeordnet. Abbildungen 10.6 und 10.7 zeigen das vorläufige Ergebnis der Interpretation aufgrund dieser Unterscheidungsfunktion.

Werden jetzt zusätzlich noch einige der nicht klassifizierten Polygone (Flurstücke ohne Haus) spezifiziert, so wird eine neue Funktion ermittelt, die nun für alle Objekte korrekte Klassifizierungen liefert. Diese neue Funktion überprüft für Objekte, die nicht in anderen enthalten sind (und somit keine Häuser sind) die Kompaktheit, d.h. die Länglichkeit ab. Kompaktere Objekte werden den Flurstücken, alle anderen den Straßen zugeordnet.

```

vars polygon_object ;
define polygon_object ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if ( contained (area) ->> val) == " yes " then
    'haus' -> klasse;
  elseif ( contained (area) ->> val) == " no " then
    if <= ( ( object_elongation (area) ->> val) , 2.4 ) then
      'flurstueck' -> klasse;

```

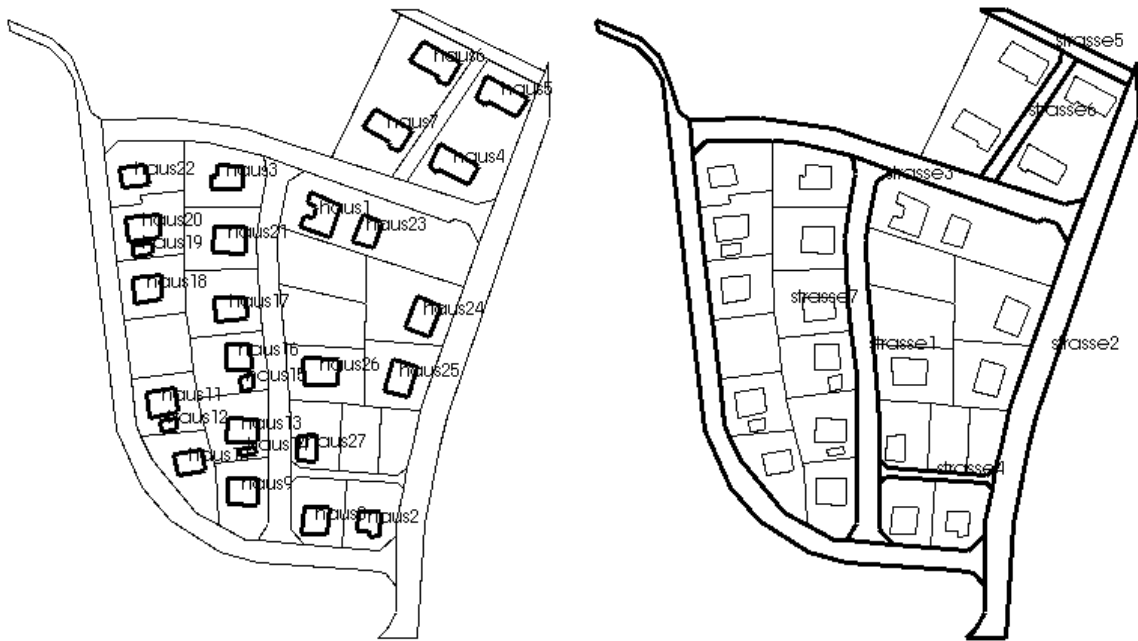


Abbildung 10.6: Erste Interpretation: alle Häuser (links) und Straßen (rechts) wurden richtig klassifiziert

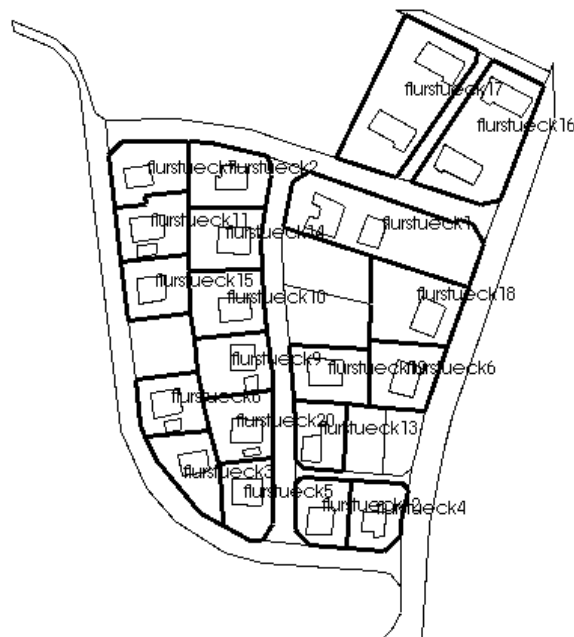


Abbildung 10.7: Erste Interpretation: einige Flurstücke konnten nicht klassifiziert werden

```

elseif > ( ( object_elongation (area) ->> val) , 2.4 ) then
    'strasse' -> klasse;
endif;
endif;
enddefine;

```

Nachdem alle Objekte klassifiziert sind, können Relationen abgeleitet werden. Zunächst soll so der Zusammenhang zwischen einem Haus und einem Flurstück offengelegt werden. Dazu wird die Relation “association” genutzt. Beispiele und Gegenbeispiele für ein Hausobjekt und sein umgebendes Grundstück können vorgegeben werden. Die aus diesen Beispielen bestimmte Relationenfunktion wird an das Objekt Haus gegeben. Die Relation ergibt sich einfach aus der Abprüfung, ob sich das Haus innerhalb des Flurstücks befindet. Hiermit erweitert sich die Beschreibung der Klasse `haus`, indem sie eine zusätzliche Assoziation erhält.

Weiterhin läßt sich eine Spezialisierung der Hausobjekte durchführen. In der Karte können Einfamilienhäuser, Reihenhäuser, sowie Garagen unterschieden werden. Die zur Unterscheidung nötigen Merkmale werden wiederum gelernt. Dies geschieht in analoger Weise wie bei der Spezialisierung der Polygone, indem Beispiele gezeigt und automatisch Unterklassen der Klasse `haus` gebildet werden. Als Unterscheidungsfunktion für die verschiedenen Haustypen ergibt sich:

```
vars haus_object ;
define haus_object ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if haus(area) then          ;;; Ueberpruefen der Oberklasse
    if > ( ( object_size (area) ->> val) , 398.72 ) then
      if <= ( ( object_lq (area) ->> val) , 0.7 ) then
        'reihen' -> klasse;
      elseif > ( ( object_lq (area) ->> val) , 0.7 ) then
        'einfam' -> klasse;
      endif;
    elseif in_interval ( ( object_size (area) ->> val) , [ 100.31 398.72 ] ) then
      'einfam' -> klasse;
    elseif <= ( ( object_size (area) ->> val) , 100.31 ) then
      'garage' -> klasse;
    endif;
  endif;
enddefine;
```

Das bedeutet, daß sich Gebäude primär durch ihre Größe und ihre Form unterscheiden lassen. Sehr kleine Objekte werden als Garage klassifiziert, größere können entsprechend Einfamilien- oder Reihenhäuser sein.

Wichtig ist, daß die Unterklassen von `haus` die Eigenschaften ihrer Oberklasse erben, also insbesondere die bereits gelernten Relationen.

Desweiteren können die Flurstücke zu größeren Einheiten aggregiert werden. Nach der Vorgabe der Beispiele wird die Relation abgeleitet und die Objektklasse `flurstück` wird somit erweitert. Die zugehörige Funktion lautet:

```
vars flur_flur ;
define flur_flur ( area1,area2 ) -> klasse ;
  vars klasse , area1,area2 ;
  undef -> klasse;
  if <= ( ( common_stem (area1,area2) ->> val) , 1 ) then
    'negativ' -> klasse;
  elseif > ( ( common_stem (area1,area2) ->> val) , 1 ) then
    'positiv' -> klasse;
  endif;
enddefine;
```

Die Funktion gibt an, daß Flurstücke, die mehr als 1 gemeinsamen TEE-STEM-Knoten besitzen, aggregierbar sind. Abbildung 10.8 (links) zeigt, welche Flurstücke die vorgegebene Relation teilen (angedeutet durch eine Verbindungslinie zwischen den Flurstücken); rechts ist eine Zwischenstufe im Ablauf der Aggregation dargestellt: durch das Zusammenfassen einzelner Flurstücke kann sich ergeben, daß die neu entstandenen ebenfalls die Relation mit anderen teilen und somit wiederum aggregierbar sind. Abbildung 10.9 zeigt das Ergebnis der Interpretation.



