Zur Validierung von Expertenwissen in medizinischen Bildanalysesystemen (Qualitätssicherung und -kontrolle)
Teil 1
Theoretische Grundlagen

Karin Kotzke

Hildesheim, 8. Juni 1993

Zusammenfassung

Der klinische Einsatz wissensbasierter Systeme setzt voraus, daß die Qualität der produzierten Ergebnisse einem definiertem Maß genügt. Aussagen über die Qualität von Software werden durch Verfahren der Qualitätssicherung gewonnen. Durch die Merkmale, in denen sich wissensbasierte Systeme von anderer Software unterscheiden, ergeben sich besondere Probleme für die Qualitätssicherung. Zu diesen Merkmalen zählen die Modellierung von Expertenwissen und dessen Abarbeitung. Entscheidungen während und am Ende der Analyse sind oftmals nicht eindeutig. Soll als Aufgabe im Rahmen der Qualitätssicherung die Entscheidung über die Richtigkeit getroffen werden, kann eine Beurteilung eines einzelnen Ergebnisses, das auch Unsicherheit ausdrückt, mit den herkömmlichen Methoden nur eingeschränkt als korrekt oder falsch klassifiziert werden.

Im Rahmen dieser Arbeit wird in Teil 1 ein Ansatz zur Überprüfung und Beschreibung der Richtigkeit der Ergebnisse der wissensbasierten Bildanalyse unter Berücksichtigung der Auswertung medizinischer Bilddaten entwickelt. Die Überprüfung erfolgt gegen spezifizierte Merkmale des Ein- und Ausgabebereichs des wissensbasierten Systems auf der Grundlage angeordneter Testfälle. In Abhängigkeit von der Ordnung ergibt sich ein spezifizierbares Muster der Analyseergebnisse. Abweichungen deuten auf Fehler in der Wissensbasis. Für die Anwendung des Ansatzes auf das medizinische Bildanalysesystem der Herzmotilitätsinterpretation wird in Teil 2 ein spezielles Software-Phantom entworfen. Die Ergebnisse zeigen, daß die wissensbasierten Analysen im Kontext der Ordnung bezüglich ihrer Richtigkeit beurteilbar sine. Darauf aufbauende Meßverfahren liefern eine quanitative Bewertung des Analysesystems.

Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftliche Mitarbeiterin am Institut für medizinische Informatik der Universität Hildesheim. Für die Anregungen und die weitreichende Förderung und Unterstützung der Arbeit danke ich Herrn Prof. Pretschner. Die Studenten Stefan Günther und Holger Läbe übernahmen im Rahmen von Studienarbeiten einen Teil der Realisierung der Software zum wissensbasierten Bildanalysesystem und zur Testumgebung. Meine Kollegin Catherina Müller war mir durch ständige Gesprächsbereitschaft eine wertvolle Hilfe. Ferner danke ich allen Kollegen und Studenten, die mit Anregungen und Diskussionsbeiträgen am Entstehen der Arbeit beteiligt waren.

Inhaltsverzeichnis

1 Einleitung **13**
1.1 Zur Qualitätssicherung und -kontrolle von Software 13
1.2 Zielsetzung 18

2 Architektur eines wissensbasierten Bildanalysesystems **21**
2.1 Bildanalytische Vorgehensweise 21
2.2 Die Low-Level Phase in der Bildanalyse 23
2.3 Die High-Level-Phase in der Bildverarbeitung 23
2.4 Konzept eines wissensbasierten Bildanalysesystems 24

3 Validierungsverfahren **27**
3.1 Begriffsbestimmungen 29
3.2 Validierung von konventioneller Software 31
3.3 Validierung von wissensbasierten Systemen 37
3.4 Validierung von wissensbasierten Systemen in der Medizin 40
3.4.1 Initiale Projekte 41
3.4.2 Validierung und Verifikation: Modelle und Werkzeuge 42
3.5 Zusammenfassung 45

4 Ein Verfahren für die klassifizierende Analyse **47**
4.1 Grundlegende Definitionen 51
4.2 Formale Grundlagen 52
4.3 Allgemeine Aspekte der Validierung 55
4.3.1 Auswahl der prinzipiellen Teststrategie 56
4.3.2 Validierungsrelevante Eigenschaften der Eingabedaten 61
4.3.3 Validierungsrelevante Eigenschaften der Analyseergebnisse 64
4.3.4 Möglichkeiten und Einschränkungen durch originale und künstliche Testdaten für eine adäquate Teststrategie 69
4.4 Systematik der Testfallgenerierung 75
4.5 Fehlerdetektion 78
4.6 Fehlerlokalisation 84

4.7 Qualitätsbeschreibung 86
4.7.1 Spezifikationsbezogene Qualitätsbeschreibung 88
4.7.2 Zuverlässigkeitsmessung im medizinischen Anwendungsbereich . . . 90
4.8 Zusammenfassung 97

Abbildungsverzeichnis

1.1 Aktionen der Qualitätssicherung 15

2.1 Vorgehensweise beim Bildverstehen 22
2.2 Komponenten eines Bildanalysesystems 24

3.1 Zeitliche Zuordnung von Verifikation, Validierung, Testen 30
3.2 Techniken zur Validierung konventioneller Software 32

4.1 Anordnung von Diagnosen in Bezug auf ein Merkmal 48
4.2 Verteilung von Testfällen im 2-dimensionalen Merkmalsraum 49
4.3 Zusammenhänge zwischen Merkmalsraum und Ein- und Ausgabebereich einer Wissensbasis 50
4.4 Ablauf einer Testphase 60
4.5 Beispiel des erwarteten Gewichtsverlaufs eines A-Objektes 79
4.6 Abgeleitete Spezialtypen des Gewichtungsverlaufs 81
4.7 Verdeutlichung eines Verlaufstyps 82
4.8 Gewichtungsverläufe mit Fehlerhinweis und Gewicht 85
4.9 Uberblick über Objekte und Ausprägungen des Ein- und Ausgabebereichs . 87
4.10 Verteilung der Analyseergebnisse einer Wissensbasis mit vorklassifizierten Testfällen 92
4.11 4-Felder-Tafel für die Einordnung der Analyseergebnisse 94
4.12 Qualitätsmessung nach dem diagnostischen Performanzmodell 95
4.13 (ROC)-Kurve für zwei hypothetische Wissensbasen 96

Tabellenverzeichnis

4.1 Ergebnisse der Analyse einer Fahrzeugbewegung 68

Kapitel 1

Einleitung

Wenn von der Qualität eines Produktes gesprochen wird, ist die Vorstellung dessen, was Qualität bedeutet, sehr konkret, wenn es sich bei dem Produkt um ein materielles Erzeugnis handelt: Ein Kraftfahrzeughersteller produziert Autos, die eine hohe Qualität besitzen, wenn sie reparaturfreundlich sind, den Insassen Schutz bei Unfällen bieten oder einen geringen Schadstoffausstoß besitzen, da der Aspekt des Umweltschutzes immer mehr an Bedeutung gewinnt. Die Komponenten, die die Qualität beschreiben, die Qualitätsmerkmale, werden nicht nur von den Produzenten bestimmt, sondern richten sich auch nach den Anforderungen der Anwender des Produktes. Sie bilden eine Basis für objektive Vergleiche von alternativen Produkten in einem Anwendungsgebiet.

1.1 Zur Qualitätssicherung und -kontrolle von Software

Allgemeiner sind die wesentlichen Qualitätsmerkmale für technische Produkte durch dauerhafte Funktion, Betriebssicherheit und Benutzerfreundlichkeit erfaßt. Im Gegensatz zu den materiellen Erzeugnissen stellen Software-Produkte immaterielle Erzeugnisse dar. Die Qualität ist nicht unmittelbar erfaßbar wie bei technischen Produkten, jedoch müssen auch hier ähnliche Maßstäbe angelegt werden. Im Zusammenhang mit Software ergibt sich der Begriff der Qualität zunächst aus verschiedenen Anforderungen, von denen die folgende Liste nur eine Auswahl nach Schmitz et al. [SCH83] darstellt: Effizienz, Funktionsabdeckung, Portabilität, Richtigkeit, Sicherheit, Verständlichkeit oder Vollständigkeit.

Der Begriff Qualität und daraus folgende Begriffe zur Erreichung und Überprüfung der Qualität können sehr unterschiedlich aufgefaßt werden. Verschiedene Wirtschafts- und Industriebereiche haben sich spezielle Terminologien zur Qualität angeeignet, und selbst Experten in qualitätsspezifischen Bereichen benutzen unterschiedliche Benennungen.

Um einen einheitlichen Begriff zu erhalten, hat sich die ISO (International Organization for Standardization) zur Aufgabe gestellt, allgemein Standards für Qualitätsmanagement und Qualitätssicherung zu setzen. Technische Komitees erarbeiten die internationalen Stan-

dards. Die Mitglieder von ISO und weitere internationale Komitees können dazu beitragen. Das Ergebnis wird den nationalen und europäischen Normenorganisationen zur Übernahme empfohlen.

Für den Bereich des Qualitätsmanagements und der Qualitätssicherung bietet die ISO die internationalen Standards ISO 8402, 9000 bis 9004 [ISO87a] bis [ISO87f]. ISO 8402 beinhaltet Begriffsdefinitionen. Dieser Standard fließt in die Definitionen der 9000er Serie ein. Sie enthalten einen Leitfaden für die Auswahl und die Anwendung von Qualitätsmanagement- und Qualitätssicherungsnormen, Modelle zur Darlegung der Qualitätssicherung in Design/Entwicklung, Produktion, Montage und Kundendienst, sowie einen Leitfaden für das Qualitätsmanagement und die Elemente eines Qualitätssicherungssystems. Die internationalen Normen fließen unverändert in die europäischen [CEN87] und die deutschen Normen [DIN92] ein. Für die europäische Norm ist das *Komitee für europäische Normung* (CEN) und für die deutsche Norm das *Deutsche Institut für Normung e.V.* (DIN) verantwortlich.

Wie die Standards des Qualitätsmanagements und der Qualitätssicherung auf die Softwareentwicklung anzuwenden sind, wird in ISO 9000-3 [ISO91] und der entsprechenden deutschen Norm [DIN92] festgelegt. Sie ist ein Leitfaden für die Anwendung von ISO 9001 auf die Entwicklung, Lieferung und Wartung von Software. Ergänzend dazu sind die nationalen Normen zu sehen, die nicht international standardisiert sind. Die deutsche Norm DIN 55350 beinhaltet beispielsweise in Teil 13 [DIN87] die Definition von Begriffen zur Genauigkeit von Ermittlungsverfahren und Ermittlungsergebnissen, deren Bedeutung teilweise auf Software anwendbar ist. Eine explizite Anwendung auf Software liegt nicht vor. Dagegen bietet DIN 66285 [DIN90] direkt eine Norm für die Gütebedingungen und Prüfbestimmungen von Anwendungssoftware (Abbildung 1.1).

Die zur Erreichung und Überprüfung von Qualität erforderlichen Strukturen und Abläufe werden im Qualitätsmanagementsystem zusammengefaßt. Qualitätssicherung ist innerhalb des Qualitätsmanagementsystems verwirklicht und wird durch die Techniken und Tätigkeiten der Qualitätslenkung realisiert. Der Begriff Qualitätslenkung ist die deutsche Übersetzung für den englischen Ausdruck *quality control*. Der deutsche Ausdruck Qualitätskontrolle wird allgemein synonym dazu verwendet. Nach ISO 8402 gelten folgende Begriffe für Qualität, Qualitätssicherung, Qualitätsmanagementsystem und Qualitätslenkung.

> **Qualität** ist die Gesamtheit von Merkmalen einer Einheit bezüglich ihrer Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen.
>
> **Qualitätssicherung** ist die Menge der geplanten und systematischen Tätigkeiten, die innerhalb des Qualitätsmanagementsystems verwirklicht sind, und die wie erforderlich dargelegt werden, um angemessenes Vertrauen zu schaffen, daß eine Einheit die Qualitätsforderung erfüllen wird.
>
> **Qualitätsmanagementsystem** umfaßt die Organisationsstruktur, Verantwortlichkeiten, Verfahren, Prozesse und erforderlichen Mittel für die Verwirklichung des Qualitätsmanagements.

Qualitätslenkung besteht aus den Arbeitstechniken und Tätigkeiten, die zur Erfüllung der Qualitätsforderungen angewendet werden.

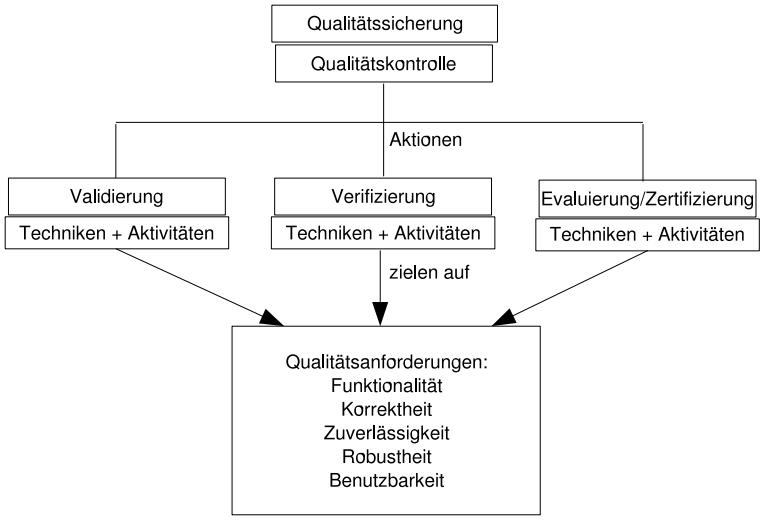

Abbildung 1.1: Aktionen der Qualitätssicherung zur Erfüllung der Qualtitätsanforderungen (Gütebedingungen) nach DIN 66285 [DIN90]. Hinter den Aktionen der Qualitätssicherung stehen die Techniken und Aktivitäten der Qualitätskontrolle, die für die Überprüfung der Qualtitätsanforderungen eingesetzt wird.

Die Aktionen der Qualitätssicherung werden mit den Begriffen Validierung, Verifikation und Evaluierung oder Zertifikation (Abbildung 1.1) erfaßt. Die Verifikation stellt während der Entwicklung des Produkts am Ende einzelner Entwicklungsphasen die Erfüllung der in der vorhergehenden Phase spezifizierten Anforderungen fest. Validierung, Evaluierung oder Zertifikation sind zeitlich auf das Ende des Entwicklungsprozesses projizierte Produktbewertungen. Die Aktionen werden eingehender in Kapitel 3 erläutert.

Die in dieser Arbeit behandelte Qualitätskontrolle von Expertenwissen bezieht sich auf die Qualitätsanforderung Korrektheit. Sie bezeichnet die Übereinstimmung des Software-Produkts mit den Vorgaben zu seiner Erstellung und die Fehlerfreiheit. Nach DIN 66285 [DIN90] gilt für Korrektheit

> *Die Programme und Daten müssen allen Angaben in der Produktbeschreibung und in der Dokumentation entsprechen. Die Funktionen müssen fachlich richtig ausgeführt sein. Insbesondere müssen alle Forderungen aus Regelungen, denen das Produkt laut Produktbeschreibung genügt, erfüllt sein.*

Zur Unterstützung der Qualitätsanforderung Korrektheit stellt sich die Entwicklung adäquater Verfahren als ein Problembereich dar. Für die Qualitätskontrolle existieren zur

Zeit keine praktikablen Methoden, um in großen Programmen Fehlerfreiheit zu beweisen. Mit den praktizierten Methoden verbleibt immer eine Restfehlerwahrscheinlichkeit.

Als Technik für die Überprüfung der Fehlerfreiheit ist der Beweis der Korrektheit von Software theoretisch möglich, sofern eine formale Spezifikation als Grundlage für den Beweis definiert werden kann. Praktisch scheitert der Ansatz bei komplexerer Software an der Erstellung der formalen Spezifikation. Eine wichtige Technik für die Praxis der Qualitätssicherung ist Testen. Beim Testen wird für einige möglichst gut ausgewählte Testdaten das Programm ausgeführt und beobachtet, ob die gewünschte Wirkung erzielt wird. Programme können meist nicht für die üblicherweise überaus große Anzahl aller möglichen Kombinationen von Eingabedaten getestet werden. Testen liefert demnach niemals Gewißheit über die Korrektheit des Programms für die noch nicht getesteten Eingabedaten.

Ein erreichbares Ziel ist die Fehlerreduktion bis zur Unterschreitung einer vorgegebenen Schwelle für die vermutlich noch existierenden und nicht erkannten Fehler. Sofern die Bedeutung einzelner Fehler unterschiedlich gewertet wird, sichert eine geringe Restfehlerwahrscheinlichkeit nicht unmittelbar eine hohe Qualität der Software. Verfälscht ein unentdeckter Fehler das Resultat derart, daß es mit schwerwiegenden Folgen für den Anwender verbunden ist, so ist die Qualität der Software geringer als im Fall eines Fehlers mit kaum bemerkenswerten Auswirkungen. Angenommen, die Überwachung eines Patienten auf der Intensivstation sei computerunterstützt, indem automatisch ein Warnsignal ertönt, wenn eine lebensbedrohende Situation für den Patienten entsteht: Wird durch einen Progammfehler eine bedrohliche Situation nicht als solche klassifiziert und kein Warnsignal erzeugt, kann es für den Patienten den Tod bedeuten. Umgekehrt ist die Klassifikation einer nicht lebensbedrohenden Lage als gefährliche Situation für den Patienten weniger schwerwiegend.

In den Phasen der Qualitätssicherung von Korrektheit gilt es, möglichst viele Fehler zu detektieren. Je komplexer die Programme werden, umso schwieriger wird eine umfassende Qualitätssicherung. Zusätzlich ist in der Regel ein Zeit- und Kostenrahmen vorgegeben, in dem die Aktionen abzulaufen haben.

Neben dem Wunsch der Entwickler, in den Phasen der Softwareentwicklung Hinweise über die Qualität zu erhalten, sind auch außerhalb der Entwicklung stehende Anwender an Informationen über die Qualität interessiert. Sie interessiert wahrscheinlich weniger die Restwahrscheinlichkeit für einen Fehler, sondern vielmehr, wie gut die Software bestimmte Probleme bearbeitet, oder in welchem Fall die Ergebnisse kritisch zu werten sind. Die Aufgabe der Qualitätsbeschreibung ist es, der Gruppe der Entwickler und der Gruppe der Anwender die gewünschten Informationen zu liefern. Zu den Methoden der Qualitätsbeschreibung zählen qualitative Bewertungen und Messungen. Qualitative Bewertungen sind subjektiv und durch Dritte nur eingeschränkt nachzuvollziehen. Messungen sind mit der Offenlegung der Meßmethode objektiv. Eine Meßmethode ist jedoch nur sinnvoll, wenn sie verschiedene Gütekriterien für Software-Qualitätsmaße erfüllt (siehe Kapitel 4).

Die Qualitätsanforderungen, wie sie allgemein für Software definiert sind, gelten ebenso für wissensbasierte Systeme. Diese sind zunächst einmal auch Software, wenn man den nach DIN ISO 9000-3 [DIN92] normierten Begriff zugrunde legt:

> *Software ist ein geistiges Produkt, das aus Programmen, Verfahren und allen dazugehörigen Beschreibungen besteht, die zur Arbeit mit einem Datenverarbeitungssystem gehören.*

Es wird für die folgenden Ausführungen der Begriff konventionelle Software zur Unterscheidung von wissensbasierten Systemen eingeführt. Wissensbasierte Systeme besitzen in der Regel eine besondere Komplexität in ihrer Struktur, ihrer Abarbeitung und ihren Inhalten. In diesen drei Merkmalen unterscheiden sie sich von konventioneller Software dadurch, daß sie auf die Modellierung eines Experten zielen durch die Modellierung des Wissens und des intelligenten Verhaltens. Hierbei werden die Struktur durch entsprechende Repräsentationsformalismen, die Abarbeitung durch die Inferenzstrategie und die Inhalte durch die Einbeziehung speziellen vagen, heuristischen Wissens geprägt.

Die Aktionen der Qualitätssicherung, die sich für konventionelle Software bewährt haben, sind nicht einfach auf wissensbasierte Systeme anzuwenden, da sie die speziellen Probleme der Modellierung intelligenten Verhaltens nicht erfassen. Eine alternative Maßnahme ist die Qualitätskontrolle durch den Anwender, der über Erklärungskomponenten die Überprüfung der *Korrektheit* selbst vornimmt. Eine Qualitätsbeschreibung durch ein Qualitätsmaß, das auch einen Vergleich mit anderen Verfahren zuläßt, kann so nicht ermittelt werden.

Die aktuellen Methoden der Qualitätssicherung von Wissen in wissensbasierten Systemen, die auf formalen Grundlagen basieren, gehen über eine Überprüfung auf der Basis der denotationellen Semantik der verwendeten Repräsentationsform nicht hinaus. Die hier als Expertise bezeichnete Heuristik und Vagheit von Wissen bleibt davon unberührt. Ansätze zur Bewertung der Expertise verwenden im idealen Fall einen *Gold Standard*. In der Regel ist dieser nicht verfügbar, und es werden Situationen konstruiert, die den idealen Fall so weit wie möglich approximieren, z.B. durch einen Test des Systems gegen einen oder mehrere Experten. Die Experten selbst werden in verschiedene Situationen versetzt, z.B. sollen sie direkt das Ergebnis beurteilen, eventuell selbst die Wissensbasis verändern oder eine Beurteilung im Blindversuch vornehmen.

Beim Test von wissensbasierten Systemen wird die qualitative Bewertung der Ergebnisse häufig zur Leistungsbeschreibung des Systems und zur Fehlerdetektion verwendet. Die Bewertung erfolgt in der Regel anhand einer Anzahl von Testfällen. Die Bedeutung des qualitativen Ansatzes erwächst aus der Tatsache, daß zur Zeit quantitative Verfahren zur Qualitätsmessung oder formale Ansätze für die Prüfung der Korrektheit nur eingeschränkt zur Verfügung stehen. Selbst wenn ausreichend viele geeignete Testfälle verfügbar wären, und es sich bei den bewertenden Personen um ausgezeichnete Experten handelt, ist die ausschließlich durch Personen durchgeführte Bewertung kritisch zu sehen. Wie sind beispielsweise bei der Verwendung von unsicherem Wissen Aussagen des Systems zu beurteilen, die durch eine Gewichtung mehr oder weniger bestätigt werden? Die qualitativen Ansätze sind hier unzureichend. Als Folge davon bleiben auch auf der Beurteilung aufsetzende *Qualitätsmessungen* fragwürdig.

1.2 Zielsetzung

Am Anfang dieser Arbeit stand ein wissensbasiertes Bildanalysesystem für die Bewegungsinterpretation des linken Ventrikels in Herzsequenzstudien, das, in der Absicht es später in der Klinik einzusetzen, hohen Qualitätsansprüchen genügen muß. Das System vereint aus der Sicht der Qualitätssicherung und -kontrolle der Korrektheit viele bis dahin ungelöste Probleme. Zu den Problemen zählt die Behandlung der hohen natürlichen Variabilität der medizinischen Bilddaten, die die geeignete Testfallauswahl erschwert. Zusätzlich wird die Akquisition geeigneter Testfälle durch Randbedingungen beschränkt, die kaum zu beeinÂ?flussen sind. Patientenhäufigkeit oder Krankheitshäufigkeit sind nur zwei Beispiele dafür. Weiterhin ist die Qualität unsicheren und heuristischen Wissens, das z.B. für die symbolische Beschreibung von numerischen Daten einfließt, zu erfassen. Zur Prüfung der Korrektheit einer Analyse ist ein Gold Standard nicht verfügbar. Statt dessen muß er durch ein geeignetes Verfahren ersetzt werden.