Abbildung 10.8: Flurstücke, die Relation **flur-flur** teilen (links); Aggregation von Objekten (rechts)

Testphase

Diese gelernten Modelle werden nun zur Verifikation auf den zweiten Ausschnitt angewandt. Es zeigt sich, daß die Objekte auf der ersten Stufe unmittelbar erkannt werden, d.h. eine Klassifizierung nach Flurstücken, Häusern und Straßen erfolgt direkt. Die weitere Klassifizierung der Häuser in ihre drei Unterklassen ist allerdings nicht direkt möglich. Während zwar alle Einfamilienhäuser identifiziert werden, trifft dies nicht auf die Reihenhäuser und Garagen zu. Daher muß hier weiter gelernt werden. Die Eingabe zusätzlicher Beispiele führt allerdings dazu, daß der Entscheidungsbaum sehr groß wird.

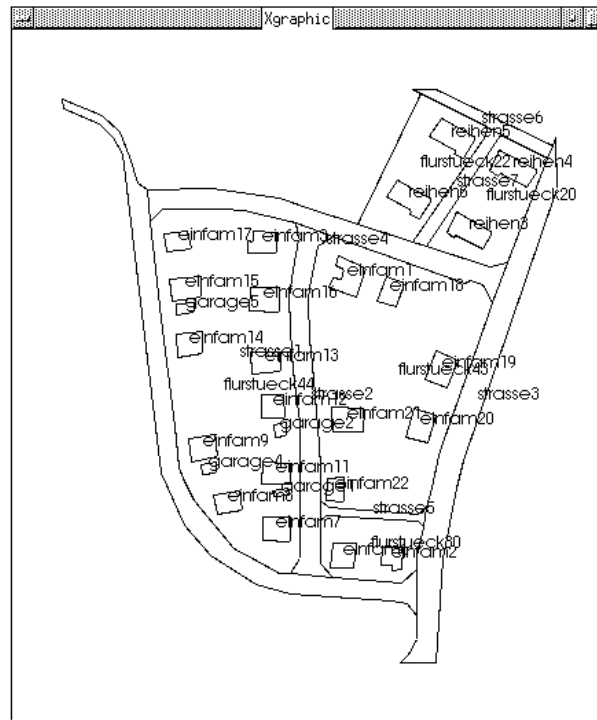


Abbildung 10.9: Ergebnis: vollständige Szeneninterpretation

```

vars haus_object ;
define haus_object ( area ) -> klasse ;
  vars klasse , area ;
  undef -> klasse;
  if haus(area) then
    ;;; Ueberpruefen der Oberklasse
    if <= ( ( object_size (area) ->> val ) , 227.38 ) then
      'garage' -> klasse;
    elseif > ( ( object_size (area) ->> val ) , 227.38 ) then
      if ( object_form (area) ->> val ) == " V " then
        'einfam' -> klasse;
      elseif ( object_form (area) ->> val ) == " H " then
        if <= ( ( object_elongation (area) ->> val ) , 1.53 ) then
          if ( polytyp (area) ->> val ) == " rechteck " then
            'einfam' -> klasse;
          elseif ( polytyp (area) ->> val ) == " polygon " then
            'reihen' -> klasse;
          endif;
        elseif > ( ( object_elongation (area) ->> val ) , 1.53 ) then
          'reihen' -> klasse;
        endif;
      endif;
    endif;
  endif;
enddefine;

```

Hier zeigt sich auch ein grundlegendes Problem des Programms ID3. Der Unterscheidungsbaum ist gut interpretierbar, solange er klein bleibt und nur wenige Verzweigungen aufweist. Der hier ermittelte Baum ist zwar für das gegebene Beispiel gültig, allerdings scheint er aufgrund der vielen Verzweigungen sehr speziell. Generell läßt sich ein einfacherer Baum eher auf unbekannte Objekte anwenden als ein solcher komplexer. Auch was die Aggregation der Flurstücke angeht, konnte hier lediglich ein Teil der Objekte zusammengefaßt werden. Für die übrigen war die geforderte Relation nicht gegeben. Offensichtlich haben die übrigen Flurstücke andere Knotenverbindun-

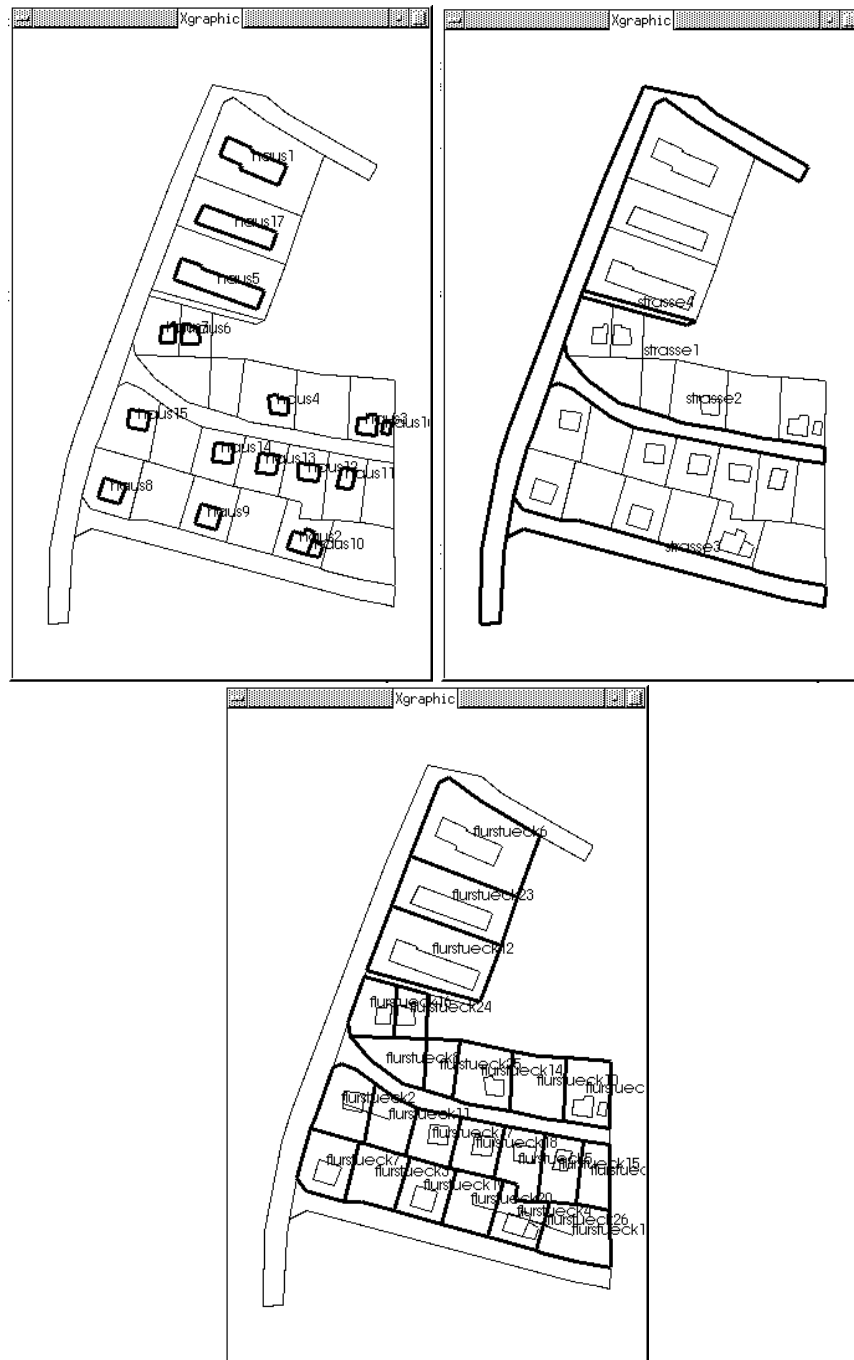


Abbildung 10.10: Interpretation der Testdaten: Häuser, Straßen und Flurstücke wurden korrekt klassifiziert

gen als die geforderten gemeinsamen TEE-STEM-Knoten. Daher muß hier eine neue Relationsfunktion gelernt werden:

```

vars flur_flur ;
define flur_flur ( area1,area2 ) -> klasse ;
  vars klasse , area1,area2 ;
  undef -> klasse;
  if <= ( ( common_sides (area1,area2) ->> val) , 0.988604 ) then
    'negativ' -> klasse;
  elseif > ( ( common_sides (area1,area2) ->> val) , 0.988604 ) then
    if <= ( ( common_ell (area1,area2) ->> val) , 0 ) then
      'positiv' -> klasse;
    elseif > ( ( common_ell (area1,area2) ->> val) , 0 ) then

```

```

    'negativ' -> klasse;
  endif;
endif;
enddefine;

```

Mit dieser ist es schließlich möglich, die Flurstücke zu aggregieren, um somit als Endergebnis die interpretierte Gesamtszene zu erhalten (Abbildung 10.2 (rechts)).

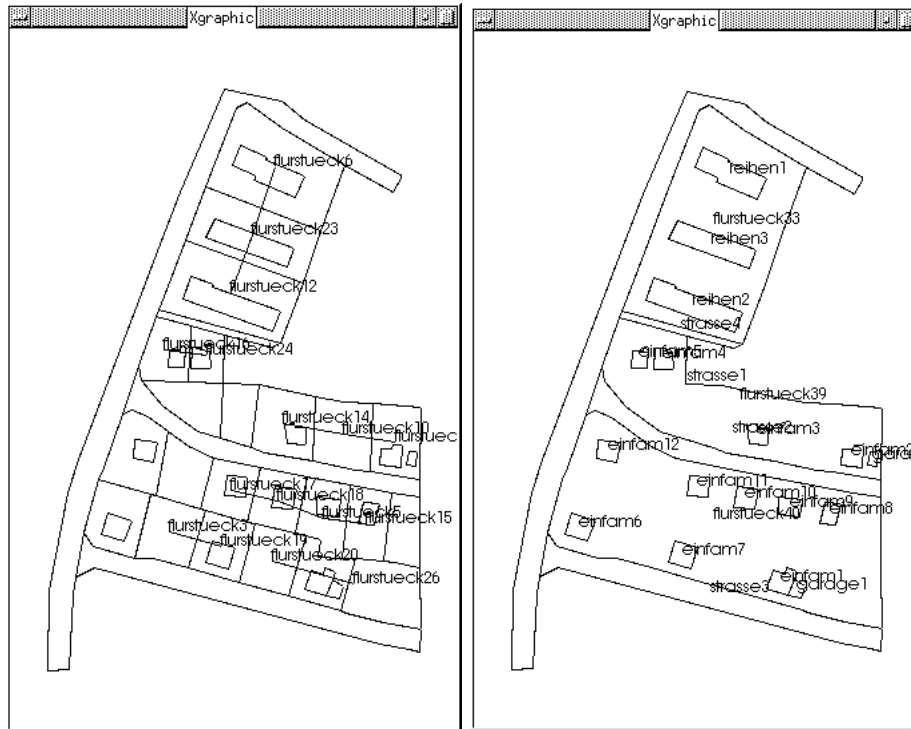


Abbildung 10.11: Flurstücke, die die Relation **flur_flur** teilen (links); Ergebnis der Interpretation (rechts)

10.3 Diskussion der Beispiele

Das Verfahren leitet Objektmodelle aus gegebenen Beispielen ab. Der Gewinn dieser neuen Beschreibung liegt in der Qualität der Objektmodelle: diese sind objektbezogen und strukturiert und sie liegen zudem direkt in einer rechnerinternen Darstellung vor.

Das gelernte Modell kann prinzipiell übertragen und für neue Beispiele benutzt werden. Allerdings zeigt sich, daß in manchen Fällen ein weiteres Lernen erforderlich ist. Hier offenbaren sich Nachteile im Programm ID3. Es bietet keine Möglichkeit, ausgehend von einem einmal ermittelten Entscheidungsbaum weiterzulernen. Diese Option ist dann wünschenswert, wenn durch eine gelernte Funktion bereits ein Großteil der Objekte richtig klassifiziert wurden. Hier ist jedoch eine Hierarchisierung einfach dadurch zu erreichen, daß ein einmal ermittelter Entscheidungsbaum als komplexes Merkmal in die nächste Iteration eingeht. Dann ist es wahrscheinlich, daß dieses Merkmal, welches erwiesenermaßen für eine große Anzahl von Beispielen gilt, zunächst als Klassifikationsmerkmal benutzt wird und somit die erste Verzweigung im neuen Baum bildet. Ein weiterer Nachteil basiert darauf, daß bei der Erzeugung des Entscheidungsbaums auf jeder Stufe nur das Merkmal benutzt wird, welches die Beispiele am besten klassifiziert. Hierbei kann nicht berücksichtigt werden, daß diese Entscheidung auf der nächsten Ebene möglicherweise sehr große Verzweigungen bedingt. An dieser Stelle sollten eventuell doch Suchverfahren eingesetzt werden, die etwa nach dem Prinzip des Beam-Search eine gewisse Anzahl der jeweils besten Merkmale weiter untersucht und aus der Menge der entstandenen Bäume denjenigen ableitet, der am kürzesten ist.

Wichtig jedoch ist, daß das abgeleitete Wissen “vernetzt” dargestellt ist. Die einzelnen Objekte existieren somit nicht autonom, sondern sind über Relationen miteinander verknüpft. Diese Nutzung dieser Relationen ist insbesondere für die Bildanalyse wichtig. Auch wenn manche der abgeleiteten Beziehungen recht offensichtlich sind, so ist mit diesem Verfahren doch ein Werkzeug geschaffen, diese allgemein abzuleiten. Außerdem liegen die Relationen gleich in einer Darstellung vor, die weiterverarbeitbar ist. Wenn auch für viele Fälle bekannt ist, daß solche Relationen existieren (z.B. die in der Bildanalyse vielzitierte Nutzung der Beziehung “Haus liegt an Straße” [McKeown 1994] für die Erkennung von Häusern), so ist nicht unmittelbar klar, wie diese zu formalisieren und in einem System abzulegen ist. Dies ist für die hier genutzten Beziehungen direkt gegeben.

11 Zusammenfassung und Diskussion

Diese Arbeit beschäftigt sich mit dem Wissenserwerb für intelligente Systeme. Dafür wird der Einsatz von Methoden des Maschinellen Lernens genutzt. Damit wurde eine neue Technik für eine gegebene Problemstellung eingesetzt. Daher ist abschließend zu klären, welches die Vor- und Nachteile des realisierten Systems sind, um so den Gewinn durch das neue Verfahren einzuschätzen.

Der Einsatz von Lernverfahren läßt sich - aufgrund der gegebenen Aufgabe, der Bildinterpretation, sehr gut begründen: zur Interpretation natürlicher Szenen ist eine flexible Modellierung erforderlich - diese können Lernverfahren prinzipiell leisten. Der Ansatz beruht auf einer engen Verbindung von Interpretation und Lernen. Die Struktur der Objekte ist vorgegeben, während die jeweiligen Eigenschaften und Relationen über Lernverfahren ermittelt werden. Auf dieser Basis wurde ein Tool zur Wissensakquisition für die Bildanalyse geschaffen. Lernen macht implizites Wissen - die gegebene Bildrepräsentation - explizit. Indem das System versucht, alles Gegebene zu erklären, wird in einem Wechselspiel zwischen Interpretation und Lernen eine komplette, strukturierte Szenenbeschreibung abgeleitet. Das Programm bildet den menschlichen Wissenserwerb in idealisierter Form nach und ermöglicht somit eine inkrementelle Erweiterung des Wissens über Objekte. Zunächst existieren lediglich namenlose Objekte (Polygone). Indem der Lehrer einige typische von ihnen klassifiziert, wird eine entsprechende Objektklasse erzeugt, sowie Funktionen zu ihrer Klassifikation. Diese Objektklassen (semantische Objekte) stehen dem System unmittelbar zur weiteren Verarbeitung zur Verfügung. Von besonderer Bedeutung ist die inkrementelle Verfeinerung der Objektdefinition, indem die Objektklassen sukzessive spezialisiert werden können. Aufbauend auf der Kenntnis dieser Objekte können jetzt Aggregationen höherer Ordnung durchgeführt werden. Die ermittelte Beschreibung - das Objektmodell - ist hierarchisch strukturiert und vernetzt, und damit für die Bildanalyse nutzbar. Diese Art der Verarbeitung kann mittels objektorientierter Darstellung sehr elegant verwaltet werden. Das Modell besteht aus Objekten mit ihren jeweiligen Eigenschaften und speziellen Methoden, die über komplexe Relationen miteinander in Beziehung stehen. Wesentlich an vorliegendem Ansatz ist, daß neue Relationen objektabhängig gelernt werden können. Das entstandene Modell enthält eine spezifische und eine generische Komponente: die spezifischen, individuellen Objektinstanzen bauen die Gesamtszene auf; die generalisierten Objektklassen stehen für allgemeine Klassenbeschreibungen, die alle Eigenschaften und Relationen ihrer Einzelinstanzen umfassen. Damit konnte das Ziel der Aufgabenstellung, die Ableitung von allgemeinen Modellen, erreicht werden.

Der Interpretationsprozeß wird dabei von den Objekten selbst gesteuert - nachdem er einmal initialisiert wurde. Die Objekte wenden nacheinander ihre Methoden an. Für eventuell im Laufe der Bearbeitung neu hinzukommende Objekte gilt dies gleichermaßen. Die Kontrolle des Lernverfahrens obliegt dem Lehrer, der eingreift, um das Lernen neuer Objekte anzustoßen, bzw. auch um bestehendes Wissen zu revidieren bzw. korrigieren.