Die als Validierung bezeichnete Bewertung der Korrektheit der Analysen zielt auf die Wissensbasis am Ende ihres Entwicklungsprozesses. Sie ist die Trägerin des problemspezifischen Wissens und der Expertise in Form eines semantischen Netzes. Die Syntax und Semantik des Repräsentationsformalismus *Semantisches Netz* nach Sagerer [SAG85] wird bereits in der Entwicklungsumgebung geprüft. Die Wissensbasis setzt sich aus problemspezifischen prozeduralen und deklarativen Komponenten zusammen, die der von Sagerer definierten Syntax und Semantik genügen. Wie gut eine Analyse durch das gewählte deklarative Modell, die Methoden und Parameter der prozeduralen Komponente ist, kann davon nicht abgeleitet werden.

Das Ziel der Arbeit ist die Entwicklung und der Einsatz von Methoden der Qualitätssicherung und -kontrolle der Qualitätsforderung Korrektheit für Expertenwissen in medizinischen Bildanalysesystemen durch Validierung der Wissensbasis. Zu diesem Zeitpunkt sind Fehler und unzureichend modellierte und implementierte Expertise zu erwarten. Für die Qualitätssicherung boten sich zunächst Aktionen und Techniken aus dem Bereich konventioneller Softwareentwicklung an, sowie qualitative Bewertungsverfahren von Expertensytemen (Kapitel 3).

Die Techniken sind aus vielerlei Hinsicht für die Validierung einer Wissensbasis unzureichend. Ein Bewertungsverfahren, das das System gegen die Aussagen eines Experten testet, ist unzureichend, um einen Überblick über die Qualität der Analyse-Ergebnisse zu erhalten. Es kann lediglich für ausgesuchte Fälle eine Aussage getroffen werden. Der Informationswert dieser Aussage ist besonders kritisch, wenn die Wissensbasis mit Sicherheitsfaktoren belegte Aussagen generiert. Die Korrektheit der numerischen Darstellung einer Aussage kann durch eine qualitative Beurteilung besonders in den *unsicheren* Fällen nicht ausreichend überprüft werden. Eine hohe Abweichung vom korrekten Wert kann von einem menschlichen Experten wahrscheinlich eindeutig als falsch beurteilt werden. Eine betragsmäßig kleine Abweichung würde eventuell fälschlicherweise als richtig eingestuft. Qualitätsmessungen auf der Basis qualitativer Bewertungen sind irrelevant. Neben dem

Problem, ein Analyseergebnis geeignet zu bewerten, hinterlassen Beschränkungen, die die Anzahl der originalen Testdaten und die Bildinhalte beeinflussen, oft größere *weiße Flecken* auf dem Spektrum möglicher Analysen des Systems, über die keine Hinweise zur Qualität zu ermitteln sind. Beispielsweise beeinflußt die Patientenhäufigkeit die Anzahl der Testfälle und die Krankheitshäufigkeit die Bildinhalte in einer Testdatenmenge.

In dieser Arbeit wird eine Grundlage für methodisches Testen von Wissensbasen im Bereich wissensbasierter medizinischer Bildanalyse entwickelt und auf die Wissensbasis VentriKS zur Bewegungsinterpretation des linken Ventrikels angewendet. Die Testmethode soll den Anteil konventioneller Software und die Expertise überprüfen und Hinweise zur Fehlererkennung und zur Verbesserung der Expertise liefern. Die für eine Validierung interessanten Bereiche der Wissensbasis sollen gezielt untersucht werden können durch die Generierung geeigneter Testfälle. Die ermittelten Ergebnisse fließen in eine Methode zur quantitativen Qualitätsbeschreibung ein, die als Qualitätsmessung bezeichnet wird.

Zur Erreichung des Ziels, eine Validierungsmethode zu entwickeln und einzusetzen, ergeben sich drei Aufgabenschwerpunkte:

Die Methode zur Testfallgenerierung (Software-Phantom): Die Vielfältigkeit von Bilddaten und daraus folgend der Segmentierungsergebnisse, verbunden mit der limitierten Verfügbarkeit von Patientenstudien, verhindert einen Test mit originalen Bilddaten, die den Anforderungen, die sich aus der Validierung ergeben, genügen müssen. Durch die Methode der Testfallgenerierung werden die erforderlichen Testfälle verfügbar.

Die Strategie der Testfallauswertung: Die besonderen Merkmale wissensbasierter Systeme wie z.B. die Repräsentation des Wissens, die Verarbeitung unsicheren Wissens, die Verknüpfung von Klassifikationen mit Sicherheitsfaktoren usw. erfordern eine Strategie der Testfallauswertung, die einen über einen Einzelfall hinausgehenden *Eindruck* der Qualität von Bereichen der Wissensbasis vermittelt. Die erarbeitete Strategie der Testfallauswertung liefert Hinweise auf Bereiche der Wissensbasis, die Kandidaten für eine Qualitätsverbesserung sind. Qualitätsverbesserung beinhaltet die Eliminierung von Fehlern und die Verbesserung der Expertise.

Die Methode der Qualitätsmessung: Die Qualitätsmessung der Ergebnisse der Analyse ist nicht nur für die Validierungsphase wichtig, um beispielsweise ein Abbruchkriterium zu erhalten, sondern auch für die vergleichende Beurteilung mit alternativen Verfahren. Ziel der Arbeit ist es, die Qualitätsmessung so zu definieren, daß sie Informationen für das Validierungsverfahren (Strategie, Abbruch) und zum Vergleich mit alternativen Verfahren liefert.

Vor der Erläuterung des Ansatzes zum methodischen Testen einer Wissensbasis in einem wissensbasierten System werden zunächst Grundlagen zur wissensbasierten Bildanalyse in Kapitel 2 vorgestellt.

Kapitel 3 beschäftigt sich mit der Validierung von konventioneller Software, wissensbasierten Systemen insbesondere, in der Medizin, um in die Problematik der Validierung einzuführen und den Stand der Entwicklung aufzuzeigen.

Kapitel 4 stellt das Verfahren zum methodischen Testen vor. Es werden die für die Validierung relevanten Eigenschaften der Ein- und Ausgabedaten einer Analyse erarbeitet, die für die Teststrategie von Bedeutung sind. Aus der Teststrategie leitet sich die Systematik der Testfallgenerierung ab. Anhand der systematisch erzeugten Testfälle werÂ**?**den Fehler detektiert. Die Lokalisation der Fehler ist ein weiterer Schritt, der durch die Durchsuchung des Validierungsobjekts realisiert wird. Die Qualitätsbeschreibung wird als Informationsquelle für die Qualitätssicherung und für den Anwenderkreis erarbeitet.

Am Beispiel von KardioAssistant, einem wissensbasierten System zur Interpretation von Herzsequenzszintigrammen, wird die Validierung der Wissensbasis VentriKS mit der entwickelten Validierungsmethode unter Verwendung einer speziellen Testfallgenerierung (Herz-Phantom) in Kapitel 5 erläutert. Zunächst wird in medizinische Grundlagen eingeführt, die für das Verständnis strategischer Erwägungen notwendig sind. Um einen Eindruck von KardioAssistant zu erhalten, wird seine Architektur vorgestellt. Die Spezifikation und die Testfallgenerierung sind Voraussetzung für die Anwendung der Testmethode und werden vor der Beschreibung der Durchführung der Validierung eingehend erläutert.

Der Qualitätsmessung der Analysen von VentriKS für den Anwendungsbereich anhand der Validierungsergebnisse ist Kapitel 6 gewidmet. Insbesondere ist hier die Definition von Verfahren der Vorklassifikation von Eingabedaten und der Nachklassifikation von Analyseergebnissen von Bedeutung.

Bislang wurde der Begriff *wissensbasiertes System* für eine Software, die intelligentes Expertenverhalten simulieren soll, verwendet. Als alternativ dazu wird der Begriff *Expertensystem* verstanden. Da es nach Meinung der Autorin bis heute nicht gelungen ist, tatsächlich ein System zu bauen, das sich einem menschlichen Experten vergleichbar verhält, erscheint der Begriff *Expertensystem* vermessen. Der Begriff *wissensbasiertes System* bezeichnet die Situation angemessener und wird auch weiterhin verwendet, sofern nicht die zitierten Autoren den Begriff Expertensystem verwenden.

Kapitel 2

Architektur eines wissensbasierten Bildanalysesystems

Bilder stellen die Eingangsdaten in einem Bildanalysesystem dar. Sie ergeben sich aus einer Menge von Sensordaten, die für die Darstellung und Verarbeitung im Rechner zu quantisieren sind. Ein Bildpunkt kann allgemein formal durch die vektorielle Funktion $g(x, y, t)$ in Abhängigkeit von den Ortskoordinaten x und y und der Zeit t beschrieben werden. Der auftretende Informationsverlust bei der Quantisierung von x, y und t wird durch das Abtasttheorem beschrieben [LIE89]. Ein Sensor ist in diesem Fall ein Gerät, das physikalische Größen in elektrische Spannungen wandeln kann. Im Prinzip ist jede Größe, die physikalisch meßbar oder berechenbar ist, bildlich darstellbar. Im Rechner ist ein Bild als Matrix von Bildpunkten repräsentiert.

2.1 Bildanalytische Vorgehensweise

Als Bildanalyse wird die Vorgehensweise bezeichnet, mit der man aus Bildern eine Beschreibung oder Interpretation ableitet. Die Vorgehensweise kann visuell mittels Bildbetrachtung durch einen Experten, rechnergestützt mit Interaktionen zwischen dem Programm und einem Benutzer oder vollautomatisch erfolgen. Von welcher Art die abgeleitete Beschreibung oder Interpretation ist, hängt von der Fragestellung ab, die im allgemeinen auch die Motivation für die Bildgenerierung liefert. Im Fall von medizinischen Bilddaten ist es beispielsweise eine Indikation, die zu einer diagnostischen Beurteilung der Bilddaten führt. Niemann und Bunke [NIE87] definieren Bildanalyse in einem strengeren Sinn. Für sie ist Bildanalyse die automatische Generierung einer den Anforderungen des Anwenders genügenden symbolischen Beschreibung der zugehörigen Sensorsignale. Durch die Forderung nach einem automatischen Ablauf ist der Anspruch an die Qualität der Ergebnisgenerierung besonders zu unterstreichen. Bei der automatischen Analyse ist der Einfluß von Beurteilungen, die durch Interaktion in die Analyse einfließen, ausgeschlossen und somit auch ein Einfluß des

Benutzers auf die Qualität.

Um das Ziel der problemspezifischen Beschreibung des Bildes zu erreichen, hat sich eine allgemeine Vorgehensweise der Bildanalyse, die hier auch als Bildverstehen bezeichnet wird, manifestiert. Unter dem Einsatz von a-priori-Wissen über den Bildinhalt und die Bildentstehung wird mittels allgemeinerem Wissen über die im Bild repräsentierte Szene und die Objekte eine auf eine bestimmte Fragestellung bezogene Interpretation abgeleitet. Der Ablauf bei der Bildanalyse (Abbildung 2.1) setzt sich aus mehreren Abschnitten zusammen, in denen die Bilddatenverarbeitung durch problemorientierte Transformationsverfahren definiert wird. Die dabei erreichten Zwischenziele und das Endziel entsprechen den folgenden Zuständen:

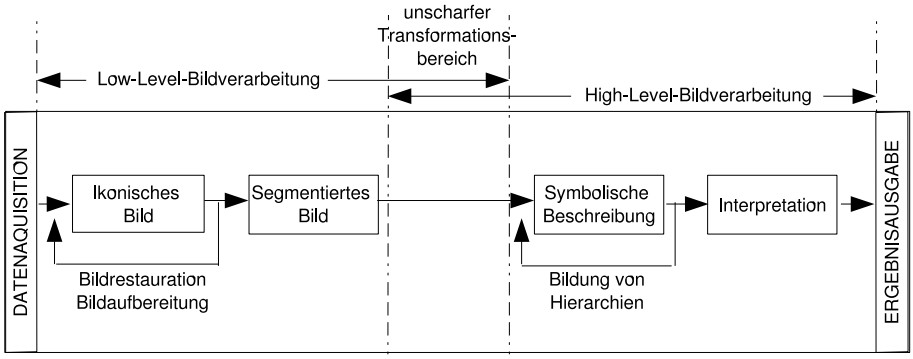

Abbildung 2.1: Überblick über die Vorgehensweise beim Bildverstehen nach [LIE89]

1. ***Ein ikonisches Bild*** *ist eine Matrix von Bildpunkten, deren Werte physikalisch meßbare Größen repräsentieren.*

2. ***Ein segmentiertes Bild*** *ist eine Matrix von Bildpunkten, deren Werten Bedeutungen zugeordnet sind.*

3. ***Eine symbolische Beschreibung*** *basiert auf zusammengefaßten Bildpunktgruppen mit derselben Bedeutung, denen symbolische Namen, Attribute und Relationen zugeordnet sind*

4. ***Die Interpretation*** *des Bildes ist das Analyseergebnis, das die Antwort auf die problemspezifische Fragestellung darstellt.*

Die Abschnitte 1 und 2 werden als Low-Level-Bildverarbeitung, die Abschnitte 3 und 4 als High-Level-Bildverarbeitung bezeichnet. Die Grenze zwischen High- und Low-Level-Bildverarbeitung ist nicht streng zu setzen, sondern eher als fließend zu bezeichnen.

2.2 Die Low-Level Phase in der Bildanalyse

Die Low-Level-Phase umfaßt die Bildvorverarbeitung und die Bildsegmentierung. Zu den typischen Aufgaben der Bildvorverarbeitung zählen die Bildrestauration (image restoration) und die Bildverbesserung (image enhancement). Das Ziel ist die qualitative Verbesserung der Werte der Bildpunkte zur Erleichterung der nachfolgenden Analyse und zur Verbesserung der Qualität des Analyseergebnisses.

Während der Bildakquisition wirken verschiedene Einflüsse verschlechternd auf die Bildqualität. Als Bildrestauration wird der Prozeß bezeichnet, der mittels bestimmter Operationen auf den akquirierten Bildern das durch Störungen überlagerte ideale Bild abschätzt. Zu den Störungen zählen:

- Rauschen
- geometrische Verzerrungen
- Bewegungen des Objekts oder der Kamera während der Aufnahme

Unter Bildverbesserung werden die Verfahren zusammengefaßt, die zu einer Verbesserung eines visuellen Eindrucks oder zu einer Verbesserung für eine durch einen Menschen oder eine Maschine durchgeführte Analyse führen. Weiterhin umfaßt die Bildverbesserung Verfahren zur Betonung oder Extraktion problemabhängiger Bildinformation, z.B. um mit einem Kantenoperator Kandidaten für eine Objektkontur aus der Bildpunktmenge hervorzuheben. Weitere Möglichkeiten ergeben sich durch Kontraststeigerungen oder Histogrammodifikationen [PRA78].

Die Bildvorverarbeitung in der Low-Level-Phase wird auf ikonischen Bildern ausgeführt und liefert als Ergebnis wiederum ikonische Bilder. Die Operationen zur Bildsegmentierung, die in einem segmentierten Bild resultieren, ordnen einzelnen Bildpunkten Bedeutungen zu. Es werden dabei nur primitive Objekte ermittelt wie Linien, Flächen oder Punkte, denen keine weitergehende Bedeutung zugeordnet ist. Ein Kriterium für eine spätere Zuordnung kann dann beispielsweise ein Schwellwert sein, der angibt, ab welchem Wert eines Bildpunktes dieser einem Objekt zuzuordnen ist. Das Wissen zur Segmentierung ist in dieser Phase im allgemeinen nicht explizit bereitgestellt. Vereinzelte Ansätze, die eine qualitative Verbesserung der Segmentierungsergebnisse beliebiger Bilddaten durch einen Experten-System-Ansatz anstrebten, konnten sich nicht durchsetzen [NAZ84], [HWA86]. Erst durch die Einfügung problemspezifischen Wissens scheint der erhöhte Aufwand für spezielle Lösungen gerechtfertigt [STA86], [DUN87]. Der Erfolg ist dennoch fraglich, da sich derartige Systeme z.B. in der medizinischen Praxis bislang nicht etablieren konnten.

2.3 Die High-Level-Phase in der Bildverarbeitung

Der Übergang zwischen Low-Level und High-Level-Bildverarbeitung ist geprägt durch die Transformation von numerischer zu symbolischer Bildverarbeitung. Die Bildpunkte, denen

Bedeutungen zugeordnet sind, werden zu Objekten zusammengefaßt, die durch symbolische Namen gekennzeichnet sind. Die Objekte zeichnen sich nicht mehr durch die numerischen Werte von Bildpunkten aus, sondern durch Attribute, die das Objekt beschreiben und Relationen, die ihre Einbettung in die im Bild repräsentierte Szene wiedergeben. Die symbolische Beschreibung liefert die Grundlage für die computergestützte Interpretation bezüglich einer Fragestellung. Mittels der symbolischen Repräsentation werden qualitative Aussagen abgeleitet, die in die Beschreibung einfließen. Aussagen wie *zu groß* oder *häufig* können mittels Attributen, Aussagen wie *innerhalb* oder *besitzt* mittels Relationen repräsentiert werden. Eine Assoziation dieser Aussagen mit den durch die Fragestellung vorgegebenen Klassen oder Teilmengen davon entspricht der Interpretation. Im Unterschied zur Low-Level-Phase ist das Wissen über den Problembereich in der High-Level-Phase explizit repräsentiert.

2.4 Konzept eines wissensbasierten Bildanalysesystems

In einem automatischen Bildanalysesystem sind die in Abbildung 2.2 aufgeführten vier Komponenten prinzipiell enthalten. Um das implizit im Low-Level-Bereich und das explizit im High-Level-Bereich repräsentierte Problemlösungswissen zu nutzen, sowie zur Darstellung von Zwischen- und Endergebnissen, ist eine Kontrolle notwendig, die die Abarbeitung und Visualisierung steuert. Von einem *idealen* System nach Niemann und Bunke [NIE 87] wird weiterhin die Fähigkeit zur Erklärung und zum Lernen erwartet. Der Bedarf wird in Bezug auf eine qualitative Verbesserung der Ergebnisse und eine Erhöhung der Akzeptanz erkannt. Mittels der Erklärungskomponente wird sie durch die direkte Evaluierbarkeit der Systemanalyse erreicht. Die Lernfähigkeit wird als Merkmal eines *wirklich leistungsfähigen* Systems betrachtet. Die Erweiterung des Leistungsspektrums um die Lernfähigkeit bewirkt die verstärkte Unabhängigkeit des Systems von einer extern betriebenen Aktualisierung des Wissens. Jeder durch das System analysierte Fall hat eine Qualitätsverbesserung zur Folge. Jedoch kann ein Bildanalysesystem auch ohne Erklärungs- und Lernkomponente hinreichend leistungsfähig sein. Dies hängt im wesentlichen von den Anforderungen an das System und den Einsatzbereich ab.

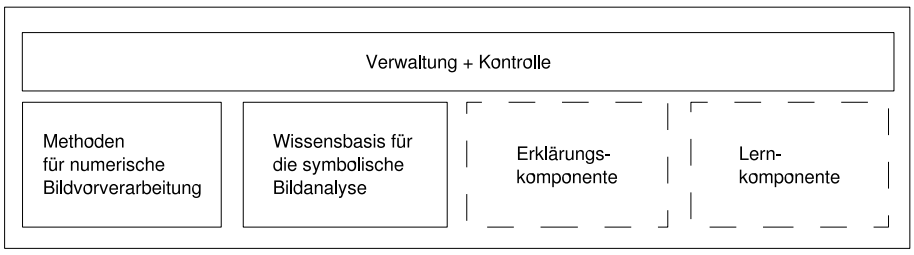

Abbildung 2.2: Komponenten eines Bildanalysesystems

Wie bei anderen Systemen der künstlichen Intelligenz wird bei Bildanalysesystemen mittels Expertenwissen und heuristikbasierten Inferenzstrategien die Lösung eines Problems angestrebt. Ein typisches Merkmal, das ein automatisches Bildanalysesystem nach dem Modell von Niemann und Bunke von anderen Expertensystemen unterscheidet, ist die Interaktion. Expertensysteme im allgemeinen agieren interaktiv mit dem Benutzer. Dadurch fließt ein Anteil von subjektiven Beurteilungselementen von außen in die Entscheidungsfindung ein. Die Interpretation durch ein Bildanalysesystem erfolgt nach dem Modell von Niemann und Bunke ausgehend von den originalen Bilddaten bis zum Analyseergebnis vollautomatisch. Die Bildelemente sind objektiv gemessene physikalische Ereignisse. Jede Behandlung dieser Daten oder Ableitung von Analysen ist durch die Methoden der Vorverarbeitung und das Analysewissen in der Wissensbasis repräsentiert. Natürlich gilt auch für automatische Bildanalysesysteme, daß eine Entscheidung auf der Basis von subjektiv repräsentiertem Expertenwissen als subjektiv zu bezeichnen ist. Jedoch ist jede Entscheidung im Vorverarbeitungs- und Analyseprozeß reproduzierbar. Eine Analyse, die durch Interaktion zugeführte Daten verwendet, ist nur schwer, teilweise überhaupt nicht zu reproduzieren.

Bekannte medizinische Expertensysteme wie MYCIN [SHO76], CASNET [KUL82] oder INTERNIST-1 [MIL82] [MIL84] erwarten Eingaben, die auf Daten basieren, die teilweise das Ergebnis einer vorangestellten Beurteilung sind. Bei einer Konsultation von INTERNIST-1, einem System zur computergestützten Diagnose in der Inneren Medizin, besteht der erste Schritt aus der Eingabe der Patientenbefunde, die u.a. die Symptome beschreiben. Als Symptom bei der alkoholbedingten Hepatitis wird in [MIL84] u.a. *akuter Leibschmerz* oder *massiv vergrößerte Leber* angegeben. Die Feststellung dieser Symptome basiert auf einer ärztlichen Beurteilung, die angelehnt an die vom Expertensystem vorgegebene Syntax zu formulieren ist. Der Vorteil dieses Verfahrens liegt in der Reduktion objektiv meßbarer Information und Hinzufügung von Expertenwissen. Dies kann zur Verminderung von Fehlerhäufigkeit bei der Analyse führen, da undefinierte Zwischen- oder Endzustände nicht erreicht oder abgefangen werden. Andererseits ist die Abhängigkeit vom Wissen des jeweiligen Benutzers kritisch zu betrachten, wenn das System unverändert einem größeren Benutzerkreis zur Verfügung gestellt werden soll. Verschiedene Benutzer werden bei der Verwendung des Systems auf der Basis gleicher objektiv gemessener Information durch die Hinzufügung ihrer subjektiven Wissens- und Erfahrungswerte zu mehr oder weniger differenzierten Ergebnissen gelangen. Dies ist bei der automatischen Bildanalyse nach Niemann und Bunke ausgeschlossen.

Kapitel 3

Validierungsverfahren

Validierung als Aktion der Qualitätssicherung ist eines der großen Aufgabengebiete im Bereich der Entwicklung konventioneller Programme. Bis heute sind die Probleme nicht vollständig gelöst. Das Ziel der Validierung kann man allgemein umschreiben mit dem Streben nach der Garantie für *gute Qualität* oder hier speziell für *Korrektheit* von Software. Zur Erreichung des Ziels werden verschiedene Methoden und Techniken formuliert. Die Ansätze umfassen Maßnahmen zur Qualitätssicherung während der Entwicklungsphasen und die Überprüfung des fertigen Programmes. Die Literatur zur Validierung konventioneller Software ist vielfältig. Vergleichsweise gering sind dazu die Ergebnisse der Forschung auf dem Gebiet der Validierung von wissensbasierten Systemen. In der Literatur werden die Vokabeln Validierung, Verifikation und Evaluierung mit mehr oder weniger differenzierten Bedeutungen belegt. Unabhängig von den von der Autorin gewählten Bedeutungen in Anlehnung an die Definition nach [ISO91] (siehe Kapitel 3.1), werden auch die aus der Literatur entnommenen abweichenden Bedeutungszuordnungen diskutiert.