Vergleich mit bestehenden Ansätzen

Der Grundgedanke der Arbeit besteht in der Bereitstellung einer "reichhaltigen" Umgebung, welche das System nutzt, um neue Konzepte zu lernen. Diese Idee liegt auch der Arbeit von Stier zugrunde. Er bildet die neuen Konzepte allerdings, indem er für alle möglichen Kombinationen der Merkmale prüft, ob sie die gegebenen Beispiele erklären können. Das dabei verwendete erschöpfende Suchverfahren zur Bestimmung der Konzepte konnte lediglich - aufgrund der Komplexität - anhand eines einfachen Beispiels demonstriert werden. In vorliegendem Fall wurde die Komplexität durch eine Beschränkung in der Struktur der Konzepte reduziert. In dieser Arbeit sind die abgeleiteten Konzept gleichzeitig auch generalisierte Beschreibungen. Damit unterscheidet sie sich von Hättichs Arbeit, welche sich auf die Beschreibung individueller Objekte beschränkt. Kappenberger geht von einer fixen Anzahl an Merkmalen aus, für die jeweils Generalisierungsvorschriften vorliegen. Connel und Brady's Ansatz ist sehr weitreichend, allerdings stellt auch er lediglich ein Prototyp dar. Im Unterschied zu Winstons Arbeit werden hier direkt für die Bildanalyse beobachtbare Merkmale benutzt, während er auf der Basis einer abstrakteren Beschreibungssprache arbeitet - in die die Beispiele zuerst transformiert werden müssen.

Wertung und mögliche Erweiterungen

Das vorgestellte Verfahren soll eine prinzipielle Vorgehensweise aufzeigen, indem es eine Realisierung eines Lernverfahrens darstellt, auf das weitergehende Verfahren aufsetzen können. Daher ist es als Prototyp anzusehen - mit vielen Vorteilen und einigen Defiziten. Es lag allerdings auch nicht in der Intention, ein operationelles

System zu entwickeln. Vielmehr sollte exemplarisch herausgestellt werden, wie und unter welchen Bedingungen Lernverfahren sinnvoll für die Objektmodellierung in der Bildanalyse einsetzbar sein können.

Als grundsätzliche Vorteile des realisierten Prototyps sind hervorzuheben:

Inkrementeller Wissenserwerb: Die Modellierung erfolgt auf der Basis von Beispielen in Interaktion mit Nutzer. Dies führt dazu, daß die Interpretation solange autonom erfolgen kann, wie Modellwissen vorhanden ist. Der jeweilige Stand der Interpretation wird dann für weiterführende Lernaufgaben als Ausgangspunkt genommen.

Direkte Nutzung der Modelle: Die Integration von Lernen und Interpretation führt dazu, daß das Gelernte unmittelbar für die weitere Interpretation genutzt werden kann, bzw. umgekehrt die neue Interpretation für weitere Lernschritte eingesetzt wird.

Beobachtbare Merkmale: Die Modelle setzen sich aus - auch für die Bildverarbeitung - beobachtbaren Merkmalen zusammen (Kanten, Flächen, Relationen (wie Parallelität, Rechtwinkligkeit ...)).

Erweiterbarer Merkmalspool: Welche Merkmale als Basismerkmale zur Verfügung stehen, kann sich aufgabenabhängig ändern. Eine Erweiterung ist aufgrund des Merkmalspools jedoch sehr einfach möglich. Z.B. kann sich während des Programmablaufs ergeben, daß weitere Merkmale sinnvoll sind. Diese können dann eingefügt werden.

Hierarchisch strukturierte Modelle: Die abgeleiteten Modelle stellen die Objekte hierarchisch strukturiert dar. Diese Struktur ist besonders für anschließende Bildanalyseverfahren wichtig.

Erweiterbares Verfahren: Das Interpretations- und Lernverfahren ist prinzipiell erweiterbar auf andere Objekte, z.B. Punkte oder Linien. Die ist insbesondere für die kartographische Objekterkennung wichtig.

Die Entwicklung eines Prototyps zeigte auch einige Schwachstellen des Systems, die Ausgangspunkt für mögliche Erweiterung sein können:

Merkmale: Das System basiert auf einer "reichen" Sprache, d.h. einer Sammlung aussagekräftiger Merkmale zur Objektbeschreibung. Diese Liste ist gegebenenfalls um weitere Merkmale zu ergänzen. Bislang wurden einfache geometrische Merkmale genutzt. Auch wurden lediglich 2D-Merkmale eingesetzt. Viele Objekte der realen Welt - insbesondere Häuser - zeichnen sich jedoch durch ihre 3D-Struktur aus, die dann auch sehr effizient für die Erkennung und Rekonstruktion eingesetzt werden kann [Haala 1994]. Eine Erweiterung des Systems könnte zudem dahingehen, nicht Flächen (Polygone) als "elementare Objekte" zu verwenden, sondern komplexere Strukturen wie z.B. Geons, um eine größere Beschreibungsmächtigkeit zu erreichen.

Inferenzstrategie: Es wurde kein echtes Blackboard genutzt, welches i.A. asynchron abgearbeitet wird, sondern lediglich eine Art von Ringliste, die die einzelnen Objekte nacheinander behandelt. Ebenfalls erfolgt keine Kontrolle und Steuerung der Effizienz des Ablaufs. Dies ist für ein größeres System - besonders im Hinblick auf eine Operationalisierung - allerdings erforderlich [Lange & Schröder 1994]. Wachsen die Anzahl der Beispielobjekte und die Anzahl der Objektklassen, so müssen geeignete Mechanismen eingeführt werden. Hierunter fallen z.B. Regeln, die angeben, welche Objekte zunächst untersucht (poly1 oder poly2 oder ...) werden sollen, bzw. die Zugehörigkeit zu welcher Objektklasse (Feld oder Straße oder Haus oder ...) zuerst betrachtet werden soll.

Lernverfahren: Mit ID3 kam ein einfaches Lernverfahren zum Einsatz. Der grundlegende Nachteil des Programms liegt darin, daß es keine höheren Relationen verarbeiten kann. Erweiterungen von ID3, die für ein operationelles System relevant sind:

- Für eine Nutzung des Systems in realen Bilddaten ist die Einbeziehung und Verarbeitung ungenauer Merkmale erforderlich: auf diese Weise kann berücksichtigt werden, mit welcher Genauigkeit bestimmte Merkmale erfaßbar sind (z.B. Verwechseln von TEE-Knoten mit FRK-Knoten). Hierzu sind die in Kapitel 8.2.3 angesprochenen Erweiterungen des Lernverfahrens zu integrieren.
- In realen Beispielen kann es vorkommen, daß manche Merkmale nicht beobachtbar sind und daher fehlen. Das Lernverfahren muß dennoch eine Klassifizierung ermöglichen.

- Es sind weitere Möglichkeiten der Vorstrukturierung der Merkmale vorzusehen, um allzu buschige Entscheidungsbäume zu vermeiden. Zudem ist der Aspekt der inkrementellen Erweiterung bestehender Entscheidungsbäume von Bedeutung.

Integration von Unsicherheit: Die Merkmalsausprägungen der Objekte werden als sicher angenommen - wenn auch mit einer gewissen Streuung. Eine Erweiterung ist dahingehend denkbar, daß die Zuverlässigkeit bzw. die Bedeutung eines Merkmals ebenfalls mit abgelegt wird. Diese Information kann dann bei der Abarbeitung verwendet werden.

Mächtigeres Interpretationsverfahren: Das realisierte Verfahren nutzt lediglich die diskriminierenden Objektfunktionen zur Interpretation. Ziel ist es allerdings, nicht nur strukturelle, sondern auch kontextuelle Information für die Erkennung einzusetzen.

Nutzung der Modelle für die Bildanalyse

Ein wichtiger Aspekt, der in der Arbeit nur die Randbedingung für die Entwicklung des Systems darstellte, ist der Einsatz dieser Modelle für die Bildanalyse. D.h. die Nutzung der Modelle in realem Bildmaterial. Hierzu sind noch einige Schritte durchzuführen. Insbesondere ist der Aspekt der Unsicherheit der ermittelten Merkmale speziell zu betrachten. Diese Merkmale gilt es, im Bild zu extrahieren. Während die Extraktion von Polygonen in idealen Daten kein Problem darstellt, ist dies für Bilder i.a. nicht trivial. Hierzu sind Gruppierungsverfahren anzusetzen [Price & Huertas 1992], [Mohan & Nevatia 1989]. Neben den Merkmalen müssen auch die Relationen zwischen den Objekten ebenfalls in den Bildern detektiert werden.