Die Unterschiede von Expertensystemen und konventioneller Software insbesondere in der Art der behandelten Probleme und der verwendeten Typen von Programmiersprachen führten nach Meseguer [MES92] dazu, daß einige Autoren die Validierung von Expertensystemen als unterschiedlich zu der von konventioneller Software betrachten. Neben der eher technischen Beschreibung des Validierungsziels aus der Sicht der Entwicklung, die auf eine Liste von Qualitätsanforderungen ausgerichtet ist, definieren Adrion et al. [ADR82] in Übereinstimmung mit Meseguer [MES92] das Validierungsziel aus der Sicht der Benutzer:

> *Das Ziel der Validierung ist die Feststellung der Korrektheit eines Programmes in Bezug auf die Bedürfnisse und Forderungen der Benutzer.*

Die Übernahme der entwickler- oder benutzerorientierten Zieldefinition der Validierung konventioneller Software für die Qualitätssicherung wissensbasierter Systeme impliziert nicht, daß die Techniken und Aktivitäten der Validierung ausreichend sind, um die Zielvorgabe zu erfüllen. Es ist klar, daß die Komplexität der Probleme bei wissensbasierten

Systemen über die bei konventionellen Programmen hinausgeht. Hoenen et al. [HOE92] diskutieren kritisch die Möglichkeit von Verifikation und Überprüfbarkeit von wissensbasierten Systemen. Verifikation definieren sie als formalen Beweis der Korrektheit der Software in Bezug auf ihre (formale!) Anforderungsspezifikation. Daneben steht die Überprüfung durch einen Test der Programmeigenschaften. Die Möglichkeit der Verifikation schließen sie aus, da es sich bei dem Problembereich um ein relativ ungenau definiertes Feld handelt. Das Expertenwissen reicht vom Regelwissen über weniger gesichertes heuristisches Wissen bis zum allgemeinen *Weltwissen*. Die Gründe, die die formale Überprüfung verhindern, führen sie an mit:

Unvollständigkeit der Wissensbasis: Der Prozeß der Explizierung des Wissens wird zu einem bestimmten Zeitpunkt abgebrochen unabhängig davon, ob überhaupt alles Wissen explizierbar ist.

Subjektivität der Wissensbasis: Das Wissen wird oft bei einzelnen Experten akquiriert und besitzt damit eine subjektive Gültigkeit. Es zeigt sich, daß Experten sogar die Terminologie ihrer Domäne unterschiedlich benutzen, und es so zu Mißverständnissen kommen kann.

Dekontextualisierung des Wissen: Bei der Wissensakquisition wird das Wissen aus dem Gesamtzusammenhang in überschaubaren Ausschnitten zusammengefaßt. Der Zusammenhang zwischen den Ausschnitten und zum größeren Kontext kann nur teilweise erfaßt werden.

Reflexions- und Kommunikationszusammenhang: Bei der Wissensakquisition erfolgt auf seiten des Experten eine Reflexion des Problemfelds als auch seines Problemlöseverhaltens. Der Wissensingenieur deutet und interpretiert die erhaltenen Informationen. Sein Bewußtsein der begrenzten Gültigkeit, Vorläufigkeit oder Subjektivität verschwindet bei der Konstruktion des Systems und in der späteren Nutzung.

Unspezifische Systemgrenzen: Die Problembereiche wissensbasierter Systeme sind relativ ungenau abzugrenzen. Es fehlt eine eindeutige Beschreibung der Leistungen des Systems und der Grenzen seiner Leistungsfähigkeit.

Einen Ausweg sehen sie in einem anderen Konzept der Systementwicklung, in dem die Bereiche des Systems, für die eine Spezifikation möglich ist, herausgearbeitet werden. Für sie wäre somit eine partielle Verifikation oder ein Test denkbar. Die Bereiche, die nicht verifiziert oder getestet sind, müßten für die Benutzer zugänglich sein, indem sie jedes einzelne Ergebnis nachprüfen könnten.

Eine Qualitätssicherung der Korrektheit wissensbasierter Systeme wird zur Zeit nur mit Methoden durchgeführt, denen aufgrund ihrer unzureichenden Möglichkeiten in Bezug auf die Anforderungen der Qualitätssicherung eher ein vorläufiger Charakter zuzuschreiben ist. Für die zukünftig zu entwickelnden Methoden wird im allgemeinen nicht

davon ausgegangen, daß die wissensbasierten Systeme vollständig neue Methoden zu ihrer Qualitätssicherung benötigen. Nach Meseguer [MES92] kann die angesammelte Erfahrung aus der Validierung konventioneller Software nicht ignoriert werden, sondern sollte unter Berücksichtigung der essentiell zu lösenden Aufgaben eingesetzt werden.

In den folgenden Kapiteln wird deshalb nach einer allgemeinen Begriffsbestimmung in Kapitel 3.1, die der Einordnung dieser Arbeit in den Bereich der Qualitätssicherung dient, in Kapitel 3.2 ein Überblick über allgemeine Validierungsverfahren für konventionelle Software gegeben. Darauf folgt in Kapitel 3.3 eine Diskussion der Probleme für die Validierung von wissensbasierten Systemen mit der Darstellung einzelner Ansätze. Anschließend werden die Ansätze zur Lösung der Validierungsproblematik bei medizinischen wissensbasierten Systemen in Kapitel 3.4 vorgestellt.

3.1 Begriffsbestimmungen

Neben dem Begriff der Validierung werden häufig die Begriffe Verifikation, Testen und Evaluierung oder eingeschränkt auch Zertifikation benutzt. Jedoch ist ihre Definition in der Literatur nicht übereinstimmend. Mit allen Begriffen ist das Ziel, *gute Qualität* von Software zu erreichen oder zu bestätigen, verbunden. Testen im Bereich der Validierung von konventioneller Software bezeichnet nur eine Gruppe von möglichen Aktivitäten zur Feststellung der Korrektheit eines Programmes. An dieser Stelle sollen im Vorgriff auf die Kapitel, in denen die verschiedenen Ansätze differenziert betrachtet werden, allgemeine Begriffsklärungen durch Einführung von Definitionen erfolgen. Nach [ISO87a] und entsprechend [DIN92] sind Verifikation und Validierung wie folgt definiert (Abbildung 3.1):

> **Verifikation** ist der Vorgang, Ergebnisse einer gegebenen Phase zu bewerten, um ihre Richtigkeit und ihre Erfüllung der Vorgaben für diese Phase sicherzustellen.
>
> **Validierung** ist der Vorgang, Software zu bewerten, um die Erfüllung festgelegter Forderungen sicherzustellen.

Die Definitionen sind ähnlich mit denen nach [IEEE83]. Des weiteren definiert [IEEE83] Testen wie folgt:

> **Testen** ist der Prozeß, ein Software-Produkt durch manuelle oder automatisierte Hilfsmittel zu bewerten, um damit die Erfüllung der spezifizierten Anforderungen nachzuweisen. Dabei können Abweichungen zwischen den erwarteten und aktuellen Ergebnissen identifiziert werden.

In DIN ISO 8402 [DIN92] wird zur Validierung angemerkt, daß sie üblicherweise an einem Endprodukt unter festgelegten Betriebsbedingungen durchgeführt wird. Sie kann aber auch schon in früheren Phasen nötig sein.

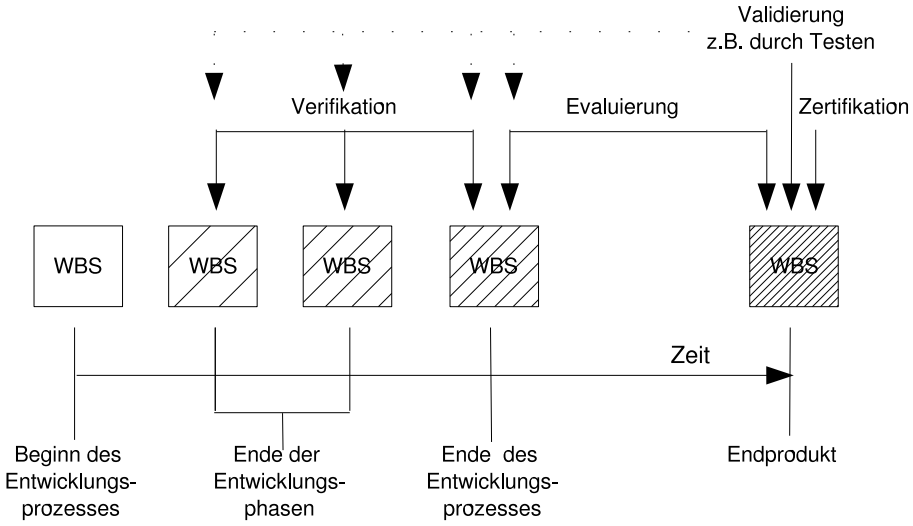

Abbildung 3.1: Zeitliche Zuordnung von Verifikation, Validierung, Testen nach [ISO87a] und Evaluierung eines wissensbasierten Systems (WBS) vom Beginn des Entwicklungsprozesses bis zur Applikation. Die Schattierung soll den Entwicklungsprozeß des WBS verdeutlichen von der Modellierung (weiß) bis zum fertigen Produkt (schwarz). Validierung wird üblicherweise am Endprodukt durchgeführt, kann aber auch schon früher nötig sein.

Neben den Begriffen Verifikation, Validierung und Testen wird oft der Begriff Evaluierung (Abbildung 3.1) für eine allgemeine Leistungsbewertung verwendet. Die ähnlichen Ziele von Validierung und Evaluierung scheinen zunächst eine Trennung der Begriffe zu erschweren. Die Unterscheidung wird hier durch die Sichtweise bei der Durchführung der Aktivitäten getroffen. Validierung ist ein Teil des Entwicklungsprozesses und wird aus der Sichtweise der Entwickler betrieben, die sich an den spezifizierten Anforderungen orientieren. Die Einbindung eines potentiellen Benutzers ist höchstens bei der Formulierung der Spezifikation notwendig. Bei der Evaluierung liegt ein Produkt nach dem Ende des Entwicklungsprozesses vor, das aus der Sicht eines potentiellen Benutzers hinsichtlich seiner benutzerorientierten Leistungsfähigkeit überprüft wird. Nach Gaschnig et al. [GAS83] umfaßt die Evaluierung die Beurteilung allgemeiner Qualitätsmerkmale und die Feststellung, wie gut das System im Vergleich zu einem Experten arbeitet. Gaschnig et al. beschreiben das eigentliche Ziel der Evaluierung damit, eine weitergehende Bewertung des Prozesses, der Struktur und der Grenzen der Expertise zu erhalten. Es geht nicht darum zu zeigen, wie gut das System das tut, wozu es entworfen ist.

Unter dem Begriff Zertifikation ist die Prüfung und Bewertung eines Software-Produkts zu verstehen, mit dem die Erfüllung vorgegebener Anforderungen nachgewiesen wird. Bei Erfüllung der gestellten Anforderungen wird von einer unabhängigen Prüfstelle ein Zerti-

fikat ausgegeben [WAL90].

Während die Begriffe Verifikation und Validierung bzw. Evaluierung die Aktivitäten zeitlich im Entwicklungsprozeß lokalisieren (Abbildung 3.1), erfaßt der Begriff des Testens eine Gruppe von Methoden als Basis für diese Aktivitäten. Testen ist ein Prozeß, bei dem ein Programm ausgeführt wird, um Fehler zu finden [MYE79], nicht die Fehlerfreiheit von Programmen zu zeigen, wie es beim Beweis der Korrektheit versucht wird.

Wird ein Fehler bemerkt, erfolgt über den Prozeß des Debugging die Lokalisation der Ursache des Fehlers, die Planung, Prüfung, Durchführung und Dokumentation der Korrektur. Danach wird im allgemeinen der Test wiederholt, um die Auswirkung der Änderung zu prüfen. Neben der Feststellung, ob der Fehler eliminiert wurde, ist zu prüfen, ob keine neuen Fehler eingeführt wurden. Der Begriff Fehler hat im deutschen Sprachgebrauch eine vielfältige Verwendung. Die englische Sprache bietet die differenzierteren Begriffe error, defect, fault oder failure. [IEEE83] definiert einen Fehler, der hier für das englische Wort *error* gesetzt ist, wie folgt:

> **Fehler** ist die Abweichung zwischen dem berechneten, beobachteten oder gemessenen Wert oder einem Zustand der Betrachtungseinheit und dem entsprechenden spezifizierten oder theoretisch richtigen Wert.

Wallmüller [WAL 90] differenziert zusätzlich zwischen Defekt (defect), Ausfall (failure) und Störung (fault). Nach seiner Definition ist

> ein **Defekt** eine Abweichung von der festgelegten (erwarteten) Ausprägung eines Merkmals einer Betrachtungseinheit ein **Ausfall** die Beendigung der Fähigkeit der Betrachtungseinheit, die geforderte Funktion auszuführen eine **Störung** die Unfähigkeit der Betrachtungseinheit, ihre geforderte Funktion auszuführen

Betrachtungseinheiten sind z.B. Programme. Defekte können die Funktionstüchtigkeit des Programms beeinträchtigen, müssen es aber nicht. Ein Ausfall kann zu einer Betriebsbeeinträchtigung oder zu katastrophalen Auswirkungen übergeordneter Einheiten führen. Eine Störung kann sowohl durch einen Ausfall, einen Defekt oder einen Fehler verursacht werden.

Beispielsweise kann der Ausfall eines Datenbanksystems in einer ärztlichen Praxis die Störung eines Programms verursachen, das die vierteljährlichen Abrechnungsarbeiten durchführt. Die Störung kann auch durch einen Software-Fehler in einem Programm verursacht werden, das die ärztlichen Leistungen verbucht.

3.2 Validierung von konventioneller Software

Die verfügbaren Prüfmethoden zur Validierung konventioneller Software lassen sich in zwei Kategorien, statische Prüfungen, dynamische Prüfungen einteilen.

Zu den statischen Prüfungen zählen Audits, Reviews, statische Analysen, Beweise und symbolische Ausführungen. Nach Evans und Marciniak [EVA87] sind Audits gekennzeichnet durch Produkt-, Prozeß- oder Systembewertungen aus einer neutralen Sicht, dem Vorschlag von Maßnahmen zur Verbesserung und dem Einsatz von Checklisten. Bei einem Audit der Produktqualität wird die Übereinstimmung des Produkts mit den Anforderungen an Produktmerkmale quantitativ bewertet.

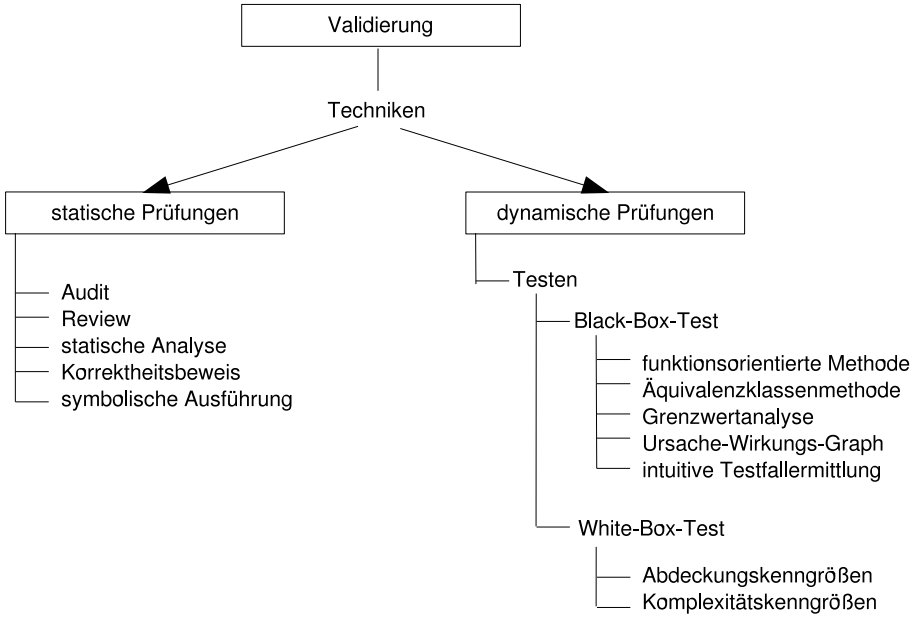

Abbildung 3.2: Übersicht über die Techniken zur Validierung konventioneller Software

Ein Review ist ein geplanter und strukturierter Analyse- und Bewertungsprozeß. In dem Prozeß werden die Projektergebnisse einer Gruppe von Gutachtern präsentiert. Die Gutachter stimmen den Ergebnissen zu oder kommentieren sie. Bei den Reviews wird zwischen Inspektionen und Walkthroughs unterschieden. Inspektionen sind formaler geplant und durchgeführt als Walkthroughs.

Die statische Analyse mit Software-Werkzeugen ist die rechnergestützte Prüfung und Bewertung von Qualitätsmerkmalen eines Produktes. Untersucht werden u.a. die Form, die Struktur, der Inhalt oder die Dokumentation. Durch die statische Analyse lassen sich Aussagen über die Merkmale *syntaktisch korrekt* oder *semantisch korrekt* des Objektes machen. Die semantische Korrektheit kann nur so weit analysiert werden, wie die Semantik formal beschrieben ist. Beispiele für statische Programmanalysatoren sind in [DEU82] und [HUA81] beschrieben.

Der Korrektheitsbeweis beinhaltet eine formale Vorgehensweise zur Analyse der funktionalen Korrektheit eines Programms. Die Methode setzt mathematische Beweisverfahren

ein, um die Übereinstimmung des Programms mit seiner Spezifikation zu beweisen. Um das zu erreichen, wird schrittweise für jede Programmoperation bewiesen, daß sich die beschriebene Nachbedingung einer Operation aus der beschriebenen Vorbedingung ergibt. Es ist weiterhin zu zeigen, daß das Programm bei Erfüllung der Vorbedingung terminiert. Der Beweis betrifft ausschließlich das Qualitätsmerkmal Korrektheit. Die Praktikabilität der Methode unterliegt verschiedenen Beschränkungen:

1. Die manuelle Beweisführung ist fehleranfällig, der Einsatz von rechnergestützten Werkzeugen zur Beweisführung reduziert die Wahrscheinlichkeit von Fehlern [WAL90].
2. Auch wenn der Beweis gefunden und die Beweisführung formal in Ordnung ist, kann man daraus nicht die Aussage ableiten, daß die Spezifikation richtig ist, und das Programm die ursprüngliche Aufgabe löst [WAL90].
3. Auch wenn theoretisch die Beweisbarkeit oder die partielle Korrektheit nachgewiesen werden kann, ist nicht sicher, ob der Beweis in vertretbarer Zeit zu finden ist [WAL90].
4. Bei der Vergrößerung des Programmumfangs liegt die Vergrößerung des Aufwands für den Beweis proportional höher [END77].
5. Selbst bei relativ kleinen Programmen ist der Aufwand zur Durchführung eines Beweises relativ groß [RAM82].

Die symbolische Ausführung ist eine Programmanalysemethode, bei der statt Datenwerten symbolische Ausdrücke zur Ausführung von Programmpfaden eingesetzt werden. Als Ergebnis liegt im allgemeinen ein relativ langer und komplexer Ausdruck vor, der zerlegbar ist. Interpretiert man den Ausdruck als Baumstruktur, repräsentiert jeder Verzweigungspunkt einen Entscheidungspunkt im Programm. Die Ergebnisse der Pfadauswertung sind mit den entsprechenden Spezifikationen des Programms vergleichbar. Ein Problem der Methode ist der erhebliche Berechnungsaufwand bei der hohen Anzahl der Pfade in größeren Programmen.

Die dynamischen Prüfungen sind durch die Ausführung des Prüfobjektes gekennzeichnet. Die Prüfung wird mit einer ausgewählten Menge von Eingabedaten durchgeführt. Die wichtigste Prüfmethode ist das Testen. Testobjekte sind Zwischen- und Endprodukte kompletter Programme oder Module und Komponenten im Software-Lebenszyklus. Als Module werden selbständig übersetzbare, aber nicht selbständig ausführbare Folgen von Anweisungen, die über einen Namen angesprochen werden können, bezeichnet. Mehrere Module bilden Komponenten. Mit Testen können sämtliche Qualitätsmerkmale überprüft werden.

Die Aufgabe des Testens ist es, mit möglichst geringem Aufwand potentiell vorhandene Fehler im Programm zu erkennen. Dabei werden verschiedenen Grundsätze verfolgt:

1. Es muß für jeden Test ein Endekriterium definiert sein. Dies wird eher über heuristische Ansätze verfolgt, da die Kriterien nur sehr schwer zu bestimmen sind.

2. Ein Testobjekt kann praktisch nicht vollständig getestet werden. Deshalb muß durch eine geeignete Testfallauswahl das Testen optimiert werden.

3. Die erwarteten Testergebnisse sind vor dem Test zu spezifizieren.

4. Testziele sind nur dann sinnvoll, wenn sie auch erreichbar sind, und dieses meßbar ist.

5. Testfälle dürfen nicht nur den Bereich der erwarteten und gültigen Eingabedaten abdecken, sie müssen auch unerwartete ungültige Eingabedaten berücksichtigen [WAL90].

6. Testen muß reproduzierbar sein [WAL90].

7. Ein Testfall wird als erfolgreich angesehen, wenn er einen unbekannten Fehler aufdeckt.

8. Module, in denen Fehler gehäuft auftreten, legen den Verdacht nahe, daß dort weitere Fehler zu erwarten sind [MYE89].

Zur Erfüllung der Aufgaben des Testens gehören verschiedene Aktivitäten. Es müssen die Ziele, die Methoden, die einzusetzenden Mittel und die aufzuwendende Zeit festgelegt werden. Die Testfälle sind auf dieser Basis zu ermitteln. Zu weiteren Aktivitäten zählen die Planung und Durchführung des Tests und die Zusammenfassung und Auswertung der Ergebnisse.

Bei dem zentralen Problem, den Testaufwand zu optimieren, bildet die Testfallermittlung einen Schwerpunkt. Goodenough und Gerhart [GOO75] erarbeiteten Grundlagen für eine Theorie der Testfallauswahl mit dem Ziel, die Charakteristiken eines idealen Tests zu definieren. Sie unterscheiden zwischen Testdaten und Testprädikaten. Testdaten sind die aktuellen Werte aus dem Eingabebereich eines Programmes, die ein Testauswahlkriterium vollständig erfüllen. Testprädikate sind Beschreibungen von Bedingungen und Kombinationen von Bedingungen, die für die korrekte Programmausführung relevant sind. Die Informationen für die Prädikate liefern die allgemeinen Anforderungen, die ein Programm erfüllen muß, die Programmspezifikation und allgemeine Charakteristiken der Implementierung. Gourlay diskutierte in [GOU83] neben [GOO75] weitere Ansätze zu formalen Grundlagen des Testens. Er erarbeitete einen mathematischen Rahmen, der vorhergehende Arbeiten zum Testen vereinheitlichte und einen Mechanismus, um die Leistung von Testmethoden zu vergleichen.