In vorliegendem Fall bildeten ideale Daten das Ausgangsmaterial für das Lernen. Prinzipiell läßt sich der Ansatz jedoch auch für das Lernen aus realen Bildern verwenden. Allerdings ist dann festzulegen, welches die elementaren Objekte sind, bzw. wie sie abgegrenzt werden können. Hier ist zudem speziell zu berücksichtigen, daß die Segmentierungsergebnisse fehlerhaft bzw. unvollständig sein können. Die grundlegende Prozessierung ist allerdings identisch.

Auch wenn die Probleme erkannt sind und mögliche Lösungswege beschritten werden, ist es immer noch ein weiter Weg zu intelligenten Systemen, die mit der "Realität" umgehen können - zumal mit D. Lenat gilt: "*Alles wird soviel häßlicher, wenn man sich durch die Daten jenseits sorgfältig ausgewählter Beispiele stören läßt.*". Allerdings ist es die Realität, an der autonome Systeme letztendlich gemessen werden.

Abbildungsverzeichnis

1.1	Zuordnung von Bild- und Modell auf verschiedenen Repräsentationsstufen	11
1.2	Bild mit überlagerten Haus- und Straßengrenzen - Eingabe für das System sind diese überlagerten Daten	13
1.3	Übersicht über die verwendeten Darstellungsformen	14
2.1	Figur-Grund-Trennung: Erkennung ist erst möglich, nachdem eine Gruppierung erfolgt ist (aus [Marr 1982])	15
2.2	Ames Zimmer: Dem Prinzip der Größenkonstanz wird - aufgrund der speziellen Konstruktion des Zimmers - scheinbar widersprochen (aus [Bruce & Green 1991])	16
2.3	Attneave'sche Katze: für die Interpretation wichtige Information ist in den Ecken gespeichert - die Verbindung dieser Eckpunkte ergibt bereits ein anschauliches Bild der Katze.	19
2.4	Beispiel für verschiedene Abstraktionsebenen eines Begriffs	20
3.1	Von der Realität zur Repräsentation	22
3.2	Zwei Möglichkeiten des geometrischen Modellierens: parametrische Beschreibung (links), Randdarstellung (rechts)	24
3.3	Beispiele für Gestaltgesetze	25
3.4	Beispiel für ein Semantisches Netz	31
4.1	Beispiel nach [Reimer 1991] für unterschiedliche Repräsentationen (rechts) der Realität (links)	35
5.1	Voraussetzungen für Lernverfahren	39
5.2	Diametrales Verhältnis zwischen Aufwand des Lehrers und des Schülers	41
5.3	Lernen eines Entscheidungsbaums aus Beispielen mit ID3	44
5.4	Beispiel und Gegenbeispiele (near miss) für das Konzept eines Bogens (Arch)	45
5.5	Beispiel für die Bedeutung von Hintergrundwissen für das Lernen: Voraussetzung für sinnvolle Einteilung der Punkte ist das Konzept eines Kreises	46
5.6	Klassifizierung von Bällen mittels Cobweb (aus [McKusick & Thompson 1990])	47
6.1	Generische Strukturen – Geons (aus [Biederman 1985])	53
6.2	Wahrscheinlichkeiten der Gruppierung von Merkmalen	54
6.3	Beispiel für Segmentierung aufgrund physikalischer Aspekte	55
7.1	Objektmodellierungssystem mit Interpretations- und Lernkomponente	59
9.1	Blackboard, visualisiert anhand von Agenten um einen runden Tisch	70
9.2	Verschiedene Knotentypen	72
9.3	Punkte, Linien und Knoten	72
9.4	Übersicht über verwendete Objektklassen: semantische Objekte sind Unterklassen von Polygonen; alle Objektklassen können weitere Unterklassen, sowie individuelle Objektinstanzen besitzen.	76
9.5	Übersicht über die Relationen der Objekte: Aggregation und Assoziation	76
9.6	Methoden der Objektklasse Generisches Objekt	77
9.7	Methoden der Objektklasse Polygon	77
9.8	Verschiedene Hierarchietypen ([Janssen 1993] nach [Husing 1993])	78
9.9	Übersicht über die Objektklassen und Methoden	78

9.10 Lernen von den Objekt-Konzepten Feld und Verkehr 85

9.11 Fehlerhafte Interpretation dreier Radwege als Straße 86

9.12 Darstellung der Assoziationen `strasse_radweg` (gestrichelt) sowie der Relation `feld_feld` (durchgezogen) (links); Ergebnis: Beschreibung der Szene auf Basis der ermittelten Konzepte (rechts) . . . 88

9.13 Abgeleitete Objekte und Relationen 88

9.14 Modellierung, Programmsynthese und Interpretation 90

10.1 Zu diesem Bild existieren digitalisierte Flurstücke, Straßen und Wegraine 93

10.2 Straßen, Flurstücke und Wegraine wurden korrekt klassifiziert 94

10.3 Aggregierbare Flurstücke (links); Ergebnis der Interpretation (rechts) 95

10.4 Ausschnitt aus einer DGK5 - Gebiet Gengenbach 96

10.5 Vorgegebene Objekte: Straßen, Flurstücke und Häuser (Die einzelnen Objektklassen sind durch unterschiedliche Grauwerte visualisiert) (links); Ablauf der Interpretation (rechts) 97

10.6 Erste Interpretation: alle Häuser (links) und Straßen (rechts) wurden richtig klassifiziert 98

10.7 Erste Interpretation: einige Flurstücke konnten nicht klassifiziert werden 98

10.8 Flurstücke, die Relation **flur_flur** teilen (links); Aggregation von Objekten (rechts) 103

10.9 Ergebnis: vollständige Szeneninterpretation 104

10.10 Interpretation der Testdaten: Häuser, Straßen und Flurstücke wurden korrekt klassifiziert 105

10.11 Flurstücke, die die Relation **flur_flur** teilen (links); Ergebnis der Interpretation (rechts) 106

Literatur

- Aamodt, A. & Plaza, E. [1994], 'Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches', *AI Communications* **7**(1), 39–59.
- Adam, A. [1994], Objekterkennung auf der Basis generischer Modelle nach dem Blackboard Prinzip, Master's thesis, IVPR/IfP, Universität Stuttgart.
- Aloimonos, J. [1990], Purposive and Qualitative Active Vision, *in*: 'Proceedings of the DAPRA Image Understanding Workshop', Pittsburg. PA, Seiten 816–828.
- Ammon, K. [1987], The Automatic Development of Concepts and Methods, Doktorarbeit, Universität Hamburg.
- Anderson, J. [1989], *Kognitive Psychologie*, Spektrum der Wissenschaft.
- Bajcsy, R. [1985], Active Perception vs. Passive Perception, *in*: 'Proc. Third Workshop on Computer Vision: Representation and Control', Bellaire, Michigan, Seiten 55–59.
- Barr, A. & Feigenbaum, E. [1982], *The Handbook of Artificial Intelligence, Vol. 1*, William Kaufmann Inc., Los Altos, California.
- Barrett, R., A. R. & Sloman, A. [1985], *POP-11, A Practical Language for Artificial Intelligence*, Ellis Horwood Ltd., Chichester, West Sussex, England.
- Bergevin, R. & Levine, M. [1993], 'Generic Object Recognition: Building and Matching Coarse Descriptions from Line Drawings', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**(1), 19–36.
- Bhanu, B. & Poggio, T. [1994], 'Introduction to the Special Section on Learning in Computer Vision', *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Biederman, I. [1985], 'Human Image Understanding: Recent Research and a Theory', *Computer Vision, Graphics and Image Processing* **32**, 29–73.
- Bill, R. & Fritsch, D. [1991], *Grundlagen der Geo-Informationssysteme: Hardware, Software und Daten*, Wichmann, Karlsruhe.
- Birnbaum, L., Brand, M. & Cooper, P. [1993], Looking for Trouble: Using Causal Semantics to Direct Focus of Attention, *in*: 'Proceedings of the ICCV 1993', Berlin, Seiten 49–56.
- Bobick, A. & Bolles, R. [1989], Representation Space: An Approach to the Integration of Visual Information, *in*: 'Proceedings of the CVPR 1989', Seiten 492–499.
- Boden, M. [1989], *Artificial Intelligence and Natural Man*, MIT Press, London.
- Bowyer, K. [1993], Computer Vision as Repeated Perceptual Organization, *in*: 'Workshop on Understanding Aerial Images with Emphasis on Urban Scenes', The Ohio State University.
- Brachman, R. & Levesque, H. [1985], *Readings in Knowledge Representation*, Morgan Kaufmann, Los Altos.
- Braun, C., Kolbe, T., Lang, F., Schickler, W., Steinhage, V., Cremers, A., Förstner, W. & Plümer, L. [1994], Modelle für die photogrammetrische Gebäuderekonstruktion, *in*: 'Workshop zum Thema "Visual Computing"', Darmstadt, Seiten 1–17.
- Brewka, G. [1989], 'Nichtmonotone Logiken - Ein kurzer Überblick', *Künstliche Intelligenz* (2), 5–12.
- Brockhaus* [1993], 19. Auflage.
- Brooks, R. [1983], 'Model-Based Three-Dimensional Interpretations of Two-Dimensional Images', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **5**(2), 140–150.
- Bruce, V. & Green, P. [1991], *Visual Perception - Physiology, Psychology and Ecology*, Lawrence Erlbaum Ass., London.
- Burns, J.B., A. H. & Riseman, E. [1986], 'Extracting straight lines', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**, 425–455.