Die praktische Durchführung von Testen muß strategisch gut vorbereitet sein. Um Zeit und Kosten zu optimieren, ist es anzustreben, die kleinste Untermenge aller Testfälle einzusetzen, die die größte Wahrscheinlichkeit bietet, möglichst viele Fehler zu finden. Ein Ansatz ist das Testen mit Zufallsdaten. Duran and Ntafos bewerteten in [DUR84] diese Methode und kamen zu dem Ergebnis, daß sie für viele Programme kosteneffektiv sein

kann. Myers [MYE89] bezeichnet sie als die wahrscheinlich schwächste Methode, eine optimale Untermenge der denkbaren Testfälle zu liefern. Die praktisch wichtigen methodischen Grundlagen des Testens teilen sich ein in die zwei Gruppen

- Black-Box-Methoden
- White-Box-Methoden

Beim Black-Box-Test wird keine Information über die Struktur des Testobjektes verwendet. Es wird untersucht, ob das Testobjekt die Anforderungen weitestgehend erfüllt. Die Informationen über die Anforderungen fließen aus der Spezifikation ein. Daraus leitet sich auch die alternative Bezeichnung funktionaler oder entwurfsorientierter Test ab. Ein vollständiger Test enthält alle Kombinationen der Eingabewerte. Dies ergibt eine kombinatorische Vielfalt, die eine praktische Durchführung eines vollständigen Testes verhindert.

Der White-Box-Test verwendet Informationen über die Struktur des Testobjektes. Deshalb wird der White-Box-Test auch alternativ als strukturelle Testmethode bezeichnet. Ein Test ist vollständig, wenn jeder Pfad von jedem Eingang zu jedem Ausgang durchlaufen wird. Bei Schleifen kann sich beispielsweise eine explosionsartige Erhöhung der Anzahl von Pfaden ergeben. Praktisch ist eine vollständige Pfadabdeckung kaum möglich. Ein weiterer Nachteil ist, daß selbst für den Fall eines vollständigen Tests das Programm noch Fehler enthalten kann. Ein Vergleich mit der Spezifikation ist nicht Teil der Methode. Deshalb bleiben fehlende oder mangelnde Übereinstimmungen unbewertet. Maschinenspezifische Eigenschaften bleiben unberücksichtigt. Ein Beispiel ist der Vergleich zweier reeller Variablen auf Gleichheit, die nach der mathematischen Formel, aus der sie abgeleitet werden, den gleichen Wert besitzen, in der Maschinenrepräsentation sich aber geringfügig unterscheiden.

Myers [MYE 89] und Wallmann [WAL 90] bezeichnen bei einem Vergleich beider Test-Ansätze den Test, der auf vollständigen Black-Box-Testmethoden basiert, als überlegen gegenüber dem Test auf der Basis von vollständigen White-Box-Testmethoden. Da vollständige Tests praktisch unmöglich durchzuführen sind, empfehlen beide zunächst einen Entwurf nach Black-Box-Methoden, die durch White-Box-Methoden ergänzt werden. Auch Duran [DUR 84], der den Entwurf von Zufallstestdaten favorisiert, empfiehlt als Schlußfolgerung eine von Fall zu Fall unterschiedliche Kombination von Teststrategien, die auf dem Testen mit Zufallsdaten aufbauen.

Aufgrund der praktischen Bedeutung der Black-Box- und White-Box-Methoden folgt ein Überblick über die wichtigsten Methoden. Zu diesen Methoden zählen beim Black-Box-Ansatz

- die funktionsorientierte Methode
- die Äquivalenzklassenmethode
- die Methode der Grenzwertanalyse

- die Methode des Ursache-Wirkungs-Graphs
- die intuitive Testfallermittlung

Bei der funktionsorientierten Methode werden die Testfälle auf der Grundlage der Ein- und Ausgabespezifikation bestimmter Funktionen des Testobjektes ermittelt. Mit den ermittelten Testfällen werden Tests durchgeführt, um das Vorhandensein und die Ausführbarkeit der Funktionen bei Normalfällen nachzuweisen. Ausnahmen und Grenzfälle werden nicht untersucht.

Die Äquivalenzklassenmethode ermittelt die idealen Testfälle, die jeweils eine Klasse von Fehlern aufdecken. Ein idealer Testfall reduziert die Anzahl der Testfälle, die notwendig sind, um ein vorgegebenes Ziel beim Testen zu erreichen. Er deckt möglichst viele unterschiedliche Eingabemöglichkeiten ab, die die Klasse aufspannen. Er ist äquivalent zu allen anderen Testfällen in seiner Klasse. Es ist zu erwarten, daß alle Testfälle in der Klasse den gleichen Fehler aufdecken oder daß ein Fehler, der von einem Testfall nicht aufgedeckt wird, durch keinen anderen Testfall der Klasse aufgedeckt wird. Die Äquivalenzklassenmethode wird durch drei Schritte charakterisiert:

- die Bestimmung der Äquivalenzklassen durch Aufteilung der Ein- und Ausgabegrößen der Wertebereiche (z.B. mittels der Spezifikation)
- Definition der Testfälle durch Auswahl eines Wertes aus jeder Äquivalenzklasse
- Überprüfung, ob die Äquivalenzklassen die Ein- und Ausgabegrößen vollständig erfassen

Aus der Erfahrung, daß die Verarbeitung von Werten an den Grenzen von Wertebereichen häufiger zu Fehlern führt, als die Verarbeitung von *normalen* Werten, leiten Myers [MYE89] und Wallmann [WAL90] einen größeren Erfolg der Methode der Grenzwertanalyse ab. Bei dieser Methode repräsentieren Testfälle Werte, die an den Grenzen von Wertebereichen (Ein-/Ausgabeäquivalenzklassen) liegen. Die Testfälle werden so ausgesucht, daß jeder Rand der Äquivalenzklassen getestet wird.

Die Methode des Ursache-Wirkungs-Graphs erfaßt außer den Ein- und Ausgabegrößen auch Kombinationen von Eingabebedingungen. Der Ursache-Wirkungs-Graph ist eine formale Sprache, in die eine Spezifikation aus der natürlichen Sprache übersetzt wird. Als zusätzlicher Effekt der Methode können sich Hinweise auf unvollständige oder ungenaue Formulierungen in der Spezifikation ergeben. Die Knoten des Graphen sind Ursachen und Wirkungen. Eine Ursache, die eine bestimmte Wirkung hat, ist mit dieser über eine Kante oder eine Kantenfolge verbunden. Die Notation des Formalismus ist der Boole'schen Algebra entnommen. Nach der Abbildung der Spezifikation in einen Graphen wird dieser in eine Entscheidungstabelle umgesetzt, bei der die Bedingungen den Ursachen und die Aktionen den Wirkungen entsprechen. Jede Spalte der Tabelle repräsentiert einen Testfall.

Neben den aufgeführten Methoden der Testfallermittlung erwähnt [WAL90] die intuitive Testfallermittlung (error guessing), die ein auf heuristisches Vorgehen basierter Ansatz ist. Die Heuristik basiert auf der Intuition, Fehler aufzuspüren und auf der Erfahrung über häufige und immer wiederkehrende Fehler in allen Applikationen.

Die White-Box-Methoden verwenden Informationen über die Struktur des Testobjektes, die graphisch als Ablaufstruktur dargestellt werden. Die Knoten stellen Anweisungen oder Zusammenfassungen von sequentiell ausführbaren Anweisungen dar. Der Kontrollfluß wird durch die gerichteten Kanten zwischen den Knoten dargestellt. Die Methoden basieren auf Kenngrößen bezüglich der Testabdeckung des Testobjektes. Die Kenngrößen der Testabdeckung beschreiben Anweisungsabdeckung, Zweigabdeckung, Bedingungsabdeckung, Abdeckung aller Bedingungskombinationen und die Pfadabdeckung. Zum Beispiel ist die Anweisungsabdeckung das Verhältnis der Anzahl der durchgelaufenen Anweisungen zur Gesamtanzahl der Anweisungen eines Testobjektes. Neben den Abdeckungskenngrößen liefern Komplexitätskenngrößen Informationen über die strukturelle Komplexität des Testobjektes. Diese Information wird wiederum für die Ermittlung von Testfällen verwendet (z.B. die zyklomatische Zahl von McCabe [MCC76].

3.3 Validierung von wissensbasierten Systemen

Um festzustellen, welche besonderen Aufgaben die Validierung bei wissensbasierten Systemen im Vergleich zu konventioneller Software zu lösen hat, ist die Definition der Anforderungen an ein derartiges System ein Anhaltspunkt. Eine allgemeingültige Definition des Begriffs wissensbasiertes System existiert nicht. In einer idealisierten Vorstellung wird der Begriff des Expertensystems mit einem Rechenautomaten verknüpft, der ähnlich wie ein intelligentes Wesen in einer komplexen Umgebung wahrnehmen, denken und agieren kann. Die künstliche Intelligenz (KI) als Forschungsdisziplin der Informatik untersucht, wie intelligentes Verhalten von Computern erfaßt und nachvollzogen werden kann oder wie allgemein mit Hilfe von Computern Probleme zu lösen sind, die Intelligenzleistungen voraussetzen [DUD88]. Als Teilgebiet der KI, auf das sich das wirtschaftliche Interesse konzentriert, bezeichnet Schefe [SCH86] die Expertensysteme und beschreibt sie als intelligente Auskunfts- und Beratungssysteme. Nach Fox [FOX90] sind die Kernkonzepte der künstlichen Intelligenz Wissensrepräsentation und Suche im Problemraum. Aus programmtechnischer Sicht ist ein Expertensystem nach [DUD88] ein Programmsystem, das *Wissen* über ein spezielles Gebiet speichert, aus dem Wissen Schlußfolgerungen zieht und zu konkreten Problemen des Gebietes Lösungen anbietet. Für Expertensysteme ist typisch, daß sie auch diffuses, vages und unformalisiertes Wissen problembezogen repräsentieren. Aus dem repräsentierten Wissen werden mittels Logik oder Heuristik Schlußfolgerungen zur Gewinnung neuen Wissens gezogen. Die Annahme, daß diese Expertensysteme leicht zu verifizieren oder validieren wären, verweist Fox [FOX90] deutlich in das Reich der Legenden.

Ähnlich wie im Bereich der Validierung von konventioneller Software kann eine Validierung wissensbasierter Systeme nicht ohne eine Definition der Benutzeranforderungen (Spezifikation) erfolgen. Diese Anforderungen werden zunächst natürlichsprachlich formuliert (informelle Spezifikation), um dann in eine formale Spezifikation übersetzt zu werden. So wird für die Spezifikation konventioneller Software eine vollständige Übersetzung theoretisch als möglich angesehen. Nach Meseguer [MES92] gilt dies nicht für Expertensysteme. Er stellt fest, daß es Benutzeranforderungen gibt, die nicht vollständig in einer formalen Spezifikation ausgedrückt werden können. Aufgrund dieser Erfahrung unterscheidet er zwischen formalisierbaren und teilweise formalisierbaren Anforderungen. Als vollständig formalisierbare Anforderung bezeichnet er

> Die Wissensbasis sollte korrekt in der Struktur sein.

Die Anforderung wird in folgende Spezifikationen übersetzt, die in einer formalen Sprache ausgedrückt werden können:

1. Die Objekte der Wissensbasis sollten syntaktisch korrekt sein.
2. Die Wissensbasis sollte widerspruchsfrei sein.
3. Die Wissensbasis sollte keine Redundanzen enthalten.
4. Die Wissensbasis sollte zyklenfrei sein.
5. Alle Objekte der Wissensbasis sollten potentiell verfügbar sein.

Eine teilweise formalisierbare Anforderung ist beispielsweise

> **Das Exertensystem soll eine *akzeptable* Diagnose für alle *typischen* Fälle liefern.**

Die Aspekte der Validierung konzentrieren sich auf zwei Gruppen: die Überprüfung von Syntax und Semantik der deklarativen und prozeduralen Anteile gegen eine formale Spezifikation und die Bewertung der Expertenleistung gegen eine teilweise formale oder informelle Spezifikation. Zur Validierung der ersten Gruppe sind Methoden erforderlich, die ähnlich zu den Methoden für die Validierung konventioneller Software sind, da hier auch ähnliche Voraussetzungen vorliegen. Die zweite Gruppe repräsentiert den herausragenden Unterschied wissensbasierter Systeme zu konventioneller Software aus der Sicht der Validierung, nämlich die Expertise in Form der Repräsentation von vagem, unscharfem, unformalisiertem Wissen und der Ableitung von Schlußfolgerungen daraus mittels Heuristik oder Logik. Wenn unformalisiertes Wissen repräsentiert wird und die Basis für Schlußfolgerungen bildet, dann kann es keine Validierung gegen eine formalisierte Spezifikation geben.

Das Problem konzentriert sich auf eine auch für menschliche Experten nicht eindeutig geklärte Frage: wie sind die Fähigkeiten eines Experten zu messen, um einen umfassenden Eindruck der Leistungsmöglichkeiten zu erhalten.

Validierungsaspekte, die der ersten Gruppe zuzuordnen sind, werden von Nguyen et al. [NGU85], [NGU87] durch ein Programm, das in ein Wissensrepräsentationswerkzeug eingebunden ist, behandelt. Das Werkzeug ist ein generisches regelbasiertes Expertensystem-Tool (Lookheed Expert System, LES), aus dem heraus die entwickelte Wissensbasis validiert werden kann. Das Validierungsprogramm kombiniert logische Prinzipien ebenso wie spezifische Informationen über den Wissensrepräsentations-formalismus. Es sucht redundante, kollidierende, zusammenfaßbare, periodisch wiederkehrende, fehlende Regeln, unerreichbare Klauseln und Sackgassen. Die letzten beiden Typen stellen Erweiterungen des in Shortliffes System ONCOCIN [SHO81] eingesetzten Testmoduls dar. Die Prüfung erfaßt nicht die Bedeutung des in den Regeln repräsentierten Wissens für den Problembereich.

Validierungsaspekte, die der zweiten Gruppe zuzuordnen sind, werden durch verschiedene heuristische Verfahren überprüft. Prinzipiell erfolgt ein Test des Systems gegen einen Experten oder ein einem Experten äquivalentes Verfahren. Das Ideal wird in einem *Gold Standard* gesehen. Hierbei liegt für jeden Fall eine als korrekt akzeptierte Antwort zum Vergleich vor. Weitere Vergleiche können mit Aussagen von Experten getroffen werden, die parallel zum System die Fälle analysieren. Experten selbst können die Ergebnisse des Systems bewerten. Zur Evaluierung bietet sich eine Blindstudie an, bei der sich u.a. die Vorbehalte, die aus der Kenntnis entstehen, die Ergebnisse eines Computer-Programms zu bewerten, eliminieren lassen. In der Blindstudie werden Analyseergebnisse des Programmes gemischt mit Interpretationen, die auf andere Art und Weise (z.B. durch Experten) erzielt werden. Bei der Auswahl der Experten kann weiterhin differenziert werden zwischen Experten, die in die Entwicklung involviert sind, und solche, die an der Entwicklung nicht beteiligt sind. Als Grundlage für eine Methode zur Evaluierung potentieller wissensbasierter Applikationen erarbeiteten Laufman et al. [LAU90] eine Liste von Merkmalen, die für die Auswahl potentieller Applikationen relevant sind. Mit der Methode sollen Personen mit unterschiedlichem technischem Wissen über die Technologien im Bereich wissensbasierter Systeme in der Lage sein, die Anwendbarkeit einer Technologie auf ein spezielles Problem in einer spezifischen organisatorischen Umgebung qualitativ zu analysieren.

Die Definition und Ausführung der Validierungsaufgaben wird in der Literatur vielfältig beschrieben. Zum einen mag es daran liegen, daß dieser Forschungszweig noch recht jung ist und keine herausragend erfolgreiche Methode vorliegt, zum anderen daran, daß die Anforderungen an ein Expertensystem eine Formalisierung zur Zeit nicht möglich erscheinen lassen. Die nachfolgend erläuterten Ansätze stellen nur einen Ausschnitt in der Diskussion über Aufgaben und Methoden der Validierung von wissensbasierten Systemen dar. Es ist damit nicht eine einheitliche Klärung der Fragen bezüglich Definition und Ausführung von Validierungsaufgaben verbunden. Vielmehr bleibt die Frage nach geeigneten Methoden offen. Dies ist beabsichtigt, um den aktuellen Stand der Diskussion zu verdeutlichen.

Ayel und Laurent [AYE91] unterscheiden die beiden Prozesse Validierung und Verifikation von wissensbasierten Systemen nach den Personen, die sie durchführen. Dabei differenzieren sie zwischen dem Benutzer und dem Entwickler. Der Benutzer führt die Validierung durch und überprüft Verwendbarkeit, Fähigkeiten, Performanz und Zuverlässigkeit des Produkts. Der Validierungsprozeß bezieht sich direkt auf das Verhalten des Systems. Unter dem Begriff Entwickler werden verschiedene Arten von Spezialisten zusammengefaßt, z.B. der Programmierer, der Wissensingenieur und der Experte, die während der Entwicklungsphase kooperieren. Der Verifikationsprozeß betrifft den Code, das Wissen und Meta-Wissen. Der Entwickler überprüft syntaktische Eigenschaften, Kohärenz, Vollständigkeit, Korrektheit, Robustheit und Performanz. Testen ist eine Technik, die während des Validierungs- und Verifikationsprozesses eingesetzt wird. Mit dem Testen ist die Existenz von Testfällen verbunden und das Problem der Repräsentativität der Testfälle.

Ayel und Laurent [AYE91] unterscheiden auf einer weiteren Ebene zwischen der Validierung oder Verifikation von wissensbasierten Systemen (Untersuchung der Module und ihres Verhaltens) und der Validierung oder Verifikation von Wissensbasen.

O' Leary identifiziert in [OLE90] verschiedene Validierungsmethoden von Experten-Systemen. Sie umfassen das Testen mit realen Daten, Testen mit künstlichen Daten, direkte Untersuchung der Wissensbasis durch einen Experten oder einen Wissensingenieur, parallele Benutzung des Systems durch einen Experten und einen Nicht-Experten und Ereignistabellen. Die Testmethode mit realen Daten untersucht das System in Bezug auf Anforderungen, die denen in der Anwendung ähnlich sind. Das Testen mit künstlichen Daten konzentriert den Validierungsaufwand auf extreme Bereiche oder auf Bereiche von bestimmtem Interesse. Der Nachteil ist der zusätzliche Aufwand für die Entwicklung der Testdaten. Liegt das Wissen in lesbarer Form vor, kann es sinnvoll für den Experten oder den Wissensingenieur sein, die Untersuchung der Wissensbasis direkt vorzunehmen.

In [OLE87] benennt er potentielle Validierer: denselben Experten, von dem das Wissen gesammelt wurde, einen anderen Experten als den, von dem das Wissen stammt, den Wissensingenieur, den Endbenutzer, einen sogenannten Sponsoren des Systems oder einen unabhängigen Validierer. Diese Personen sind für die Validierung mehr oder weniger geeignet.

O' Leary diskutiert in [OLE91] Validierungsmethoden von Entwicklern von Experten-Systemen. Er stellt fest, daß die Effektivität des Validierungsprozesses und die Validierungstechnik stark korrelieren. 98

3.4 Validierung von wissensbasierten Systemen in der Medizin

Der Bereich der künstlichen Intelligenz in der Medizin hat sich in den letzten 20 Jahren als aktive und wachsende Disziplin hervorgehoben. Nach Clancey und Shortliffe [CLA84b] sind die Aufgaben der künstlichen Intelligenz in der Medizin die Erstellung von Diagnosen und

Therapievorschlägen. Im Gegensatz zu medizinischen Applikationen, die auf statistischen und probabilistischen Methoden basieren, sind symbolische Modelle von Krankheiten und ihren Beziehungen zu Patientenfaktoren und klinischen Manifestationen die Basis für die Programme im Bereich medizinischer künstlicher Intelligenz.

3.4.1 Initiale Projekte

Den Grundstein legten Systementwicklungen in den späten 60er und frühen 70er Jahren. Zu diesen Systemen zählen MYCIN, CASNET und INTERNIST. Sie stellen initiale Projekte dar, die viele andere Projekte zur Folge hatten, z.B. das Projekt ONCOCIN, das nach MYCIN von Shortliffe gestartet wurde, oder EXPERT und CADUCEUS, die direkt auf den Erfahrungen der Entwicklung von CASNET beziehungsweise INTERNIST aufbauten. Einen detaillierteren Überblick über medizinische Expertensysteme bietet Schill [SCH90]. Hier folgt im Vorgriff auf die Erläuterung von Validierungsansätzen eine kurze Beschreibung der Systeme MYCIN, CASNET, INTERNIST und ONCOCIN. MYCIN [SHO76] wurde an der Universität Stanford von Shortliffe, Davis und Buchanan entwickelt. Es dient zur Identifikation eines vorliegenden bakteriologischen Organismus und der Behandlung der durch Infektion daraus resultierenden Krankheit. Die Komponenten zur Wissensrepräsentation basieren auf Produktionsregeln und einer Faktensammlung. Die Komponenten zur Abarbeitung des Wissens sind eine Inferenzkomponente, die die Produktionsregeln hauptsächlich mittels "Backward chainingäbarbeitet und ein Mechanismus, der auf der Basis von Unsicherheiten schlußfolgert.

CASNET [KUL82] wurde 1970 an der Universität Rutgers von Kulikowski, Sholom und Weiss entwickelt. Es zieht diagnostische und therapeutische Schlußfolgerungen auf der Basis von physiologischen und funktionellen Zusammenhängen von Krankheiten im Bereich der Glaukomerkrankungen. Die Wissensrepräsentation basiert auf Frames. Sie stellen die kausal verbundenen pathophysiologischen Zustände dar. Die Pfade beschreiben Krankheitsprozesse. Der Übergang von einem pathophysiologischen Zustand zum nächsten wird durch ein Wahrscheinlichkeitsmaß bewertet.

ONCOCIN [SHO81] wurde an der Universität Stanford von Shortliffe, Scott, Bischoff, Campbell, van Melle und Jacobs entwickelt. Es unterstützt den Arzt bei der Planung einer ambulanten Krebstherapie und generiert Empfehlungen unter Berücksichtigung der Medikamentenwahl, Medikamentendosis, Labortests und in Abstimmung mit gleichzeitig laufenden Behandlungen. Die Behandlung basiert u.a. auf einem experimentellen Protokoll. Jedes Protokoll enthält detaillierte Hinweise für die Chemotherapie und die Radiotherapie. Medikamentenkombinationen werden über eine Reihe von Tagen angewendet. Die Wissensrepräsentation ist regelbasiert. Unsicherheiten werden nicht betrachtet.

INTERNIST [MIL82] wurde an der Universität Pittsburgh von Miller, Pople und Myers entwickelt und 1974 zum ersten Mal vorgestellt. Es behandelt den relativ großen Bereich der inneren Medizin. Frames bilden die Grundlage für die Wissensrepräsentation. Krankheits- und Manifestations-Entitäten stellen die beiden wesentlichen Einheiten der Wissensbasis

dar. Zwischen den beiden Einheiten existieren Relationen, denen eine Gewichtung angefügt ist. Sie ist ein Ausdruck für die Stärke der Verbindung.

3.4.2 Validierung und Verifikation: Modelle und Werkzeuge

Schill [SCH90] stellt fest, daß die meisten von ihr untersuchten medizinischen Expertensysteme bis auf wenige Ausnahmen heuristische Lösungsansätze verwenden. Dies hat den Nachteil, daß sie nicht auf einer formalen Ebene betrachtet werden können. Eine Beurteilung der Qualität kann nur im Vergleich mit Experten oder optimalen Verfahren erfolgen. Das Problem der Validierung ergibt sich nach Miller [MIL88] aus dem Mangel an objektiven *Gold Standards*. Es erwächst aus der Variation der Erfahrung, der subjektiven Beurteilung und der unverhohlenen Tatsache, daß Experten in ihren Meinungen voneinander abweichen, sogar in den ihrer Ansicht nach *besten* Lösungen eines Problems. Daraus resultiert die Schwierigkeit, einen angemessenen *Gold Standard* zu definieren, gegen den die Leistung des Expertensystems gemessen werden kann.

3.4.2.1 Verifikation gegen ein Paralleles Modell

Mars und Miller [MAR88] schlagen vor, daß ein Werkzeug zur Verfeinerung der Wissensbasis sein eigenes Wissen und seinen eigenen Schlußfolgerungsmechanismus besitzen soll. Das Werkzeug sollte ein *paralleles Modell* des Bereichs besitzen, mit dem die zu überprüfende Wissensbasis verglichen werden kann. Ohne das parallele Modell sind beispielsweise nur Konflikte oder Redundanzen aufdeckbar. Eine Einteilung der Typen paralleler Modelle nach Mars und Miller basiert auf den Wissensquellen des Modells:

Konstruktion durch das System selbst durch Inspektion der originalen Wissenbasis: Das System sucht nach Mustern in der Wissensbasis, von denen verallgemeinernde Hypothesen abgeleitet werden. Sie begründen Erwartungen über das, was die Wissensbasis enthalten sollte, und werden zur Verifikation bestehenden Wissens oder zur Akquisition neuen Wissens eingesetzt. Die abgeleiteten Hypothesen können in jedem Bereich automatisch gewonnen werden und erfordern keinen zusätzlichen Aufwand durch den Bereichsexperten. Es ist nicht ausgeschlossen, daß abgeleitete Verallgemeinerungen fehlerhaft sind und so zur Betrachtung von Modifikationen führen, die unsinnig für jemanden sind, der mit dem Bereich vertraut ist.

Konstruktion durch einen (menschlichen) Experten: Der Bereichsexperte definiert explizit ein paralleles Modell bestimmter Aspekte in einem Bereich. Die Aufgabe des Experten geht über die eigentliche Spezifizierung der Wissensbasis hinaus und erfordert deshalb einen zusätzlichen Zeitaufwand. Das Modell sollte jedoch im Gegensatz zum vorhergehenden Punkt keine offensichtlich unsinnigen Fehler enthalten.

Konstruktion durch das System selbst aus anderen Informationsquellen, beispielsweise anhand von Testdaten aus dem Bereich: Der Bereichsexperte beschäftigt sich hier nur mit der Spezifikation der Wissensbasis. Der Validierungsprozeß reflektiert aufgrund der Testdaten den speziellen Charakter des Bereichs.

3.4.2.2 Verifikation mit speziellen Werkzeugen

Zur Unterstützung der Verifikation wurden verschiedene den Expertensystemen zugeordnete Werkzeuge implementiert. Speziell die Werkzeuge für MYCIN, ONCOCIN und CASNET werden hier erläutert.

Verifikationswerkzeuge für MYCIN

MYCIN selbst enthält verschiedene Möglichkeiten, die Wissensbasis zu überprüfen. Der Einsatz eines parallelen Modells ist dafür nicht erforderlich:

1. Auf die Produktionsregeln des Bereichswissens kann interaktiv zugegriffen werden.
2. Während einer Konsultation sind Erklärungen zu den Schlußfolgerungen des Systems verfügbar.
3. Eine Reihe von Hilfsmitteln erlaubt die Analyse einer Anzahl von Testfällen mit bekanntem korrektem Ergebnis, um die Auswirkung von Änderungen an der Wissensbasis zu untersuchen.
4. Das System führt Statistiken über die Häufigkeit der Aktivierung von Regeln und die Resultate.

Zusätzlich zu diesen Möglichkeiten wurde zur weiteren Erleichterung der Wissensakquisition die Komponente TEIRESIAS [DAV79] entwickelt. Wenn die Diagnose von MYCIN nicht mit den Vorstellungen des medizinischen Experten übereinstimmt, führt TEIRESIAS durch die Regelverkettungen von MYCIN, so daß der Arzt den Fehler korrigieren kann. Neben der Möglichkeit, die Regeln in einem *Quasi-Englisch* zu formulieren, gruppiert TEIRESIAS die Regeln anhand struktureller Gemeinsamkeiten. Wird eine neue Regel hinzugefügt, verweist TEIRESIAS auf alle Prädikate in der korrespondierenden Regelgruppe, um zu verhindern, daß wesentliche Komponenten weggelassen werden.

Eine weitere Entwicklung ist EMYCIN, die die Eigenschaften von TEIRESIAS, den Inferenzmechanismus von MYCIN und weitere neue Eigenschaften besitzt. Zur Unterstützung einer begrenzten syntaktischen und semantischen Prüfung der Wissensbasis wird in EMYCIN jede neue Regel mit den bereits vorhandenen Regeln bezüglich Subsumierung und Kontradiktion verglichen. Bei Subsumierung kann die Schlußfolgerung der betrachteten

Regel von einer anderen Regel mit einer schwächeren Prämisse gezogen werden. Bei Kontradiktion erreicht die betrachtete Regel eine Schlußfolgerung, die sich von der Schlußfolgerung einer anderen Regel mit derselben Prämisse unterscheidet.

Verifikation von ONCOCIN

ONCOCIN integriert die Hilfsmittel zur Überprüfung der Regeln [SUW82]. Es werden alle Regeln gesammelt, die mit denselben Parametern folgern und die im selben Kontext anwendbar sind. Die Regeln einer zusammengefaßten Menge werden gegenseitig auf Subsumierung, Kontradiktion und Unvollständigkeit überprüft. Die Überprüfung basiert nur auf der Regelsyntax. Ein paralleles Modell wird nicht verwendet.

Nguyen et al. erweiterten die Möglichkeiten der Regelprüfung. Sie entwickelten das CHECK-System als Teil der kommerziell verfügbaren Expertensystem-Shell TIMM [NGU85], [NGU87]. Das Programm kombiniert logische Prinzipien, ebenso wie spezifische Informationen über den Wissensrepräsentationsformalismus. Es sucht in Ergänzung zu dem in ONCOCIN eingesetzten Testmodul fehlende Regeln, unerreichbare Klauseln und Sackgassen. Es führt keine syntaktische Regelprüfung durch und berücksichtigt keine Sicherheitsfaktoren. Als *Abfallprodukt* der Regelprüfung erzeugt CHECK ein Diagramm der Abhängigkeiten von Regeln.

Shwe et al. [SHW89] entwickelten ScriptGen, ein dynamisches Validierungssystem für ONCOCIN. Die vorhandenen statischen Methoden schienen ihnen nicht ausreichend, die Korrektheit des Systems zu demonstrieren. Die Standardmethode der Validierung erforderte bis dahin einen Wissensingenieur und einen konsultierenden Onkologen, um ein Testskript, die Eingabe von ONCOCIN, die aus zeitabhängigen Patientendaten besteht, manuell zu generieren. Der konsultierende Onkologe prüfte die Empfehlung von ONCOCIN. Bei Fehlern korrigierte der Wissensingenieur die Wissensbasis. ScriptGen automatisiert die Testdatengenerierung. Es benutzt ein paralleles Modell der ONCOCIN- Wissensbasis und eine eigene Inferenzmaschine zur Konstruktion von Testfällen. Um ein Skript zu generieren, schickt ScriptGen Zielschablonen, die vom konsultierenden Onkologen entworfen werden, an seine Inferenzmaschine. Das Ergebnis sind Skripte für die Analyse durch ONCOCIN.

Verifikation von EXPERT

Die Expertensystem-Shell EXPERT wurde aus dem Expertensystem CASNET abgeleitet in ähnlicher Weise wie EMYCIN aus MYCIN. SEEK [POL84], [WEI88] ist ein Verifikationswerkzeug, das zur Verfeinerung der Wissensbasen in EXPERT dient. Voraussetzung ist eine initiale Wissensbasis, die vom Wissensingenieur in den Produktionsregeln von EXPERT definiert ist. Von dieser Wissensbasis wird angenommen, daß sie grundsätzlich korrekt ist, jedoch verfeinert werden muß. SEEK verwendet zusätzliches Wissen, um anhand von Testfällen mit bekannten korrekten Schlußfolgerungen Vorschläge für mögliche Verfeinerungen der Wissensbasis zu generieren. Die Vorschläge beziehen sich auf einfache

Verallgemeinerungen oder Spezialisierungen der Regeln. Sie sind eingebunden in eine Menge bereichsunabhängiger heuristischer Regeln. Die Testfälle sind dagegen bereichsabhängig.

Die grundlegenden Prinzipien der Verfeinerung der Wissensbasis basieren auf der Repräsentation der Entscheidung in zwei Abschnitten: bedeutende und weniger bedeutende Beobachtungen, die für die Ableitung einer Diagnose relevant sind, eine Menge von Produktionsregeln zur Ableitung der Diagnose. Unsicherheit wird durch drei Ebenen beschrieben: possible, probable und definite. Bei der Verwendung von SEEK wird eine bestimmte Anzahl von Testfällen mit bekannter korrekter Diagnose analysiert. Werden Testfälle durch EXPERT falsch klassifiziert, kann SEEK Modifikation der aktuellen Kriterientabelle vorschlagen, z.B. erhöhe oder verringere die Anzahl der bedeutenden Kriterien oder der Kriterien mit untergeordneter Bedeutung. Für die Vorschläge zur Regelverfeinerung setzt SEEK Heuristiken im Zusammenhang von statistischem Verhalten und strukturellen Eigenschaften von Regeln und Klassen von Regelverfeinerungen ein. Die endgültige Entscheidung über die Akzeptanz oder das Verwerfen der Vorschläge liegt beim Wissensingenieur. Der Nachfolger von SEEK, SEEK2, besitzt Elemente zur Bewertung des aktuellen Modells anhand der Datenbasis mit Testfällen und dem dazugehörigen Expertenwissen. Es berechnet die Anzahl richtig positiver beziehungsweise falsch positiver Diagnosen des Modells oder die Häufigkeit, mit der eine Regel die Ursache einer falschen Diagnose ist. Wie die Statistiken zur Regelverfeinerung einzusetzen sind, basiert wiederum auf heuristischen Regeln.

3.5 Zusammenfassung

Es existiert zur Zeit kein Verfahren, das die Qualität eines wissensbasierten Systems vollständig erfaßt. Alle Verfahren aus der konventionellen Software-Entwicklung erfassen nicht die Eigenschaften von wissensbasierten Systemen, die sie von konventioneller Software unterscheiden. Anpassungen oder Erweiterungen resultieren bei wissensbasierten Systemen in qualitativen Verfahren, denen vereinzelt einfache Messungen aufgesetzt sind, z.B. das Zählen von korrekten und falschen Antworten. Eine wichtige Frage ist, wer darüber entscheidet, was korrekt und was falsch ist, wenn Antworten differenzierter sind, als eine binäre Entscheidung; denn hier fließt wiederum subjektives, nicht nachvollziehbares Wissen ein. Dabei sollte zumindest bei der Qualitätsbeschreibung aus der Sicht der Benutzer implizites Wissen so weit wie möglich reduziert werden.

Aus der Sicht der Entwicklung kann es nicht ausreichend sein, ein Bild von der Qualität eines komplexen Systems ausschließlich über die Befragung von Experten für ausgesuchte Fälle zu erhalten. Einerseits ist es schwierig, zwischen Experten unterschiedlicher Fachgebiete (z.B. Informatik und Medizin) so zu kommunizieren, daß die relevante Information zur Qualitätsverbesserung erarbeitet wird. Andererseits ist eine Analyse einer festen Anzahl von Fällen, die nur beschränkt auswählbar sind, zur Approximation der Vollständigkeit der Qualitätsbeschreibung ungeeignet. Diese Situation ist für die Analyse medizinischer Bilddaten typisch. Approximation impliziert Iteration. Es bedeutet, daß erst während der

Qualitätsprüfung deutlich werden kann, daß weitere Untersuchungen notwendig werden. Dafür ist die regelbare Dynamik der Größe der Testdatenmenge Voraussetzung.

Die vorgestellten Ansätze zur Lösung des Problems, die über die reine qualitative Beurteilung durch Experten hinausgehen und die Prüfung vagen und heuristischen Wissens mit berücksichtigen, sind

- teilweise Formalisierung und Expertenprüfung der nicht formalisierbaren Anteile [MES92]
- quantitative Methoden zur Qualitätsbeschreibung [OKE87]
- auf die Entwicklungszeit bezogen zeitlich früher Beginn der Qualitätssicherung durch Einsatz quantitativer Verfahren in SEEK [POL84], [WEI88]

Die Ansätze, sofern sie auf der Analyse von Testdaten basieren, berücksichtigen nicht die Zusammenstellung der Testfallmenge. Eine Strukturierung wie sie im Bereich konventioneller Software bei den Testmethoden vorhanden ist, z.B. durch Äquivalenzklassenbildung, hat für die wissensbasierten Systeme keine vergleichbare Bedeutung.

Kapitel 4

Ein Validierungsverfahren für die klassifizierende wissensbasierte Analyse von medizinischen Bilddaten

Die klassifizierende wissensbasierte Analyse von Bilddaten beinhaltet aus der Sicht der Validierung viele ungelöste (oder nur für spezielle Systeme gelöste) Probleme. Ein Beweis der Korrektheit ist (zur Zeit) ausgeschlossen. Der Grad der Approximation von Korrektheit ist in Bezug auf das im System eingesetzte unsichere und heuristische Wissen schwer zu beschreiben. Neben dem einfachen Anspruch, simple Programmierfehler zu eliminieren, ist auch die Qualität der Expertise für die Validierung von Bedeutung. Um sich ein Bild davon zu verschaffen, bedarf es einer Testfallmenge, die bestimmten Anforderungen zu genügen hat. Eine besondere Schwierigkeit bildet die Variabilität von Bilddaten, die explosionsartig die Anzahl möglicher Eingaben erhöht und damit die Gefahr einer hohen Restfehlerwahrscheinlichkeit, da der Anteil getesteter Fälle relativ zur Gesamtmenge im allgemeinen klein ist.

Ein Validierungsverfahren für ein wissensbasiertes Bildanalysesystem muß sinnvolle Hinweise zur Qualität für die Entwickler und außerhalb der Entwicklung stehende Benutzer liefern. Für die Entwickler, um Fehler zu eliminieren und die Expertise zu verbessern und für Benutzer, um das System für den Einsatz beurteilen zu können.

Das Validierungsverfahren für wissensbasierte Systeme, das hier vorgestellt wird, soll neben der Detektion von konventionellen Fehlern zusätzlich unzureichend modelliertes unsicheres und heuristisches Wissen, die Expertise, erfassen. Dazu wird ein quantitativer Ansatz erarbeitet. Er verfolgt eine Teststrategie, die auf der strukturierten Betrachtung des Eingabebereichs beruht. Ein weiterer Schwerpunkt liegt in der Definition einer Methode zur Generierung von Testfällen, die die Validierung von den Beschränkungen originaler

Daten befreit.

Obwohl eine formale Spezifikation nicht vorhanden ist, wird davon ausgegangen, daß die angestrebte Leistung des Systems in einer informellen Spezifikation beschrieben werden kann. Sie soll so umfassend sein, daß die Ergebnisse der Tests nach dem entwickelten Verfahren gegen die Spezifikation oder Informationen, die aus der Spezifikation ableitbar sind, überprüft werden können.

Die Teststrategie basiert auf

- der quantitativen Strukturierung des Eingabebereichs der Wissensbasis durch die Einführung eines Merkmalsraums
- der Erhaltung der Struktur durch die Analyse
- der Musterbildung durch die Analyseergebnisse unter Bezug auf die Struktur des Eingabebereichs

Diagnosen entstehen auf der Basis verschiedener Informationen, z.B. Laborwerten, Funktionsmessungen (Puls), Zustandsmessungen (Fieber, Blutdruck) usw. Betrachtet man die quantifizierbaren Informationen, die im folgenden als Merkmale bezeichnet werden, sind bestimmte Werteintervalle der Merkmale typisch für bestimmte Diagnosen. Faßt man die Diagnosen unter dem Gesichtspunkt, gleiche Merkmale zu besitzen, zusammen, und ordnet sie analog dem zugehörigen Werteintervall eines gemeinsamen Merkmals an, dann ergibt sich ein Diagnoseband, wie es in Abbildung 4.1 zu sehen ist.

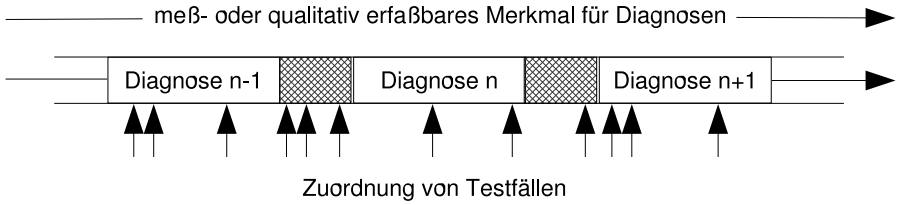

Abbildung 4.1: Anordnung von Diagnosen in Bezug auf ein Merkmal, das für die Diagnosen relevant ist, und Zuweisung von Testfällen zu den Diagnosen in Abhängigkeit von den Merkmalen. Patientendaten, die Testfälle darstellen, verteilen sich ungleichmäßig über das Diagnoseband. Die schraffierten Bereiche stellen die Übergangszonen zwischen Diagnosen dar, in denen eine eindeutige Zuordnung zu ausschließlich einer Diagnose nicht möglich ist.

Eine Untersuchung eines Patienten weist auf einen Punkt des Bandes, da die Referenz auf ein Merkmal eindeutig ist (es gilt nur ein Merkmalswert). Die Verteilung der Fälle auf dem Diagnoseband erscheint zunächst unstrukturiert. In Bezug auf die Merkmale wird eine Struktur deutlich. Diese Struktur wird durch die Analyse nicht verändert. Ist sie bekannt, können Aussagen über die Ergebnisse getroffen werden. Ist die Struktur z.B. die Reihenfolge des Auftretens der Fälle in Bezug auf ein diagnoserelevantes Merkmal, wirkt

sie sich auf die Ergebnisse aus, indem durch das wissensbasierte System die Reihenfolge der im Diagnoseband auftretenden Diagnosen analog der Reihenfolge der Fälle einzuhalten ist (Abbildung 4.3).

Voraussetzung für die Ausnutzung von Strukturen durch das Validierungsverfahren sind Eigenschaften von Eingabedaten (Testfällen), die alle dieselben Merkmale besitzen:

1. Ähnlichkeit von Testfällen: Testfälle sind sich ähnlich, wenn die Differenz der Werte ihrer Merkmale einen bestimmten Schwellwert unterschreitet. Beeinflussen Merkmale der gemeinsamen Merkmalsmenge, in denen sich die Testfälle ähnlich sind, die Entscheidungsfindung, sind auch die Analyseergebnisse ähnlich.

2. Anordnung von Testfällen: Testfälle können anhand ausgewählter Merkmale angeordnet werden (Abbildung 4.2). Unterscheiden sie sich beispielsweise in einem Merkmal und in den übrigen Merkmalen nicht, ist die Anordnung durch den aufsteigenden bzw. absteigenden Wert des Merkmals gegeben.

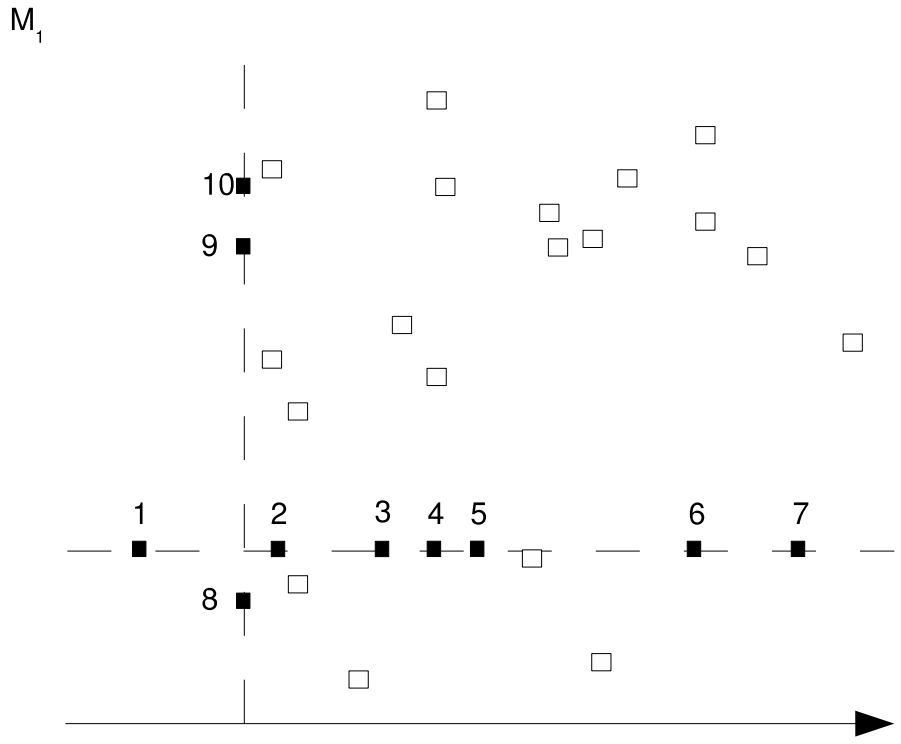

Abbildung 4.2: Verteilung von Testfällen in dem durch M_1 und M_2 aufgespannten 2-dimensionalen Merkmalsraum. Die Testfälle verteilen sich ungleichmäßig im Raum bis auf die Fälle 1-10. Die Fälle 1-7 unterscheiden sich nur in M_2, die Fälle 8-10 in M_1

3. Spezifikation eines Ergebnismusters angeordneter Testfälle: Die Analyseergebnisse angeordneter Testfälle bilden in Bezug auf die Anordnung ein spezifizierbares Muster (Abbildung 4.3).