- Campbell, J. [1989], *The improbable Machine*, Simon and Schuster.
- Canon, H. [1980], "Flavors", Technical report, MIT Artificial Intelligence Laboratory, Cambridge (Mass.).
- Cheeseman, P., Kelly, J., Self, M., Stutz, J., Taylor, W. & Freeman, D. [1988], AutoClass: A Bayesian Classification System, *in: '5th Int. Conf. on Machine Learning'*, Ann Arbor, Seiten 54–64.
- Chen, D. & Schunck, B. [1990], Robust Statistical Methods For Building Classification Procedures, *in: 'Int. Workshop on Robust Computer Vision'*, Seattle, Seiten 72–85.
- Clocksink, W. & Mellish, C. [1981], *Programming in Prolog*, Springer-Verlag, Berlin.
- Clowes, M. [1971], 'On seeing things', *Artificial Intelligence* **2**, 79–112.
- Cohen, P. & Feigenbaum, E. [1982], *The Handbook of Artificial Intelligence, Vol. 3*, William Kaufmann Inc., Los Altos, California.
- Connell, J. & Brady, M. [1985], Learning Shape Descriptions, *in: 'Proceedings of the 9th IJCAI'*, Morgan Kaufmann, Los Angeles, Seiten 922–925.
- Dickinson, S., Pentland, A. & Rosenfeld, A. [1992], '3-D Shape Recovery Using Distributed Aspect Matching', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**(2), 174–198.
- Draper, B., Collins, R., Brolio, J., Hanson, A. & Riseman, E. [1989], 'The Schema System', *International Journal of Computer Vision* **2**, 209–250.
- Draper, B., Hanson, A. & Riseman, E. [1991], Learning Blackboard-based Scheduling Algorithms for Computer Vision, Technical report, Dept. of Computer Science, University of Massachusetts, Amherst, MA.
- Eggert, D., Bowyer, K. & Dyer, C. [1992], Aspect Graphs: State-of-the-Art and Applications in Digital Photogrammetry, *in: 'International Archives of Photogrammetry and Remote Sensing'*, Washington, D.C., Seiten 633–645.
- Eggleston, P. [1989], Symbolic Fusion of MMW and IR Imagery, Technical report, Amerinex Artificial Intelligence, Inc.
- Engelmore, R. & Morgan, T. [1988], *Blackboard Systems*, Addison-Wesley, Wokingham, England.
- Etemadi, A., Schmidt, J.-P., Matas, G., Illingworth, J. & Kittler, J. [1991], Low-level Grouping of Straight Line Segments, *in: 'Proceedings of the British Machine Vision Conference'*.
- Fensel, D. [1991], Knowledge Elicitation, Technical Report 215, Institut für angewandte Informatik und formale Beschreibungsverfahren, Karlsruhe.
- Fensel, D. [1993], 'Ein integriertes System zum maschinellen Lernen aus Beispielen', *Künstliche Intelligenz* **2**(3), 17–23.
- Fisher, D. [1987], 'Knowledge Acquisition Via Incremental Conceptual Clustering', *Machine Learning* **2**, 139–172.
- Flynn, P. & Jain, A. [1991], 'CAD-Based Computer Vision: From CAD Models to Relational Graphs', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**(2), 114–132.
- Förstner, W. [1989], Image Analysis Techniques for Digital Photogrammetry, *in: '42nd Photogrammetric Week 89'*, Schriftenreihe des Instituts für Photogrammetrie, Institut für Photogrammetrie, Stuttgart, Seiten 205–221.
- Förstner, W. [1991], *Statistische Verfahren für die automatische Bildanalyse und ihre Bewertung bei der Objekterkennung und -vermessung*, Vol. C370, Deutsche Geodätische Kommission, München.
- Förstner, W. [1994], A Framework for Low Level Feature Extraction, *in: 'Proceedings of the 3rd ECCV'*, Stockholm, Seiten 383–394.
- Forsyth, R. & Rada, R. [1986], *Machine Learning - Applications in Expert Systems and Information Retrieval*, Ellis Horwood Limited, New York.

- Franck, E. [1991], *Künstliche Intelligenz*, J.C.B. Mohr, Tübingen.
- Fritsch, D. [1993], Image Understanding and GIS, *in*: 'Workshop on Understanding Aerial Images with Emphasis on Urban Scenes', The Ohio State University.
- Fritsch, D., Sester, M. & Schenk, T. [1994], Test on Image Understanding, *in*: H. Ebner, C. Heipke & K. Eder, Hrsg., 'Spatial Information from Digital Photogrammetry and Computer Vision', Vol. 30/3, ISPRS, Munich, Germany, Seiten 243–248.
- Fu, K. [1980], Syntactic (Linguistic) Pattern Recognition, *in*: K. Fu, Hrsg., 'Digital Pattern Recognition', Springer-Verlag, Berlin.
- Fu, K. [1982], *Syntactic Pattern Recognition and Applications*, Prentice Hall, New Jersey.
- Fua, P. & Hanson, A. [1987], 'Resegmentation using generic shape: locating general cultural objects', *Pattern Recognition Letters* Seiten 243–252.
- Fuchs, C. & Förstner, W. [1995], Polymorphic Grouping for Image Segmentation, *in*: 'Proceedings of the 5th ICCV '95', Boston.
- Gibson, J. [1950], *The Perception of the Visual World*, Houghton Mifflin Co, Boston.
- Guzman, A. [1968], Decomposition of a Visual Scene into Three-dimensional Bodies, *in*: 'AFIPS Proceedings of the Fall Joint Computer Conference'.
- Haala, B., Hahn, M. & Schmidt, D. [1993], Quality and performance analysis of automatic relative orientation, *in*: D. M. J. E.B. Barret, Hrsg., 'Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision', SPIE Vol. 1944, Orlando, Seiten 140–150.
- Haala, N. [1994], Detection of Buildings by Fusion of Range and Image Data, *in*: H. Ebner, C. Heipke & K. Eder, Hrsg., 'Spatial Information from Digital Photogrammetry and Computer Vision', Vol. 30/3, ISPRS, Munich, Germany, Seiten 341–346.
- Hansen, C. & Henderson, T. [1989], 'CAGD-Based Computer Vision', *IEEE Transaction on Pattern Analysis and Machine Intelligence* **11**(11), 1181–1192.
- Haralick, R. [1985], Computer Vision Theory: The Lack Thereof, *in*: 'Proc. Third Workshop on Computer Vision: Representation and Control', Bellaire, Michigan, Seiten 113–121.
- Haralick, R. & Shapiro, L. [1993], *Computer and Robot Vision*, Vol. II, Addison-Wesley.
- Hättich, W. [1989], Automatische Modellerstellung für wissensbasierte Werkstückerkennungssysteme, Doktorarbeit, Universität Karlsruhe (TH).
- Heitger, F. & von der Heydt, R. [1993], A Computational Model of Neural Contour Processing: Figure-Ground Segregation and Illusory Contours, *in*: 'Proceedings of the ICCV 1993', Berlin, Seiten 32–40.
- Hilgard, E. & Bower, G. [1973], *Theorien des Lernens I und II*, Ernst Klett, Stuttgart.
- Husing, E. [1993], Land Use Zones and Land Use Patterns in the Atlantic Zone of Costa Rica, Doktorarbeit, Wageningen Agricultural University, The Netherlands.
- IEEE Transactions on Pattern Analysis and Machine Intelligence* [1994], Vol. 16.
- Ikeuchi, K. & Kanade, T. [1988], 'Automatic Generation of Object Recognition Programs', *Proceedings of the IEEE* **76**(8), 1016–1035.
- Illert, A. [1990], Automatische Erfassung von Kartenschrift, Symbolen und Grundrißobjekten aus der Deutschen Grundkarte 1:5000, Doktorarbeit, Fachrichtung Vermessungswesen, Universität Hannover.
- Janssen, L. [1993], Methodology for updating terrain object data from remote sensing data, Doktorarbeit, Wageningen.
- Kappenberger, M. [1990], Eine Wissenserwerbskomponente zum Erlernen von Konzepten durch Konstruktive Induktion, Doktorarbeit, Universität Karlsruhe. VDI-Forschungsberichte.