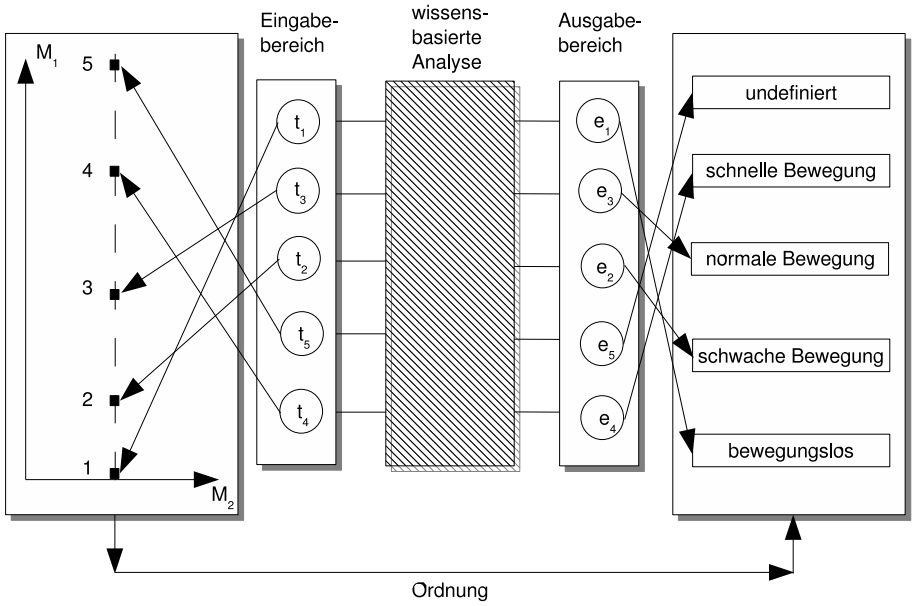

Abbildung 4.3: Übersicht über die Zusammenhänge zwischen dem Merkmalsraum, Ein- und Ausgabebereich, einer Wissensbasis, die Bewegungen analysiert und Analyseergebnissen. Die Testfälle $t_i, i = 1, .., 5$ sind im Eingabebereich ungeordnet. Für alle Testfälle ist der Wert von M_2 gleich. Sie werden über das Merkmal M_1 angeordnet, z.B M_1 = Geschwindigkeit. Die zugehörigen Analyseergebnisse e_i bilden, wenn man sie in der gleichen Ordnung betrachtet, ein bestimmbares Muster

Das Objekt der Validierung ist eine klassifizierende Wissensbasis. Der Eingabebereich der Wissensbasis setzt sich aus Repräsentationen von segmentierten Objekten des Bildes zusammen. Aus diesen Eigenschaften wird eine objektorientierte Betrachtung des Ein- und Ausgabebereichs abgeleitet. Die Objekte korrespondieren mit Klassen bzw. Objekten in der Wissensbasis. Im Eingabebereich sind es die Klassen, die die Schnittstelle bilden, im Ausgabebereich die Klassen, die das Ergebnis repräsentieren. Die Objekte im Eingabebereich werden durch Testfälle instanziiert, im Ausgabebereich durch die konkreten Analyseergebnisse.

Zunächst werden zu dem Validierungsverfahren grundlegende Begriffe in Kapitel 4.1 und formale Grundlagen in Kapitel 4.2 definiert. Aspekte der Validierung, die sich auf Eigenschaften der Eingabedaten und Ergebnisklassen beziehen, werden in Kapitel 4.3 diskutiert. Die Ausnutzung von Merkmalen bei der Verteilung von Ergebnissen ist die Basis

für die Definition der Systematik der Testfallgenerierung in Kapitel 4.4. Die Ergebnismuster liefern Hinweise auf Fehler (Kapitel 4.5), die in Abhängigkeit von der Struktur der Wissensbasis lokalisiert werden (Kapitel 4.6). Die Ergebnisse der wissensbasierten Bildanalyse sind Instanzen von Klassen der Wissensbasis, wobei der Sicherheitswert einen Teil der Instanz darstellt. Eingebettet in das Ergebnismuster liefern quantitative Verfahren der Qualitätsbeschreibung (Kapitel 4.7) numerische Informationsträger für eine Bewertung der Qualität der Analyse.

4.1 Grundlegende Definitionen

Mit Hilfe der Validierung sollen Aussagen über die Qualität der Ergebnisse der wissensbasierten Analyse möglich werden. Das Verfahren soll einerseits dem Entwickler Hinweise auf Fehler liefern, andererseits Qualitätsbeschreibungen unterstützen, die sowohl dem Entwickler als auch dem Anwender helfen, die Fähigkeiten oder die kritischen Bereiche des Systems abzuschätzen. Um dieses Ziel zu erreichen, wird überprüft, ob das Klassifikationsverhalten der wissensbasierten Analyse den in der Spezifikation festgelegten Bedingungen genügt. Die Validierung ist praktisch die Kontrolle der Übereinstimmung zwischen spezifizierter und implementierter Wissensbasis. Sie umfaßt die Methoden, die notwendig sind, um die Anforderungen, die mit der Validierung verknüpft sind, zu erfüllen.

In der Bildanalyse werden die Objekte der im Bild festgehaltenen Szene bezüglich bestimmter Eigenschaften beurteilt. Die Beurteilungen können im klassifizierenden System ebenfalls als Objekte aufgefaßt werden. Die Objekte entsprechen den Klassen und werden in der intensionalen Repräsentation durch die generischen Strukturen erfaßt. Betrachtet man das klassifizierende wissensbasierte System als abgeschlossenes System, das Objekte aus einem definierten Eingabebereich auf Objekte eines definierten Ausgabebereichs abbildet, dann wird die funktionale Beschreibung des Systems als Spezifikation aufgefaßt. Sie beinhaltet die vollständige Definition der Objekte des Ein- und Ausgabebereichs und der funktionalen Beziehung zwischen dem Ein- und Ausgabebereich. Die Spezifikation ist hier nicht formal. Sie beinhaltet eine qualitative objektbezogene Beschreibung der inhaltlichen Leistungsanforderungen an das wissensbasierte System.

Die operationale Beschreibung wird durch die Wissensbasis des Systems realisiert. Für die Validierung wird der Black-Box-Ansatz gewählt. Die Spezifikation beinhaltet bezüglich des Ein- und Ausgabebereichs intensionale Beschreibungen der Objekte. Die Elemente der extensionalen Menge werden als Ausprägungen bezeichnet.

In einem wissensbasierten System nach Niemann [NIE87] ist die Wissensbasis eine Komponente von mehreren modular angeordneten. Ein im Auswertungsverfahren vorangestelltes Methodenmodul extrahiert die für die Analyse relevanten Objekte. Es ist unabhängig von der wissensbasierten Analyse, da es unbewertete Informationen über die Objekte zur Verfügung stellt. Fehlerhafte Ergebnisse der Methoden werden in der wissensbasierten Auswertung entsprechend behandelt. Die Validierung des Methodenmoduls

erfolgt unter veränderten Bedingungen im Vergleich zur wissensbasierten Analyse und wird deshalb gesondert betrachtet. Die Steuerung der Analyse übernimmt das Kontrollmodul. Es ist unabhängig von der Wissensbasis und wird deshalb bei der Validierung nicht betrachtet. Die Daten, die zur Steuerung der Analyse an die Kontrolle zu übergeben sind, werden durch deklarative und prozedurale Anteile der Wissensbasis festgelegt und somit von der Validierung erfaßt. Datenbank- oder Wissenserwerbskomponenten werden hier nicht betrachtet.

4.2 Formale Grundlagen

Mit der Diskretisierung der gemessenen Information durch Pixelzuordnung in der Bildmatrix sind alle weiteren *Informationsextrakte* aus den Bilddaten ebenso diskret. Eingabedaten in die Wissensbasis sind objektbezogene Informationen aus dem Bild, z.B. Konturen oder Masken. Der Eingabebereich sowie der Ausgabebereich der Wissensbasis werden als geschlossene Menge von Objekten aufgefaßt, die in einem durch die Wissensbasis realisierten funktionalen Zusammenhang stehen. Den Objekten werden bei der Analyse eines Falles Werte zugeordnet, die allgemein als Ausprägung eines Objektes bezeichnet werden. Die Menge aller Ausprägungen der Objekte des Eingabebereichs wird durch die Bilddimension und die Bitanzahl für die Informationscodierung beschränkt. Die funktionale Beziehung ist vollständig auf den Objekten des Eingabebereichs definiert und bildet auf den Ausgabebereich ab.

Sei $S(w) = S = \{S_E, S_A, F\}$ die Spezifikation der Wissensbasis w mit dem Eingabebereich S_E, dem Ausgabebereich S_A und der funktionalen Abbildung F. Die zugehörigen Objektmengen sind

$S_E = \{E_1, E_2, \ldots, E_N\}$ mit N Objekten des Eingabebereichs
$S_A = \{A_1, A_2, \ldots, A_M\}$ mit M Objekten des Ausgabebereichs

$E_n, n = 1, ..., N$ und $A_m, m = 1, ..., M$ bezeichnen die Mengen aller möglichen Ausprägungen eines Objektes des Eingabebereichs bzw. eines Objektes des Ausgabebereichs. Ein Objekt des Eingabebereichs wird als E-Objekt, eine Ausprägung des Eingabebereichs als E-Ausprägung, ein Objekt des Ausgabebereichs als A-Objekt und eine Ausprägung des Ausgabebereichs als A-Ausprägung bezeichnet.

Den Objekten werden in einer konkreten Analyse Ausprägungen zum aktuellen Fall zugeordnet. Die Anzahl aller möglichen Ausprägungen und Kombinationen von Ausprägungen wird durch die funktionale Beziehung beschränkt. Angenommen, F ist für jede E-Ausprägung definiert, dann wird die Anzahl differenzierbarer A-Ausprägungen umso mehr reduziert, je weiter F *Information* reduziert. Zwei Analysen unterscheiden sich, wenn sie sich in ihren E- und/oder A-Ausprägungen unterscheiden. Zählt man die E- und A-Ausprägungen der Objekte durch, dann ist

e_n^p, p-te E-Ausprägung des n-ten Objektes
a_m^q, q-te A-Ausprägung des m-ten Objektes

Sei P_n die Anzahl aller E-Ausprägungen des n-ten E-Objekts und Q_m die Anzahl aller A-Ausprägungen des m-ten A-Objekts. Die Mengen der Ausprägungen eines Objektes sind dann beschrieben durch

$$E_n^p = \{e_n | 1 \leq p \leq P_n\}, 1 \leq n \leq N \tag{4.1}$$

$$A_m^q = \{a_m | 1 \leq q \leq Q_m\}, 1 \leq m \leq M \tag{4.2}$$

Die funktionale Bezeichnung $F : S_E \rightarrow S_A$ beschreibt die Abbildung der Ausprägungen der Objekte des Eingabebereichs auf die Objekte des Ausgabebereichs. Ordnet man die E- und A-Ausprägungen in einem Vektor an, dann ist eine Analyse beschrieben durch

$$F(e_1^{p^1}, e_2^{p^2}, \ldots, e_N^{P^N}) = (a_1^{q^1}, a_2^{q^2}, \ldots, a_M^{q^M}) \tag{4.3}$$
$$\tag{4.4}$$

Eingabebereich

Die im Eingabebereich zur wissensbasierten Bildanalyse vorliegenden Ergebnisse der Bildvorverarbeitung können mit einem einfachen Symbol verknüpft sein, z.B. der Benennung. Sie stellen aber im Prinzip hinsichtlich der angestrebten Aussage interpretationsfreie numerische Daten dar. Aus diesen Daten können zunächst weitere eindeutige, objektive (interpretationsfreie) numerische Parameter abgeleitet werden. Diese Parameter beeinflussen die Problemlösung durch die Wissensbasis direkt oder indirekt oder fließen nicht ein. Sie liefern jedoch unabhängig von der Problemlösung ein numerisches Merkmal, das für einzelne Ausprägungen oder Zusammenhänge zwischen Ausprägungen typisch ist.

Das Merkmal ist auf der Menge $\mathbb{R}$ der reellen Zahlen definiert. Sei $G = \{G_h | 1 \leq h \leq H_E\}$ die Menge der Funktionen, die die Abbildungen der E-Objekte aus der Menge $E = \{E_n | 1 \leq n \leq N\}$ auf die Mermale der Menge $K = \{K_h | 1 \leq h \leq H_E\}$ beschreiben. Der Definitionsbereich einer Abbildung G_h ist die Teilmenge E_h von E und damit die zu E_h gehörenden E-Ausprägungen. Für eine Funktion in G gilt

$$G_h : E_h \rightarrow K_h, \qquad G_h = G_h(E_h) = \mu_h \in \mathbb{R} \tag{4.5}$$

Die Menge der Merkmale spannt einen H_E-dimensionalen Merkmalsraum K auf. Jedes Merkmal K_h korrespondiert mit einer Abbildungsvorschrift G_h.

Die Elemente des Merkmalsraums lassen sich anordnen. Eine Vorschrift zur eindimensionalen Anordnung der Teilmenge der E-Ausprägungen der E-Objekte, die sich nur in einem Vektorelement unterscheiden, ist die Sortierung nach aufsteigendem Wert dieses Vektorelements.

$$k_\mu = (\mu_1, \mu_2, \ldots, \mu_H), k_\nu = (\nu_1, \nu_2, \ldots, \nu_H), \mu_h \neq \nu_h \tag{4.6}$$

sind Merkmalsvektoren der E-Ausprägungen der μ-ten und ν-ten Analyse, die sich im h-ten Merkmal unterscheiden. Sei $\mu_h < \nu_h$, dann steht der Vektor der μ-ten Analyse nach der Anordnungsvorschrift vor dem Vektor der ν-ten Analyse. Die Richtung der Anordnung wird durch den Pfeil $\rightarrow$ ausgedrückt mit der Bedeutung Vorgänger $\rightarrow$ Nachfolger. Für das Beispiel gilt

$$k_\mu \rightarrow k_\nu \Rightarrow e_n^\mu \rightarrow e_n^\nu \tag{4.7}$$

Die eindimensionale Anordnung kann ebenso auf die Sortierung auf der Basis mehrerer Merkmale zurückgeführt werden. Über die funktionalen Beziehungen G ist die Anordnung auf die E-Ausprägungen einer Analyse projizierbar. Damit lassen sich die zugehörigen konkreten Fälle in Form der E-Ausprägungen in gleicher Weise anordnen

$$\{e_n^\mu\} \rightarrow \{e_n^\nu\} \tag{4.8}$$

Ausgabebereich

Die Objekte des Ausgabebereichs besitzen eine Ergebnisstruktur, deren Elemente als Merkmale aufgefaßt werden. Sie manifestieren sich bei einer Analyse aus der Sicht des Ausgabebereichs als A-Ausprägungen. Die A-Ausprägungen besitzen dieselbe Ergebnisstruktur. Die Elemente sind hier die Merkmalswerte. Anders als im Eingabebereich erhält man hier nicht die Merkmale über Abbildungsvorschriften in einen Merkmalsraum, sondern durch Extraktion aus der Ergebnisstruktur. Das Ergebnis selbst oder ein Teil davon ist von Interesse. Die zu extrahierenden Teilstrukturen, die die Merkmale des Ausgabebereichs darstellen, und ihre Manifestierung sind zu benennen. Um zwischen der Abbildung auf Merkmale im Eingabebereich und der Extraktion von Merkmalen im Ausgabebereich zu unterscheiden, wird folgende *Punkt*-Notation eingeführt. Der Wert eines Merkmals K der A-Ausprägung a_m^μ des A-Objektes A_m wird notiert mit

$$A_m \cdot K(a_m^\mu) = k_m \tag{4.9}$$

Ein Merkmal eines Objekts des Ausgabebereichs, das eine Aussage repräsentiert, kann beispielsweise eine Gewichtung dieser Aussage sein. Als Beispiel für eine Manifestation der

A-Ausprägung bezüglich des Merkmals käme ein Wert aus dem Intervall $[0, 100]$ in Frage, der eine Übereinstimmung des aktuellen Falles mit der durch das Objekt repräsentierten Aussage prozentual ausdrückt.

Die funktionale Beziehung F der Spezifikation impliziert durch die Anordnung der Fälle anhand ihrer E-Ausprägungen die gleiche Reihenfolge für die A-Ausprägungen der A-Objekte und somit ihrer Merkmale. Die Muster der Merkmalswerte, die sich aus der Anordnung ergeben, müssen nicht übereinstimmen. Vielmehr ergibt sich aus F auf der Basis des Wertemusters der Merkmale angeordneter Eingabedaten das Wertemuster der Merkmale der Ausgabedaten. Es gilt ebenfalls, daß eine Anordnung auf der Basis von Merkmalen, die weder direkt noch indirekt durch F erfaßt werden, für die Validierung irrelevant ist.

Die Spezifikation enthält die generische Beschreibung der Objekte des Ein- und Ausgabebereichs, die den Definitionsbereich der Ausprägungen darstellen. Auf der Basis der Informationen aus der Spezifikation können die Merkmalsfunktionen definiert werden. Sind in der Spezifikation nicht alle relevanten Merkmale vollständig explizit notiert, ist der Einfluß implizit gegebener Merkmale auf die Analyse durch die Wissensbasis w nur eingeschränkt zu beurteilen.

Die Validierung nutzt die Hinweise aus, die mit der Anordnung der E-Ausprägungen und dem durch F implizierten Muster der geordneten A-Ausprägungen vorliegt. Die Realisierung des spezifizierten Programms muß die Muster im Ausgabebereich erhalten. Ergibt sich in der implementierten Version eine Abweichung, liegt die Ursache in der operationalen Beschreibung, also in der Methode selbst oder im implementierten Sourcecode.

Eine Grundlage für eine optimale Validierung ist eine Spezifikation, die trotz des qualitativen Charakters die Informationen vollständig explizit zur Verfügung stellt. Die Probleme stellen sich weniger in der Beschreibung der Ein- und Ausgabebereiche als in der funktionalen Beziehung. Die Elemente aus S_E und S_A sind eindeutig definierbare Objekte mit eindeutigen Definitionsbereichen. Ihre Bedeutung erhalten sie durch F.

Die Definition der funktionalen Beziehung F ist umso schlechter, je weniger Information direkt für die Durchführung der Validierung daraus abgeleitet werden kann. Hier ist es die Aufdeckung der Merkmale, die die Analyse beeinflussen und die Art der Beeinflussung der Analyse, die für eine unabhängige Validierung deutlich zu benennen sind. Sind nicht alle Merkmale explizit, kann die Validierung für die unerkannten Merkmale keine Aussagen über die Analyse treffen. Versucht man andererseits neben den spezifizierten Merkmalen möglichst viele Merkmale unabhängig von F zu benennen und zu untersuchen, ist ein Teil des Aufwandes für die Validierung überflüssig.

4.3 Allgemeine Aspekte der Validierung

Das Validierungsverfahren umfaßt die Teststrategie, den Aufbau der Testdatenmenge und die Auswertung der Testergebnisse. Mit der Teststrategie wird die Testmethode festge-

legt, Teilziele definiert, die die Überprüfung von unterschiedlichen Analysen betreffen und Testfälle selektiert, um die Teilziele zu erreichen. Die Summe der Teilziele liefert die notwendigen Informationen für Aussagen zur Qualität der Wissensbasis des wissensbasierten Bildanalysesystems. Mittels vorgestellter Verfahren werden Informationen extrahiert, die einerseits den Entwicklern zur Qualitätsverbesserung dienen, andererseits den Anwendern die Basis für Aussagen zur Qualität liefern, die Vergleiche mit alternativen Verfahren unterstützen. Die Testdatenmenge besteht nicht nur aus geeigneten originalen Patientendaten. Hinsichtlich der Anforderungen für die Sicherstellung hoher Qualitätsansprüche ist die Generierung künstlicher Daten notwendig, um die Bereiche abzudecken, für die originale Daten nicht zur Verfügung stehen.

4.3.1 Auswahl der prinzipiellen Teststrategie

Prinzipiell bieten sich zwei Ansätze zum Test der Wissensbasis an: der White-Box- und der Black-Box-Ansatz. Beim White-Box-Ansatz sind die internen Strukturen der Wissensbasis und die dynamischen Abläufe während einer Analyse im Detail zu betrachten. Die bei einem Programmlauf abzuarbeitenden Anweisungen sind für möglichst viele Kombinationen zu prüfen. Der Black-Box-Ansatz betrachtet das Testobjekt als abgeschlossene Einheit, die ein Eingabedatum auf ein Ausgabedatum abbildet. Weicht das Ergebnis von dem erwarteten ab, ist das ein Hinweis auf einen Fehler im System. Hinweise auf die Ursache können im allgemeinen daraus nicht abgeleitet werden.

Bei den komplexen Strukturen einer Wissensbasis, die aus deklarativen und prozeduralen Elementen besteht, den unterschiedlichen Arten von modelliertem Wissen (algorithmisch, heuristisch, vage) und der Intention, daß das System ähnlich einem menschlichen Experten Informationen beurteilen soll, ist eine Strategie erforderlich, die über die Prüfung von Programmcode hinausgeht. Die Komplexität der Semantik der produzierten Ausgabedaten geht bei einem wissensbasierten System über die Semantik konventioneller Software hinaus. Zusätzlich zur Kontrolle von Abläufen, die auf eine denotationelle Semantik zurückführbar sind, stellt sich bei einem wissensbasierten System das Problem der Kontrolle der vom System generierten, nicht formalisierbaren subjektiven Beurteilung. Als Beispiel soll der Vergleich eines konventionellen Bildauswertungs-programms mit einem wissensbasierten Bildanalysesystem das Problem verdeutlichen. Das konventionelle Programm liefert bedeutungsunabhängige numerische Parameter, die Eigenschaften der Objekte im Bild beschreiben. Betrachtet man die Projektion eines geschlossenen Objektes im Bild, dann sind die Fläche oder der Schwerpunkt des Objekts derartige numerische Parameter. Sie stellen eine aus der reinen Grauwertdarstellung aufbereitete objektive Information dar, die für eine nachfolgende Beurteilung eine Grundlage bilden kann.

Im Vergleich zur numerischen Bildauswertung liefert das wissensbasierte Bildanalysesystem zusätzlich eine Beurteilung der numerischen Parameter. Wäre die Fläche des Objektes ein Eingangsparameter, so könnte eine Beurteilung sein *die Fläche des Objektes ist zu groß*. Bei schrittweiser Überprüfung der konventionellen Auswertungssoftware oder des

Ergebnisses läßt sich die Korrektheit des Programms oder Ergebnisses für diesen Testfall zeigen. Bei gleicher Vorgehensweise beim wissensbasierten Bildanalysesystem kann ebenso gezeigt werden, daß die Zwischen- und Endzustände innerhalb der zulässigen Wertebereiche liegen, und weitere Aspekte ähnlich wie bei konventioneller Software erfüllt sind. Ob das Ergebnis der wissensbasierten Analyse einer korrekten Expertenbeurteilung entspricht, ist mit der lokalen Sicht auf der Ebene der schrittweisen Prüfung der Programmverläufe oder eines einzelnen Ergebnisses nicht aufzuzeigen.

Für die Validierung ergeben sich zwei Aspekte der Betrachtung von Wissen, die bei der Validierung zu berücksichtigen sind

- die Methodik (z.B. deklaratives, prozedurales Wissensmodell)
- die Expertise (z.B. Heuristik, vages Wissen)

Unter dem Aspekt der Methodik ist die Wissensbasis ähnlich der konventionellen Software. Eine Qualitätsprüfung einer implizit repräsentierten, subjektiven Beurteilung stellt sich hier nicht. Der Aspekt der Expertise liefert die Unterscheidung zwischen konventioneller und wissensbasierter Software. Im Validierungsverfahren müssen hier durch Ergänzung oder Adaption der Testverfahren bei konventioneller Software die besonderen Anforderungen Berücksichtigung finden.

Die Methodik umfaßt die allgemein gültigen Erkenntnisse im jeweiligen Fachgebiet. Sie basiert auf naturwissenschaftlichen Erkenntnissen beispielsweise über Lage, Anordnung oder Funktion von Organen und dem Wissen über die Zuordnung von Erscheinungen zu Diagnosen. Dem prozeduralen Wissensmodell liegen Methoden zugrunde, die objektiv überprüfbar sind, ohne subjektives (Experten-)Wissen (z.B. Erfahrungswerte für quantitative Entscheidungsschwellen) einfließen zu lassen. Die Terminierung der Analyse für die spezifizierten Eingaben innerhalb zulässiger Wertebereiche zu prüfen, gehört zu den Validierungsaufgaben auf dieser Ebene. Ob die Definition der statischen und prozeduralen Anteile den spezifizierten Experten adäquat repräsentiert, bleibt hier unberücksichtigt.

Die Expertise spiegelt sich in den Wissensanteilen wider, die für Entscheidungen bei einer Analyse herangezogen werden. Sie liefern während der Inferenz beispielsweise die Kosten, die über die Pfadauswahl entscheiden, Daten, die für die Transformation von numerischer in symbolische Darstellung benötigt werden oder die Übereinstimmung eines aktuellen Falles mit der generischen Klasse bewerten.