- Kay, A. & Goldberg, A. [1977], 'Personal Dynamic Media', *Computer* **10**, 31–41.
- Kepuska, V. & Mason, S. [1991], An Artificial Neural Network Approach to Signalized Point Recognition in Aerial Photographs, *in*: 'The First Australian Photogrammetric Conference', Sydney.
- Kodratoff, Y. [1988], *Introduction to Machine Learning*, Pitman, London.
- Kodratoff, Y. [1994], 'Guest Editor's Introduction', *AI Communications* **7**(2), 83–85.
- Kodratoff, Y. & Tecuci, G. [1988], 'Learning Based on Conceptual Distance', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **10**(6), 897–909.
- Kropatsch, W., Hrsg. [1994], *Mustererkennung 1994, "Erkennen und Lernen"*, Springer-Verlag.
- Krzystek, P. [1991], Fully Automatic Mensuration of Digital Elevation Models with MATCH-T, *in*: '43rd Photogrammetric Week 91', Schriftenreihe des Instituts für Photogrammetrie, Institut für Photogrammetrie, Stuttgart, Seiten 203–214.
- Lange, H. & Schröder, C. [1994], Analysis and Interpretation of Changes in Aerial Images: Knowledge Representation and Spatial Reasoning, *in*: H. Ebner, C. Heipke & K. Eder, Hrsg., 'Spatial Information from Digital Photogrammetry and Computer Vision', Vol. 30/3, ISPRS, Munich, Germany, Seiten 475–482.
- Langley, P. [1977], Rediscovering Physics with Bacon-3, *in*: 'Proceedings of the 5th IJCAI'.
- Leclerc, Y. [1988], Constructing Simple Stable Descriptions for Image Partitioning, *in*: 'Proc. Image Understanding Workshop', Vol. 1, Cambridge, Mass., Seiten 365–382.
- Lee, C.-M., Pong, T.-C., Esterline, A. & Slagle, J. [1992], KOR: A Knowledge-Based Object Recognition System, *in*: R. Shapiro, Hrsg., 'Computer Vision and Image Processing', Academic Press, Seiten 329–362.
- Lenat, D. [1990], When Will Machines Learn?, *in*: S. Marcus, Hrsg., 'Knowledge Acquisition: Selected Research and a Commentary', Kluwer Academic Publishers, Seiten 255–257. Reprinted from *Machine Learning*, Vol. 4, Nos. 3-4, 1989.
- Lorenz, K. [1977], *Die Rückseite des Spiegels*, dtv, München.
- Lowe, D. [1985], *Perceptual Organization and Visual Recognition*, Kluwer Academic Publishers.
- Marr, D. [1982], *Vision*, W.H. Freeman and Co., San Francisco.
- Mayer, H. [1994], *Automatische wissensbasierte Extraktion von semantischer Information aus gescannten Karten*, Vol. C417, Deutsche Geodätische Kommission, München.
- McKeown, D. J. [1994], Top Ten Lessons Learned in Automated Cartography, *in*: H. Ebner, C. Heipke & K. Eder, Hrsg., 'Spatial Information from Digital Photogrammetry and Computer Vision', Vol. 30/3, ISPRS, Munich, Germany.
- McKusick, K. & Thompson, K. [1990], COBWEB/3: A Portable Implementation, Technical report, NASA Ames Research Center.
- Meier, A. [1986], *Methoden der graphischen und geometrischen Datenverarbeitung*, B.G. Teubner, Stuttgart.
- Messer, T. [1992], *Wissensbasierte Synthese von Bildanalyseprogrammen*, DISKI, infix, St. Augustin.
- Michalski, R. [1984], A Theory and Methodology of Inductive Learning, *in*: R. Michalski, J. Carbonell & T. Mitchell, Hrsg., 'Machine Learning', Springer-Verlag, Berlin, Seiten 83–134.
- Michalski, R., Carbonell, J. & Mitchell, T. [1984], *Machine Learning - An Artificial Intelligence Approach*, Springer-Verlag, Berlin.
- Michalski, R., Carbonell, J. & Mitchell, T. [1986], *Machine Learning II - An Artificial Intelligence Approach*, Vol. 2, Springer-Verlag, Berlin.
- Michalski, R. & Stepp, R. [1983], 'Automated Construction of Classifications: Conceptual Clustering versus Numerical Taxonomy', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **5**(4), 396–409.

- Michalski, R. & Stepp, R. [1984], Learning from Observation: Conceptual Clustering, *in*: R. Michalski, J. Carbonell & T. Mitchell, Hrsg., 'Machine Learning', Springer-Verlag, Berlin, Seiten 331–363.
- Minsky, M. [1975], A Framework for Representing Knowledge, *in*: P. Winston, Hrsg., 'The Psychology of Computer Vision', McGraw-Hill, New York, Seiten 211–277.
- Minsky, M. & Papert, S. [1969], *Perceptrons*, MIT Press, Cambridge, Mass.
- Mohan, R. & Nevatia, R. [1989], 'Using Perceptual Organization to Extract 3-D Structures', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**(11), 1121–1139.
- Molenaar, M. [1989], Knowledge Representation in Vector Structured Data Bases, *in*: M. Schilcher & D. Fritsch, Hrsg., 'Geo-Informationssysteme: Anwendungen - Neue Entwicklungen', Wichmann Verlag, Karlsruhe, Seiten 311–321.
- Newell, A. [1973], Production Systems: Models of Control Structures, *in*: W. Chase, Hrsg., 'Visual Information Processing', Academic Press, New York.
- Niemann, H. [1981], *Pattern Analysis*, Springer Verlag, Berlin.
- Nilsson, N. J. [1982], *Principles of Artificial Intelligence*, Springer Verlag, Berlin.
- Pahlavan, K., Uhlin, T. & Eklundh, J.-O. [1993], Dynamic Fixation, *in*: 'Proceedings of the ICCV 1993', Berlin, Seiten 412–419.
- Pao, Y.-H. [1989], *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley, New York.
- Plietker, B. [1994], Semiautomatic Revision of Street Objects in ATKIS Database DLM 25/1, *in*: 'International Archives of Photogrammetry and Remote Sensing', ISPRS Commission IV, Atlanta, USA, Seiten 311–317.
- Price, K. & Huertas, A. [1992], Using Perceptual Grouping to Detect Objects in Aerial Scenes, *in*: 'International Archives of Photogrammetry and Remote Sensing', Washington, D.C., Seiten 842–855.
- Quillian, M. [1968], Semantic Memory, *in*: M. Minsky, Hrsg., 'Semantic Information Processing', MIT Press, Cambridge, Mass.
- Quinlan, J. [1984], Learning Efficient Classification Procedures and their Application to Chess End Games, *in*: R. Michalski, J. Carbonell & T. Mitchell, Hrsg., 'Machine Learning', Springer-Verlag, Berlin, Seiten 463–482.
- Quinlan, J. [1986], 'Induction of Decision Trees', *Machine Learning* **1**(1), 81–106.
- Quinlan, J. [1987], Decision Trees as Probabilistic Classifiers, *in*: 'Proceedings of the 4th International Workshop on Machine Learning', Irvine, Seiten 31–37.
- Quinlan, J. [1990], 'Learning Logical Definitions from Relations', *Machine Learning* **5**, 239–266.
- Reimer, U. [1991], *Einführung in die Wissensrepräsentation*, Teubner, Stuttgart.
- Resnikoff, H. [1989], *The Illusion of Reality*, Springer Verlag, New York.
- Rich, E. [1988], *KI-Einführung und Anwendungen*, McGraw Hill.
- Ringland, G. & Duce, D. [1988], *Approaches to Knowledge Representation - An Introduction*, Research Studies Press LTD., England.
- Rock, I. [1986], *Wahrnehmung - vom visuellen Reiz zum Sehen und Erkennen*, Bibliothek: Spektrum der Wissenschaft, Heidelberg.
- Rumelhard, D. & McClelland, J. [1986], *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, Cambridge.
- Sagerer, G. [1985], *Darstellung und Nutzung von Expertenwissen für ein Bildanalyseystem*, Springer-Verlag, Berlin.

- Samuel, A. [1967], 'Some Studies in Machine Learning using the Game of Checker II: Recent Progress', *IBM J. Research and Development* 11.
- Schalkoff, R. [1989], *Digital Image Processing and Computer Vision*, Wiley, New York.
- Schalkoff, R. [1992], *Pattern Recognition: Statistical, Structural and Neural Approaches*, John Wiley.
- Schenk, T. [1992], Algorithms and Software Concepts for Digital Photogrammetric Workstations, Technical Report 14, The Ohio State University, Columbus, Ohio, USA.
- Schickler, W. [1992], Feature Matching for Outer Orientation of Single Image Using 3-D Wireframe Control-points, *in: 'International Archives of Photogrammetry and Remote Sensing'*, Washington, D.C., Seiten 591–598.
- Schutte, K. [1994], Knowledge Based Object Recognition of Man-Made Objects, Doktorarbeit, University of Twente, Enschede, Netherlands.
- Segen, J. [1988], Learning Graph Models of Shape, *in: 'Proceedings 5th International Conference on Machine Learning'*, Ann Arbor, Seiten 29–35.
- Sester, M. [1992], Automatic Model Acquisition by Learning, *in: 'International Archives of Photogrammetry and Remote Sensing'*, Washington, D.C., Seiten 856–863.
- Sester, M. & Förstner, W. [1989], Object Location based on Uncertain Models, *in: 'Mustererkennung 1989'*, Vol. 219 of *Informatik Fachberichte*, Springer Verlag, Hamburg, Seiten 457–464.
- Shannon, C. & Weaver, W. [1949], *The Mathematical Theory of Communication*, University of Illinois Press.
- Shapiro, A. [1987], *Structured Induction in Expert Systems*, Addison-Wesley.
- Sharples, M., Hogg, D., Hutchinson, C., Torrance, S. & Young, D. [1989], *Computers and Thought*, MIT Press.
- Shortliffe, E. [1976], *Computer-based Medical Consultations: MYCIN*, North-Holland, New York.
- Simon, H. [1984], Why should Machines learn ?, *in: R. Michalski, J. Carbonell & T. Mitchell, Hrsg., 'Machine Learning'*, Springer-Verlag, Berlin, Seiten 25–38.
- Sleeman, D. [1994], 'Towards a Technology and a Science of Machine Learning', *AI Communications* 7(1), 29–38.
- Sommer, E., Emde, W., Kietz, J.-U., Morik, K. & Wrobel, S. [1993], *Mobal 2.2 User Guide*, GMD, St. Augustin.
- Sommer, G. [1992], 'Wissensverarbeitung in der Bildanalyse, Teil I, II, III', *Bild und Ton*.
- Spreeuwiers, L. [1992], A neural network edge detector, Technical report, University of Twente, Enschede.
- Stark, L. & Bowyer, K. [1991], 'Achieving Generalized Object Recognition through Reasoning about Association of Function to Structure', *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13(10), 1097–1104.
- Stier, S. [1991], Bildverstehen als Lernprozeß aus elementarem Wissen, Doktorarbeit, Universität Hamburg.
- Stilla, U. & Jurkiewicz, K. [1991], Objektklassifikation mit einem Blackboardorientierten Inferenzmechanismus, Bericht 230, Forschungsinstitut für Informationsverarbeitung und Mustererkennung (FIM).
- Strat, T. [1994], Advancing Computer Vision through Advances in Photogrammetry, *in: H. Ebner, C. Heipke & K. Eder, Hrsg., 'Spatial Information from Digital Photogrammetry and Computer Vision'*, Vol. 30/3, ISPRS, Munich, Germany, Seiten 784–792.
- Strat, T. & Fischler, M. [1991], 'Context-Based Vision: Recognizing Objects Using Information from Both 2-D and 3-D Imagery', *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13(10), 1050–1065.
- Tank, D. & Hopfield, J. [1988], 'Kollektives Rechnen mit neuronenhähnlichen Schaltkreisen', *Spektrum der Wissenschaft*.

- Thompson, K. & Langley, P. [1991], Concept formation in structured domains, *in*: 'Concept Formation: Knowledge and Experience in Unsupervised Learning', Morgan Kaufmann, Inc.
- Thorpe, C., Amidi, O., Gowdy, J., Hebert, M. & Pomerleau, D. [1992], Integrating Position Measurement and Image Understanding for Autonomous Vehicle Navigation, *in*: 'High Precision Navigation 91', Dümmler, Seiten 639–660.
- Towell, G. G. & Shavlik, J. W. [1992], 'Extracting Refined Rules from Knowledge-Based Neural Networks', *Machine Learning*.
- Tsotsos, J. [1987], A 'Complexity Level' Analysis of Vision, *in*: 'Proceedings of the ICCV', London, Seiten 346–355.
- Uhr, L. [1973], *Pattern Recognition, Learning, and Thought*, Prentice Hall, New Jersey.
- Vogel, F. [1975], *Probleme und Verfahren der numerischen Klassifikation*, Vandenhoeck & Ruprecht.
- Vosselman, G. [1992], *Relational Matching*, Vol. 628 of *Lecture Notes in Computer Science*, Springer Verlag.
- Walker, P. & Moore, D. [1988], 'SIMPLE: an inductive modelling and mapping tool for spatially-oriented data', *International Journal of Geographical Information Systems* **2**(4), 347–363.
- Waltz, D. [1975], Understanding Drawings of Scenes with Shadows, *in*: P. Winston, Hrsg., 'The Psychology of Computer Vision', McGraw-Hill, New York, Seiten 19–91.
- Weber, R. & Zimmermann, H.-J. [1991], 'Automatische Akquisition von unscharfem Expertenwissen', *Künstliche Intelligenz* **2**(2), 20–26.
- Weng, J., Ahuja, N. & Huang, T. [1993], Learning Recognition and Segmentation of 3-D Objects from 2-D Images, *in*: 'Proceedings of the ICCV 1993', Berlin, Seiten 121–128.
- Wertheimer, M. [1958], Principles of Perceptual Organization, *in*: D. Beardslee & M. Wertheimer, Hrsg., 'Readings in Perception', Princeton, N.J., Seiten 115–135.
- Wilding, J. [1982], *Perception - From sense to object*, Hutchinson University Library.
- Winograd, T. [1985], Frame Representation and the Declarative / Procedural Controversy, *in*: R. Brachman & H. Levesque, Hrsg., 'Readings in Knowledge Representation', Morgan Kaufmann, Seiten 357–370.
- Winston, P. [1975], Learning Structural Descriptions from Examples, *in*: P. Winston, Hrsg., 'The Psychology of Computer Vision', McGraw-Hill, New York, Seiten 157–209.
- Winston, P. [1977], *Artificial Intelligence*, Addison-Wesley, Reading, MA.
- Wong, A. & You, M. [1985], 'Entropy and Distance of Random Graphs with Application to Structural Pattern Recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **7**(5), 599–609.
- Zadeh, L. [1974], 'Fuzzy Logic and its Applications to Approximate Reasoning', *Information Processing 1974* Seiten 591–594.

Dank

Die vorliegende Arbeit entstand während meiner Tätigkeit als Wissenschaftliche Mitarbeiterin im Sonderforschungsbereich 228 "Hochgenaue Navigation" am Institut für Photogrammetrie der Universität Stuttgart. Für die Ermöglichung der Arbeit bedanke ich mich herzlich sowohl bei dem früheren Leiter des Instituts Herrn Prof. Ackermann als auch dem jetzigen, Herrn Prof. Fritsch. Der Deutschen Forschungsgemeinschaft sei für die finanzielle Unterstützung meiner Mitarbeit im SFB gedankt.

Danken möchte ich hiermit allen, die - direkt oder indirekt - an dieser Arbeit beteiligt waren und somit zu ihrem Gelingen beitrugen. Dazu zählt, wer direkt konstruktiv geholfen hat, aber auch, wer durch beharrliche Diskussionen zur Konkretisierung und Beschleunigung beigetragen hat. Besonders hervorheben möchte ich Herrn Prof. Wolfgang Förstner, der mich zu dem faszinierenden Thema des Maschinellen Lernens hinführte, und mich auf dem Weg durch Höhen und Tiefen begleitet hat. Prof. Fritschs Engagement hat der Arbeit, besonders in der Schlußphase, sehr gutgetan.

Meinen Kolleginnen und Kollegen danke ich für das angenehme und inspirierende Arbeitsklima am Institut.

Zum Schluß sei noch allen gedankt, die mich in dieser Zeit begleitet haben - besonders meiner Familie und meinen Freunden - für den Rückhalt, den ich immer bei ihnen gefunden habe.

Lebenslauf

von
Monika Sester

19. September 1961 als Tochter von Agnes und Mathias Sester geboren in Gengenbach
- 1968 - 1981 Schulbesuch in Reichenbach und in Gengenbach
21. Mai 1981 Abitur
- 1981 - 1987 Studium des Vermessungswesens an der Universität Karlsruhe
- 1987 - 1988 Wissenschaftliche Mitarbeiterin am Institut für Photogrammetrie und Fernerkundung der Universität Karlsruhe
- 1988 - 1994 Wissenschaftliche Mitarbeiterin am Institut für Photogrammetrie der Universität Stuttgart im Sonderforschungsbereich 228 "Hochgenaue Navigation"
- seit 1994 Wissenschaftliche Angestellte und Leiterin der GIS-Gruppe am Institut für Photogrammetrie der Universität Stuttgart