Bei der Expertise ist aufbauend auf die Prüfung der Methodik zu untersuchen, ob die prozedurale oder deklarative Repräsentation der Methode und die Parameter der Entscheidungsfindung für die Problemlösung adäquat sind. Der Unterschied zu den Prüfungen der Methodik ist, daß hier, vorausgesetzt die Methode ist vollständig und zuverlässig, festzustellen ist, ob sie als ein Teil eines Systems, das einen Experten simulieren soll, diese Simulation geeignet unterstützt. Es ist nicht zu untersuchen, ob die Methoden die Art und Weise der Schlußfolgerungen eines Experten wiedergeben, sondern, ob die Wissensbasis ein Ergebnis liefert, das der Beurteilung des Experten entspricht. Während die Methoden die Strategie

der Interpretation tragen, dient die Einstellung von Parametern bei Kostenfunktionen oder Funktionen zur Beschreibung vagen Wissens der Quantisierung der subjektiven Heuristik des Experten. Bezogen auf das Beispiel eines einfachen Bildanalysesystems, das Objekte im Bild mit dem *Split and Merge* - Verfahren nach Pavlidis [PAV77] segmentiert und benennt, ist die Methodik die Auswahl und Anordnung der Operatoren und Verfahren der Bildverarbeitung, z.B. neben dem *Split and Merge* - Verfahren zur Glättung oder Hervorhebung von Grauwertbereichen. Die Expertise ist implizit in der Definition des Prädikates zur Regionenbildung beim Teilen und Zusammenfügen. Ist das Prädikat ein Schwellwert, dann ist die Expertise umso besser, je genauer mit dem Schwellwert die Objekte segmentiert werden.

Durch einen White-Box-Ansatz ist die Methodik einer komplexen Wissensbasis mit einem vertretbaren Zeitaufwand nicht ausreichend überprüfbar. Die Expertise mit diesem Ansatz zu beurteilen, erscheint weit schwieriger. Als Kritikpunkt ist zu sehen, ob bei einer induktiven lokalen Sicht auf Zustände und Zustandsänderungen die globale Qualität der Expertise, die auf das Zusammenwirken der Teile des Gesamtsystems zurückzuführen ist, ausreichend erfaßbar ist. Erst das Endergebnis spiegelt alle lokalen Wirkungen der bei der Analyse aktivierten Wissensarten wider. Die ausgewählte Teststrategie verwendet aus diesen Gründen den Black-Box-Ansatz.

Die operationale Beschreibung der Wissensbasis, beziehungsweise die Implementation, kann sich auf der Basis einer nicht formalisierbaren Spezifikation nur einer idealen Vorstellung über das reale System annähern. Die Symptome reichen von einer geringen Abweichung des Ergebnisses vom erwarteten Ergebnis bis zum Systemabsturz. Die Ursachen für die Abweichung vom idealen System sind sowohl in der Methodik als auch in der Expertise zu suchen. Sie werden im folgenden allgemein als Fehler bezeichnet und weiter als Defekte, Schwächen und Unwissen spezifiziert:

Definition 4.1 *Ein Defekt ist eine Anweisung im Quelltext, die zu unerlaubten Zugriffen auf Speicherbereiche führt.*

Definition 4.2 *Eine Schwäche ist auf Elemente einer Teilmenge des Eingabebereichs oder daraus abgeleiteter Daten eines Zwischenzustands der Analyse zurückzuführen, die falsch oder nicht explizit in den Prozeduren oder deklarativen Beschreibungen zur Bestimmung des nachfolgenden Zwischen- oder Endzustands der Analyse behandelt werden.*

Definition 4.3 *Unwissen ist falsche, unvollständige oder redundante Expertise.*

Ein {underlineDefekt wird mit den verfügbaren Compilern (für prozedurales und deklaratives Wissen) nicht erkannt. Es handelt sich dabei um Fehler, die beispielsweise zu einem Überlauf des allokierten Speichers durch ein falsches Abbruchkriterium bei einer inkrementellen Indizierung führen. Ein Defekt kann im Extremfall zum Absturz des Analyselaufs führen.

Eine Schwäche liegt vor, wenn Strukturen und Methoden des Gesamt- oder Teilsystems in Bezug auf den jeweiligen Wertebereich der Eingabeparameter unvollständig definiert sind. Die Unvollständigkeit kann bereits in der Planung vorliegen. Sie wird spätestens mit der Implementierung realisiert. Eine Schwäche kann zu einem Defekt führen. Andererseits ist die defektfreie Terminierung ihrer Verarbeitung durch gewollte oder zufällige Definitionen möglich, beispielsweise durch einen *else*-Zweig für alle nicht explizit betrachteten Fälle.

Unwissen liegt vor, wenn die Elemente der Wissensbasis, die die Expertise repräsentieren, falsche, redundante oder unvollständige Informationen generieren. Diese Informationen können im weiteren Verlauf der Analyse zu einem falschen Endergebnis führen. Die Auswirkung auf die Korrektheit des Endergebnisses ist nicht zwingend, da andere für die Bestimmung der Diagnose konkurrierende oder alternative Informationen im konkreten Fall dominieren können. Ist die Expertise unvollständig, ergeben sich Abschnitte des Wertebereichs, für die keine Beurteilung abgeleitet werden kann. Dieser Zustand kann beabsichtigt sein, wenn kein allgemeines Wissen über die Klassifikation vorhanden ist. Ist er unbeabsichtigt, liegt die Ursache in einer unvollständigen Zuordnungsbeschreibung für die Klasse. Die Unvollständigkeit oder Redundanz von Expertise ist typisch für vage Beschreibungen von Übergangsbereichen beispielsweise bei der komplexen Transformation von numerischen Daten in symbolische Beschreibungen.

Ist in der Wissensbasis ein Fehler der Art Defekt, Schwäche oder Unwissen enthalten, und wird der Bereich, der den Fehler repräsentiert, während der Analyse aktiviert, dann impliziert das Analyseergebnis die Auswirkung des Fehlers. Auch ein Systemabsturz wird als Analyseergebnis definiert, wobei das Resultat die leere Menge aus der Menge der Ausprägungen der resultierenden Objektklassen ist. Existiert ein Verfahren, das diese Wirkung sichtbar macht, ist mit dem Black-Box-Ansatz die Validierung der Wissensbasis eines wissensbasierten Bildanalysesystems den Anforderungen entsprechend durchführbar.

Der Ansatz für das Validierungsverfahren ist die Prüfung der Übereinstimmung zwischen der Spezifikation und Implementation der Wissensbasis mit einem speziellen Black-Box-Verfahren. Für die Durchführung des Verfahrens wird eine geeignete Menge von Testdaten benötigt. Um Fehler der Art Defekte, Schwächen oder Unwissen aufzudecken, muß ein Fall in der Testdatenmenge vorhanden sein, der den Bereich der Wissensbasis aktiviert, in dem der Fehler auftritt. Um die Auswirkung des Fehlers zu erkennen, wird eine Teststrategie verfolgt, die auf spezifizierten Abhängigkeiten der Ausprägungen der Ein- und Ausgabeobjekte im Merkmalsraum basiert. Nach der Definition einer validierungsrelevanten Konfiguration werden die entsprechenden Testfälle in die Testdatenmenge aufgenommen.

Ein Testfall umfaßt die spezifizierte Menge von Ausprägungen der Objekte des Eingabebereichs für eine Analyse. Auf der Basis der Abhängigkeiten werden Testfälle geordnet und Nachbarschaften von Testfällen ermittelt. Von benachbarten Fällen wird eine bestimmte zum Nachbarn relativierbare Darstellung im Analyseergebnis erwartet. Ein durch einen Testfall repräsentierter Fehler wird im Kontext anderer Testfälle durch eine Abweichung von der relativierten Darstellung zum Nachbarn erkannt. Für diesen Ansatz müssen aus-

reichend viele und den Anforderungen genügende Testfälle vorhanden sein.

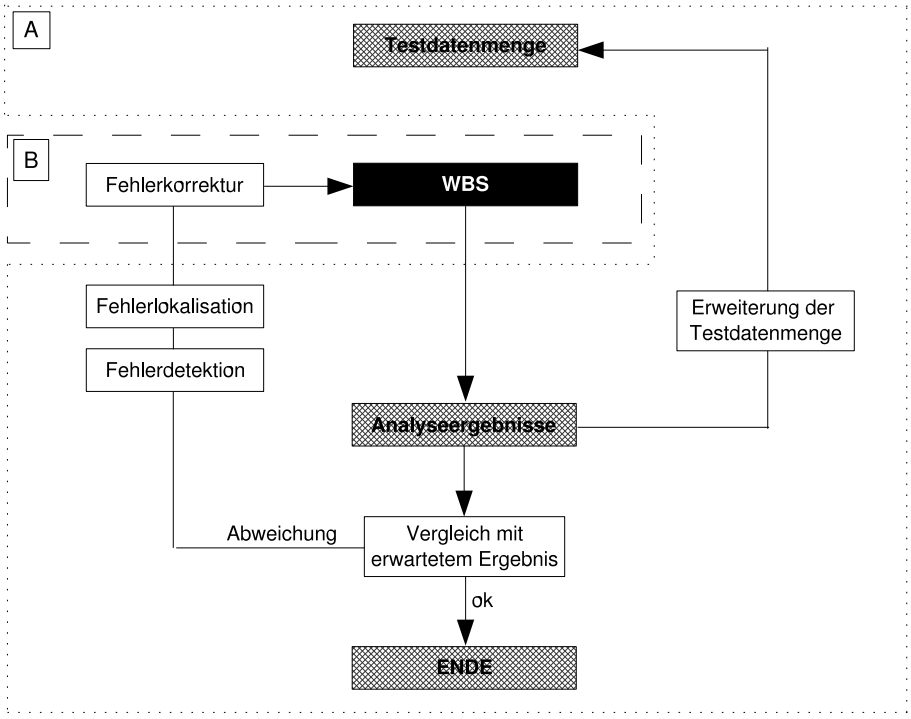

Abbildung 4.4: Ablauf einer Testphase im Validierungsverfahren
A. Auswahl von Testdaten, die durch das wissensbasierte System (WBS) analysiert werden. Die Ergebnisse werden überprüft. Bei der Detektion eines Fehlers werden weitere Maßnahmen zur Lokalisation des Fehlers getroffen. Wird kein Fehler gefunden, ist die Testphase beendet. Sind die Testfälle nicht ausreichend, muß die Testdatenmenge ergänzt werden.
B. Die Fehlerkorrektur ist nicht die Aufgabe der Validierung. Nach der Durchführung der Fehlerkorrektur ist das WBS jedoch erneut zu überprüfen.

Der Ablauf des Verfahrens wie es in Abbildung 4.4 dargestellt ist, beginnt mit der Analyse der ausgewählten Testfälle, die als Testphase bezeichnet wird. Nach der Untersuchung der Analyseergebnisse ist entweder ein Fehler detektiert worden oder die Ergebnisse sind zufriedenstellend. Ist das Ergebnis zufriedenstellend, ist die Testphase beendet. Ist ein Fehler detektiert worden, werden weitere Maßnahmen zur Fehlerlokalisation getroffen. Die Ergebnisse der Nachforschungen führen zur Fehlerbehandlung. Diese gehört nicht zu den Aufgaben der Validierung. Sie setzt wieder an, nachdem das wissensbasierte System verändert wurde, um die Auswirkungen der Fehlerbehandlung zu überprüfen. Sind die Testfälle für die Zielvorgaben der Testphase unzureichend, sind weitere Testfälle in die Testdatenmenge aufzunehmen.

4.3.2 Validierungsrelevante Eigenschaften der Eingabedaten

Die originalen Daten der bildgebenden Verfahren in der Medizin sind Grauwertdarstellungen des interessierenden Objektbereichs. Farbdarstellungen werden aus der Grauwertdarstellung durch Zuordnung von Farbtabellen abgeleitet. Die Menge der Ausprägungen eines Bildes wird somit durch die Dimensionen des Bildes und den Speicherplatz für den Grauwert eines Pixels absolut beschränkt. Mit X Zeilen, Y Spalten einer Bildmatrix und P Bits zur Codierung der Pixelinformation ergibt sich eine Menge von 2^{PXY} codierbaren Ausprägungen.

Durch die Aufnahme ausgewählter Bereiche ergibt sich in den Bildern bei den speziellen medizinischen Fragestellungen eine relativ feste Lage der entsprechend der Anatomie angeordneten Objekte im Bild. Bestimmte Bildbereiche sind in der Regel Hintergrund, andere Bereiche in der Regel Objekten zuzuordnen. Die Konstanz basiert auf der aktiven Beeinflussung von Aufnahmeparametern wie der relativen Lage des Patienten zur Kamera und dem Abstand zwischen Kamera und Aufnahmeobjekt oder der Dauer der Aufnahme. Beispielsweise wird bei einer Indikation für die nuklearmedizinische Diagnostik die Gammakamera so positioniert, daß ein Organ in einer bestimmten Ansicht betrachtet werden kann. Die interessierenden Bereiche werden vor Beginn der Aufnahme im Bild zentriert. In der Routine ergeben sich Standardansichten, die durch die Auswertungssoftware der angeschlossenen Bilddatenauswertung unterstützt werden. Andererseits ist durch den festen strukturellen Aufbau des menschlichen Körpers bei Aufnahmen des gleichen Körperbereichs bei verschiedenen Patienten eine ähnliche Struktur vorhanden.[1]

Mit den Einschränkungen, die durch die feste Aufgabenstellung für die bildgebenden Verfahren in der Medizin gegeben sind, reduziert sich die Anzahl möglicher Ausprägungen im Eingabebereich eines Analysesystems. Unter der Annahme, daß Hintergrund durch niedrige Grauwerte und Objektbereiche durch höhere Grauwerte charakterisiert sind, werden stabile Hintergrundbereiche keine quasi homogenen hohen Pixelwerte aufweisen, stabile Objektbereiche keine quasi homogenen niedrigen Pixelwerte. Quasi homogen umschreibt einen visuellen Eindruck von Homogenität, ohne daß die Grauwertverteilung zwangsläufig einem quantitativen Homogenitätskriterium entspricht. Die Variationen ergeben sich aus den unterschiedlichen natürlichen Erscheinungsformen (normal und pathologisch), dem statistischen Rauschen und Übertragungsfehlern bei der Datenakquisition. Nicht oder nur schlecht reproduzierbare Aufnahmebedingungen wie beispielsweise die Lage des Patienten unter der Kamera bei einer nuklearmedizinischen Untersuchung oder die Verteilung des Radiotracers im Körper führen allein bei einem Patienten mit unverändertem Zustand bei zwei zeitlich dicht aufeinanderfolgenden Untersuchungen zu differierenden Grauwerten in korrespondierenden Pixeln.

Mit der Bildvorverarbeitung in der Low-Level-Phase werden die Bilddaten speziell für die nachfolgende High-Level-Analyse vorbereitet. Praktisch bedeutet es, daß die *Informa-*

[1] Abweichungen davon können durch natürliche oder unnatürliche Einflüsse entstehen, z.B. durch Behinderung oder Unfall.

tion im Bild durch die Methoden der Bildverbesserung und Segmentierung auf den für die Fragestellung relevanten Anteil reduziert wird. Der Übergang zwischen der Low- und High-Level-Phase wird durch primitive symbolische Beschreibungen charakterisiert, auch als Segmentierungsergebnis zu bezeichnen. Primitiv bedeutet, daß einem Datum des Übergangsbereichs eine einfache Bedeutung durch Benennung zugewiesen wird. Typische Ergebnisse sind beispielsweise Konturen oder Binärmasken, die ein Objekt des Bildbereichs repräsentieren.

Trotz der Beschränkungen der Anzahl von Fällen im Eingabebereich, die durch Standardansichten, die Anatomie und die informationsreduzierenden Verarbeitungsschritte in der Low-Level-Phase vorliegen, ergibt sich eine kombinatorische Vielfalt, die einen Test des Analyseverhaltens für alle Ausprägungen ausschließt.

Für das wissensbasierte System stellt sich der Eingabebereich von Ausprägungen eines Objektes zunächst unstrukturiert dar. Eine Ordnung wird durch das modellierte Wissen in der Wissensbasis repräsentiert und führt während der Analyse zu der spezifizierten Klassifikation. Unabhängig von der in der Wissensbasis repräsentierten Ordnung können die Elemente des Eingabebereichs, die numerische Darstellungen von Objekten sind, auf verschiedene charakteristische quantifizierbare Merkmale abgebildet werden. Diese Abbildung liefert die Grundlage für eine Strukturierung des Eingabebereichs, die unabhängig von der Wissensbasis ist. Bezogen auf das Merkmal können über Ausprägungen relativierende Aussagen getroffen werden, z.B. durch eine Anordnung.

Die Nachbarschaftseigenschaft auf der Basis der Merkmale liefert lokal Informationen über die Position der Ausprägungen in der Anordnung. Benachbarte Ausprägungen sind für die nachfolgend erläuterte Validierungsstrategie von Bedeutung. Über die Definition eines Abstands und die Auswahl einer Schranke können ähnliche Eingabekombinationen beschrieben werden. Seien für die Merkmale K_h auf den Teilmengen E_h der Eingabeobjekte die Abbildungsvorschriften gegeben mit $G_h : E_h \rightarrow K_h, 1 \leq h \leq H$

Definition 4.4 *Die A-Ordnung ist das Ergebnis der Anordnung von Φ Kombinationen von E-Ausprägungen $\{e_n^\phi | n = 1, \ldots, N\}, \quad 1 \leq \phi \leq \Phi$ anhand einer Sortiervorschrift bezüglich des Objektmerkmales K_h. Die Argumente der Sortiervorschrift sind die Funktionswerte $G_h(E_h) = \mu_h$.*

Definition 4.5 *Zwei Kombinationen μ und ν von E-Ausprägungen liegen in der A-Umgebung, wenn der Abstand D bezüglich aller Merkmale H des Merkmalsraums K kleiner als eine vorgegebene Schranke Δ ist*

$$D = \sqrt{(\mu_1 - \nu_1)^2 + (\mu_2 - \nu_2)^2 + \cdots + (\mu_H - \nu_H)^2} < \Delta \tag{4.10}$$

Definition 4.6 *Zwei Kombinationen von E- bzw. A-Ausprägungen sind benachbart bzw. sie sind Nachbarn (kurz $\approx$), wenn sie in der A-Ordnung direkt aufeinander folgen.*

Definition 4.7 *Zwei Kombinationen von E- bzw. A-Ausprägungen sind k-benachbart bzw. sie sind k-Nachbarn (kurz $\overset{k}{\approx}$), wenn zwischen den beiden Ausprägungen in der A-Ordnung $(k-1)$ Ausprägungen liegen, für $k = 1, 2, \ldots$.*

Bemerkung 1 *Die Anzahl der resultierenden Ausprägungen des Ausgabebereichs ist in einem automatischen wissensbasierten Bildanalysesystem ohne Lernkomponente und monotone Schlußfolgerung höchstens so groß wie die Anzahl der Ausprägungen des Eingabebereichs.*

> Erläuterung: Die Wissensbasis ist abgeschlossen. Es gibt keinen variierbaren Wissensgewinn während der Analyse von außen (z.B. durch Interaktion). Die beim Start der Analyse in die Wissensbasis eingegebenen Daten werden transformiert. Lernt das System nicht, dann gibt es auch für jede Eingabekombination im Verlauf der Zeit immer nur eine Ausgabekombination.

Im folgenden werden A-Ausprägungen, die aus unterschiedlichen E-Ausprägungen resultieren, als unterschiedlich behandelt, auch wenn für zwei Fälle Gleichheit erwartet wird. Die Unterscheidung ist relevant für die Beurteilung des Analyseverhaltens der Wissensbasis beispielsweise hinsichtlich der Unterscheidungsfähigkeit (Auflösungsvermögen) bei dicht benachbarten Testfällen oder zur Fehlerdetektion. Zur Erläuterung betrachte man folgendes Beispiel.

Beispiel:

Einfache Merkmale von Objekten in Bildern sind ihre Projektionsflächen, gemessen durch Zählung der zur Fläche gehörenden Pixel, ihr Schwerpunkt oder der mittlere Grauwert. Betrachtet man eine Sequenz von Bildmatrizen, können Merkmale in einer Zeitreihendarstellung die Flächen- oder Schwerpunktänderung sein.

Sei $\mathbf{E} = E_{Niere}$. Wird das Objekt E_{Niere} in einer nuklearmedizinischen Aufnahme auf das Merkmal $K_{Projektionsfläche}$ abgebildet, ist eine denkbare Sortiervorschrift für eine A-Ordnung:

> Sortiere die Ausprägungen nach den ganzzahligen Werten der Pixelsummen der Projektionsflächen in aufsteigender Reihenfolge.

Das Merkmal $K_{Projektionsfläche}$ ist bei einer Bilddimension von $X * Y$ Pixeln auf dem Intervall $[0, X * Y]$ definiert. Die Schranke für die Δ-Umgebung für zwei Kombinationen μ und ν von E-Ausprägungen ist für alle Ausprägungen des Objektes E_{Niere} aus dem Intervall

$$\Delta \in [\quad min|G(e^{\mu}_{Niere}) - G(e^{\nu}_{Niere})|, max|G(e^{\mu}_{Niere}) - G(e^{\nu}_{Niere})| \quad] \tag{4.11}$$

Sei die Projektionsfläche der Niere durch die Kontur manifestiert, beispielsweise als Binärdarstellung in der $X * Y$-Matrix. Ein Punkt in der Matrix wird mit (x, y) indiziert, $1 \leq x \leq X, 1 \leq y \leq Y$.

Die Abbildungsvorschrift $G_{Projektionsfläche} = G$, die auf $\mathbf{E} = \mathbf{E}_{Projektionsfläche}$ definiert ist, ergibt sich aus

$$\delta(e^{\mu}_{Niere}, x, y) = \begin{cases} 1 & , \quad (x,y) \; liegt \; innerhalb \; der \; Kontur \; e^{\mu}_{Niere} \\ 0 & , \quad sonst \end{cases} \tag{4.12}$$

$$G(e^{\mu}_{Niere}) = \sum_{x=1}^{X} \sum_{y=1}^{Y} \delta(e^{\mu}_{Niere}, x, y) \tag{4.13}$$

Sei $X = 64$ und $Y = 64$, dann ist der Wertebereich von $K = [0, 4096]$. Für E_{Niere} liegen vier Ausprägungen vor

$$E_{Niere} = \{e^1_{Niere}, e^2_{Niere}, e^3_{Niere}, e^4_{Niere}\} \tag{4.14}$$

mit den Merkmalswerten

$$G(e^1_{Niere}) = 96, G(e^2_{Niere}) = 68, G(e^3_{Niere}) = 108, G(e^4_{Niere}) = 100 \tag{4.15}$$

Nach der Sortiervorschrift ist die A-Ordnung der Ausprägungen

$$e^2_{Niere} \rightarrow e^1_{Niere} \rightarrow e^4_{Niere} \rightarrow e^3_{Niere} \tag{4.16}$$

Die Nachbarschaften sind

$$e^2_{Niere} \overset{1}{\approx} e^1_{Niere}, e^1_{Niere} \overset{1}{\approx} e^4_{Niere}, e^4_{Niere} \overset{1}{\approx} e^3_{Niere} \tag{4.17}$$

$$e^2_{Niere} \overset{2}{\approx} e^4_{Niere}, e^1_{Niere} \overset{2}{\approx} e^3_{Niere}, e^2_{Niere} \overset{3}{\approx} e^3_{Niere} \tag{4.18}$$

4.3.3 Validierungsrelevante Eigenschaften der Analyseergebnisse

Sind die E-Ausprägungen derselben E-Objekte aus zwei verschiedenen Testfällen innerhalb einer geeignet gewählten Schranke Δ_E benachbart, so ist anzunehmen, daß auch die durch die Wissensbasis generierten A-Ausprägungen innerhalb einer von Δ_E abhängigen Schranke

Δ_A benachbart sind. Werden die Testfälle bezüglich eines Merkmals der E-Ausprägungen angeordnet, gilt die Ordnung auch für die A-Ausprägungen. Die Nachbarschaftseigenschaft bleibt erhalten. Die funktionale Beziehung F impliziert bei angeordneten Testfällen ein Muster der Merkmalswerte der A-Ausprägungen.

Art und Umfang des Ausgabebereiches ergibt sich aus der Menge der A-Objekte und ihrer Abhängigkeiten. Gegenseitiger Ausschluß von Objekten im Sinne von Klassen, Eindeutigkeit der Systemantwort oder eine Gewichtung der Ergebnisse sind Aspekte, die für die Interpretation von Analyseergebnissen einzeln und im Kontext mit anderen Ergebnissen zu berücksichtigen sind. Das Spektrum der Struktur des Ausgabebereichs reicht von einer einfachen Struktur mit zwei A-Objekten, die sich gegenseitig ausschließen, eine eindeutige Systemantwort repräsentieren und keine Gewichtung besitzen auf der einen Seite, bis zu einer komplexen Struktur mit hunderten von A-Objekten, die sich nicht gegenseitig ausschliessen, keine eindeutige Systemantwort liefern und eine Gewichtung besitzen, auf der anderen Seite.

Die folgenden Betrachtungen beziehen sich auf die Validierung eines Mehrobjektsystems mit der Gewichtung als ein Element der Ergebnisstruktur eines Objektes. Die Objekte müssen sich nicht gegenseitig ausschließen.

Ein Ziel der Validierung ist die Feststellung, ob die resultierende A-Ausprägung im Sinne der Spezifikation korrekt ist. Bei der Beurteilung einer Gewichtung wird das Erreichen des Ziels umso schwieriger, je feiner die Gewichtungen sind. Fehler, die sich über gröbere Abweichungen vom erwarteten Ergebnis darstellen, sind auch qualitativ durch visuelle Beurteilung zu erfassen. Jedoch kann eine fehlerhafte Gewichtung richtig erscheinen, da beispielsweise die numerische Abweichung in einem komplexeren Ergebnismuster bei visueller Beurteilung nicht auffällig ist. Das Problem soll mit dem folgenden Beispiel weiter erläutert werden.

Beispiel:

Gegeben sei eine Wissensbasis für die Geschwindigkeitsanalyse eines Autos mit der Spezifikation

$S(w = \text{Fahrzeuggeschwindigkeit}) = S = \{S_E, S_A, F\}$

$S_E = \{\text{AUTO}\}$

$S_A = \{\text{KEINE_BEWEGUNG}, \text{LANGSAM}, \text{NORMAL}, \text{SCHNELL}, \text{UNDEFINIERT}\}$

Die Wissensbasis bildet die Bewegung eines Autos, das parallel zur Projektionsfläche einer starr aufgebauten Kamera auf einer Straße vorbeifährt, auf die Objekte des Ausgabebereichs ab. Zwei Aufnahmen werden in einem festen Zeitabstand durchgeführt. Mit der Klassifikation ist eine Gewichtung $g \in [0.0, 1.0]$ verknüpft, die die Übereinstimmung des aktuellen Falles mit der Bewegungsbeschreibung ausdrücken soll. Mit wachsender Gewichtung nimmt die Übereinstimmung zu. Die Gewichtung ist ein Element der A-Ausprägung eines A-Objektes. Die Objekte schließen sich nicht gegenseitig aus. Es sind gleichzeitig al-

ternative positive Gewichtungen zulässig. Das fokussierte Merkmal K der A-Ausprägungen A_m ist $K =$ Gewichtung für

$$m \in \{\text{KEINE_BEWEGUNG}, \text{LANGSAM}, \text{NORMAL}, \text{SCHNELL}, \text{UNDEFINIERT}\}$$

$$A_m.\text{Gewichtung}(a_m^\mu) = g_m$$

Es liegen vier Kombinationen von E-Ausprägungen vor. Ein Merkmal der E-Objekte ist der als Geschwindigkeit bezeichnete absolute Wert der Schwerpunktänderung der Projektionsfläche des Autos in den beiden Bildmatrizen. Aus der Schwerpunktänderung ist aufgrund der einfachen Straßenlage und Verkehrssituation die Geschwindigkeit in km/h einfach rechnerisch zu ermitteln, so daß das Merkmal auf dem Werteintervall für km/h definiert werden kann

$$G(e_{\text{AUTO}}^\mu) = \text{Geschwindigkeit in km/h} \in [0, 200]$$

$$E_{\text{AUTO}} = e_{\text{AUTO}}^1, e_{\text{AUTO}}^2, e_{\text{AUTO}}^3, e_{\text{AUTO}}^4$$

$$F(e_{\text{AUTO}}^\mu) = (a_{\text{KEINE_BEWEGUNG}}^\mu, a_{\text{LANGSAM}}^\mu, a_{\text{NORMAL}}^\mu, a_{\text{SCHNELL}}^\mu, a_{\text{UNDEFINIERT}}^\mu)$$

$$G(e_{\text{AUTO}}^1) = 50, G(e_{\text{AUTO}}^1) = 40, G(e_{\text{AUTO}}^1) = 60, G(e_{\text{AUTO}}^1) = 70$$

Nach der Analyse liegen folgende Resultate vor: die A-Ausprägungen sind durch den Wert ihres Merkmals Gewichtung repräsentiert

$$F(e_{\text{AUTO}}^\mu) = (g_{\text{KEINE_BEWEGUNG}}, g_{\text{LANGSAM}}, g_{\text{NORMAL}}, g_{\text{SCHNELL}}, g_{\text{UNDEFINIERT}})$$

$$F(e_{\text{AUTO}}^1) = (0.0, 0.2, 0.6, 0.2, 0.4)$$

$$F(e_{\text{AUTO}}^2) = (0.0, 0.4, 0.7, 0.0, 0.3)$$

$$F(e_{\text{AUTO}}^3) = (0.0, 0.0, 1.0, 0.6, 0.0)$$

$$F(e^4_{\mathrm{AUTO}}) = (0.0, 0.0, 0.8, 1.0, 0.0)$$

Für die Validierung liegen die Eingangsdaten in das wissensbasierte Bildanalysesystem (z.B. Binärmasken) und das Merkmal Geschwindigkeit des Autos in den vier Fällen vor. Unabhängig von der Qualität der Bildvorverarbeitung soll geprüft werden, wie gut die Ergebnisse der wissensbasierten Analyse der qualitativen Spezifikation genügen.

Zunächst soll ein Experte anhand der vorhandenen Informationen qualitative Aussagen treffen. Die Fälle werden nacheinander zur Beurteilung vorgelegt. Der Experte erwartet die mit NORMAL beschriebene Geschwindigkeit für die vorliegende Verkehrssituation im Bereich um 60 km/h. Das Ergebnis der Analyse des ersten Falles (Geschwindigkeit = 50 km/h) erscheint akzeptabel. Die Bewertung $g(NORMAL) = 0.6$ drückt aus, daß die gemessene Geschwindigkeit mit deutlicher Sicherheit als normal zu bezeichnen ist. Jedoch bleibt eine Unsicherheit, die vom Experten nachvollziehbar ist, da eine Differenz von 10 km/h zu der als normal akzeptierten Geschwindigkeit besteht. Weiterhin liegen die alternativen positiven Bewertungen $g(LANGSAM) = 0.2$ und $g(SCHNELL) = 0.2$ im Erwartungsbereich des Experten. Für die weiteren drei Fälle ergibt die Evaluierung ähnliche positive Resultate.

Da weder ein Programmabsturz erfolgte, noch eine negative Expertenmeinung vorliegt, taucht die Frage auf: könnte das System als einsetzbare Version bezeichnet werden?

Die Betrachtung der Fälle in einer A-Ordnung liefert ein anderes Ergebnis. Anhand des Merkmals Geschwindigkeit werden die Ausprägungen des Eingabebereichs für die A-Ordnung in der Reihenfolge aufsteigender Werte sortiert

$$e^2_{\mathrm{AUTO}} \rightarrow e^1_{\mathrm{AUTO}} \rightarrow e^3_{\mathrm{AUTO}} \rightarrow e^4_{\mathrm{AUTO}}$$

Die A-Ordnung der E-Ausprägungen gilt gleichfalls für die A-Ausprägungen.

In der Tabelle 4.1 sind die Gewichtungen der A-Objekte für die 4 Fälle spaltenweise in der Reihenfolge der A-Ordnung eingetragen. Auffällig ist, daß in der Klasse NORMAL bei Fällen mit monoton steigenden Geschwindigkeiten oszillierende Bewertungsänderungen, von 0.7 auf 0.6 fallend, auf 1.0 steigend und wieder auf 0.8 fallend, vorliegen. Zu erwarten wäre maximal ein wechselndes Vorzeichen bei der Bewertungsänderung durch eine wachsende Gewichtung für NORMAL bei steigender Geschwindigkeit, die dann wieder abfällt, wenn die Geschwindigkeit für normale Verhältnisse zu schnell wird. Nachdem die Gewichtung für NORMAL bei monoton steigender Geschwindigkeit den maximalen Wert angenommen hat, sollte sie mit Beginn der Bewertungsminderung nur monoton fallen. Eine später folgende erneute Zunahme der Sicherheit ist falsch.

Die Bewertung 0.6 in der Ausprägung e^1_{AUTO} erscheint bei isolierter Betrachtung als akzeptable Lösung. Erst der direkte Vergleich mit benachbarten Fällen in der A-Ordnung deckt eine Diskrepanz der Ausprägungen in dem Wertemuster der implizierten Anordnung

A-Objekte	Testfall 2	Testfall 1	Testfall 3	Testfall 4
KEINE_BEWEGUNG	0.0	0.0	0.0	0.0
LANGSAM	0.4	0.2	0.0	0.0
NORMAL	0.7	0.6	1.0	0.8
SCHNELL	0.0	0.2	0.6	1.0
UNDEFINIERT	0.3	0.4	0.0	0.0

Tabelle 4.1: Aus der Analyse einer Fahrzeugbewegung resultierende Gewichtungen $\in [0.0, 1.0]$ der A-Objekte KEINE_BEWEGUNG, LANGSAM, NORMAL, SCHNELL, UNDEFINIERT für vier Testfälle. Die Reihenfolge der Spalten entspricht der A-Ordnung *aufsteigende Geschwindigkeit* der Testfälle.

auf. Zu erwarten wäre ein Verlauf der Gewichte g_m als Ergebnis der Analysen der nach aufsteigenden Geschwindigkeiten sortierten E-Ausprägungen, der entweder monton steigt, monoton fällt oder aus einem monoton steigenden gefolgt von einem monoton fallenden Abschnitt zusammengesetzt ist, je nachdem welcher Ausschnitt der in dieser A-Ordnung angeordneten Testfälle betrachtet wird.

Ist das Muster des Gewichtsverlaufs aus der Spezifikation ableitbar, so stellt die Abweichung einen Widerspruch zur Spezifikation dar. Die Ursache ist auf einen Fehler in der Wissensbasis zurückzuführen. Im idealen Fall stimmt nach der Korrektur des Fehlers das Muster der Gewichte mit dem spezifizierten Verlauf überein. Für die Beschreibung dieses Zustandes wird der Begriff homogen (Definition 4.8) eingeführt. Kann der Zustand nicht als homogen bezeichnet werden, ist das als Aufmerksamkeitssignal für einen Fehler zu interpretieren. Die Art und die Lokalisation des Fehlers kann daraus nicht abgeleitet werden. Jedoch ist anhand des fehleraktivierenden Testfalls ein Ansatzpunkt für eine Suche gegeben.

Definition 4.8 *In einem klassifizierenden wissensbasierten Bildanalysesystem wird die Anordnung von A-Ausprägungen bezüglich in einer A-Ordnung geordneter E-Ausprägungen als homogen bezeichnet, wenn das Wertemuster der Merkmale der geordneten A-Ausprägungen mit dem durch die Spezifikation festgelegten oder aus der Spezifikation ableitbaren Muster übereinstimmt.*

Wird wie im Beispiel w=Fahrzeuggeschwindigkeit das Muster der A-Ausprägungen durch das Merkmal Gewichtung manifestiert, dann sind in dem Fall die angeordneten A-Ausprägungen homogen bezüglich der zugehörigen angeordneten E-Ausprägungen, wenn keine A-Ausprägungen existieren, deren Gewichtungsänderungen im Bereich der k- Nachbarschaft ($k > 1$) wechselnde Vorzeichen besitzen. In der 1-Nachbarschaft steht der Vorzeichenwechsel von plus nach minus nicht zwangsläufig im Widerspruch zur Spezifikation.

In der simplen Beispielswissensbasis kann vielleicht eine Homogenität nach Definition 4.8 erreicht werden. Bei einem komplexeren und umfangreicheren System ist damit

zu rechnen, daß die Homogenität nur annäherungsweise erreichbar ist. Hierfür können verschiedene Ursachen verantwortlich sein, die an dieser Stelle nicht weiter untersucht werden. Am konkreten Beispiel der Validierung in Kapitel 5 werden spezielle Ursachen diskutiert. Zur Beschreibung des Zustandes der approximierten Homogenität wird der Begriff quasi-homogen eingeführt.

Definition 4.9 *In einem klassifizierenden wissensbasierten Bildanalysesystem wird die Anordnung von A-Ausprägungen bezüglich in einer A-Ordnung geordneter E-Ausprägungen als quasi-homogen bezeichnet, wenn das Wertemuster der Merkmale der geordneten A-Ausprägungen mit dem durch die Spezifikation festgelegten oder aus der Spezifikation ableitbaren Muster höchstens innerhalb eines zugelassenen Bereichs abweicht.*

Die Aussagen über Homogenität gelten nur im Zusammenhang mit Merkmalen, die direkt oder indirekt für die Analyse relevant sind. Merkmale, die direkt oder indirekt innerhalb der wissensbasierten Analyse keine Kriterien für die Entscheidungsfindung darstellen, sind für die Validierung uninteressant. Von Bedeutung sind die Merkmale, die zur Entscheidungsfindung beitragen. Die gesamte Menge der Merkmale, auf die die E-Ausprägungen abbildbar sind, bilden eine Übermenge der Merkmale, die in der Wissensbasis die Basis für die Entscheidungsfindung bilden.

Wird für die Überprüfung der Wissensbasis w=Fahrzeuggeschwindigkeit das Merkmal Projektionsfläche als Ordnungskriterium benutzt, ist klar, daß bei der Geschwindigkeitsklassifikation keine Hinweise ableitbar sind, wenn die Lageänderung anhand der Schwerpunktänderung des Objektes in zwei Bildern das relevante Kriterium in der Wissensbasis ist. Wird die Projektionsfläche jedoch in der Wissensbasis anstatt der Schwerpunktänderung beispielsweise für die Geschwindigkeitsbeschreibung von entgegenkommenden oder sich entfernenden Fahrzeugen eingesetzt, verliert das Merkmal Schwerpunktänderung an Bedeutung.

In einem komplexeren System als dem im Beispiel vorgestellten liefern in der Regel mehrere Parameter Informationen für die Klassifikationen. Die Abhängigkeit der Parameter voneinander ist für die Validierung zu verdeutlichen. Diese Bedingungen beeinflussen wesentlich die Teststrategie.

4.3.4 Möglichkeiten und Einschränkungen durch originale und künstliche Testdaten für eine adäquate Teststrategie

Ein allgemein praktizierter Ansatz der Validierung von wissensbasierten Systemen in der Medizin ist es, die Analyseergebnisse einzelner originaler Fälle durch einen oder mehrere Experten, die teilweise auch an der Entwicklung des Systems beteiligt waren, prüfen zu lassen oder mit einem *Gold Standard* zu vergleichen. Im ersten Fall besteht die Gefahr, wie es das Beispiel aus Kapitel 4.3.3 verdeutlicht hat, daß die Beurteilung einer Gewichtung bei einem fehlerhaften Wert diese Situation unter bestimmten Bedingungen nicht aufdeckt. Die

Abweichung der Gewichtung von dem korrekten Wert ist relativ gering. Je nach Fähigkeit des Experten wären mehr oder weniger große Abweichungen als Fehler erkennbar.

Der relativ große Aufwand der Entwicklung eines automatisch ablaufenden Systems mit reproduzierbaren Resultaten ist kritisch zu bewerten, wenn für die Validierung ausschließlich das nicht reproduzierbare vage Wissen eines Experten über die Qualität entscheidet. Wie genau können eigentlich Experten sehr fein differenzierbare gewichtete Aussagen abwägen und wie sind die nicht vom Experten beurteilten später in der klinischen Routine auftretenden Resultate zu bewerten? Soll eine Validierung im Sinne der Qualitätssicherung umfassend und informativ sein, sind die von den Experten gelieferten Hinweise nicht ausreichend. Ihre qualitative Beurteilung eines Analyseergebnisses basiert auch auf vagem Wissen und Heuristiken, das zu nicht reproduzierbaren Systembewertungen führt. Mit dem Verfahren können Fehler nicht zweifelsfrei aufgedeckt werden. Es ergibt sich lediglich ein subjektiver Gesamteindruck. Der Vergleich mit einem *Gold Standard* ist erstrebenswert, da dabei zweifelsfrei das Vorhandensein oder Nichtvorhandensein einer Krankheit feststellbar ist. Der Nachteil ist, daß ein *Gold Standard* oft nicht zur Verfügung steht.

Im Gegensatz zur qualitativen Methode unterstützen quantitative Methoden feine Differenzierungen und liefern reproduzierbare Werte. Angenommen, es existiert eine entsprechende Methode, ist es denkbar, daß die Qualitätsbeschreibung nicht nur auf einzelne Testfälle bezogen ist, sondern auch auf eine Teilmenge von Fällen verallgemeinert werden kann. O'Keefe et al. fordern in [OKE87] überall dort, wo es möglich ist, quantitative Methoden einzusetzen, um die Information zur Produktion von *Vertrauens-Intervallen* auszunutzen, im Gegensatz zu Einzelfall-Abschätzungen.

Abgesehen von der Kritik an qualitativen oder quantitativen Verfahren sind für diese Verfahren Testdaten nötig. Im idealen Fall decken sie den Eingabebereich vollständig ab. Der ideale Fall tritt praktisch nicht auf, da der Zeitaufwand für die Analyse aller Fälle zu hoch wäre, sofern es überhaupt gelingt, alle Fälle in die Testdatenmenge aufzunehmen. Für die Zusammenstellung der Testdaten gibt es zwei Hauptgesichtspunkte

- die Auswahl der Testdaten soll es ermöglichen zu zeigen, daß die Wissensbasis im Sinne der Spezifikation korrekt ist
- mit den ausgewählten Testdaten sollen möglichst viele Fehler in der Wissensbasis aufgedeckt werden

Als Testdaten bieten sich zunächst die originalen Patientenstudien an. Die Auswahl oder Verfügbarkeit von originalen Bilddaten wird durch verschiedene Umstände beschränkt, die sich auf die Möglichkeiten der Validierung auswirken:

1. Die Zusammensetzung des Patientenkollektivs ist durch die Morbiditätsrate geprägt: Die Häufigkeit einer Erkrankung wird durch die Morbiditätsrate ausgedrückt. Die Anzahl der Patienten, die mit dieser Erkrankung in einem Krankenhaus oder einer Praxis behandelt werden, hängt von dieser Rate ab. Wird ein bildgebendes Verfahren

zur Diagnose eingesetzt, liegen entsprechend der Morbidität mehr oder weniger Bilddaten einer Krankheitsform vor. Werden Studien für eine Validierung gesammelt, sind in dem Datenpool Krankheiten, die sehr selten vorkommen, auch nur durch wenige Studien repräsentiert, während für häufige Krankheiten ein breiteres Spektrum von Ausprägungen verfügbar ist. Ein dichteres Netz von Fällen bedeutet auch eine bessere Abdeckung des Spektrums der Analysemöglichkeiten der Wissensbasis, während ein grobmaschiges Netz die Beurteilung der Analysen ähnlicher Fälle behindert.

2. Beschränkt durch die Belastungen während der Untersuchung können nicht beliebig viele Patienten beliebig oft untersucht werden:
   Die Belastungen einer Untersuchung sind einerseits patienteninhärent, d.h. eine schmerzhafte oder auf andere Art belastende Ruhigstellung, die für eine Aufnahme notwendig ist, kann nur bei einer Indikation für eine Diagnosefindung gerechtfertigt sein. Wiederholte oder zeitlich verlängerte Aufnahmen sind aus dem alleinigen Anspruch einer Systemvalidierung heraus ausgeschlossen. Andererseits kann die Belastung durch das bildgebende Verfahren vorliegen, z.B. die Strahlenbelastung bei der Röntgen-Computertomographie.

3. Beschränkt durch die Kosten können nicht beliebig viele Studien akquiriert werden:
   Mit jeder Untersuchung fallen Material- und Personalkosten an, die abzudecken sind.

4. Ein definierter Fall des spezifizierten Analysespektrums der Wissensbasis, der nicht durch einen Patienten des Kollektivs repräsentiert ist, kann nicht getestet werden:
   Aus der Sicht der Validierung muß das spezifizierte Analysespektrum der Wissensbasis möglichst dicht getestet werden. Die Vollständigkeit wird nicht durch einen Datenpool definiert, sondern über die Spezifikation des Systems. Entsprechend dieser Spezifikation sollen die Testfälle das Spektrum des Eingabebereichs möglichst dicht abdecken. Sind für einen Teilbereich des Systems im Datenpool keine repräsentativen Testfälle vorhanden, kann dieser Bereich auch nicht validiert werden. Die Korrektheit der Analyse eines Falles, der im späteren Routineeinsatz auftritt und diesem Bereich zuzuordnen ist, ist somit nicht zweifelsfrei.

5. Ist nur die Wissensbasis zu testen, können Einflüsse durch die Bildvorverarbeitung nicht ausgeschlossen werden:
   Die Eingabedaten in die Wissensbasis sind vorverarbeitete Bilddaten. Verwendet man für die Validierung nur originale Bilddaten, müssen diese zunächst segmentiert werden. Spezifische Einstellungen des eigenständigen Methodenmoduls beeinflussen das Analyseergebnis. Tritt ein Fehler auf, ist die Lokalisation nicht auf die Wissenbasis beschränkt, der Suchraum wird unverhältnismäßig groß. Die Effektivität der Validierung steigt mit der Trennung der Low- und High-Level-Bereiche. Die Qualität der Analyse wird unabhängig von der Vorverarbeitung festgestellt. Die Anpassung an andere Vorverarbeitungsmodule wird unterstützt.

6. Die Variabilität der Akquisitionsbedingungen ist durch die interaktive Steuerung gegeben:
   Die Akquisitionsbedingungen variieren bei der Aufnahmezeit, der Objektlage im Bild, dem Aufnahmemodus, der Kameraeinstellung (z.B. Zoom der Gammakamera) usw. Ein definiertes Maß zur Relativierung der Akquisitionsbedingungen der Testdaten ist jedoch notwendige Voraussetzung für die Vergleichbarkeit der Ergebnisse.

Die Verwendung originaler Daten bietet für die Validierung eines Systems auch Vorteile:

1. Sie repräsentieren die natürliche Variabilität der Bilddaten, deren Einfluß auf die Analyse auszuschliessen ist:
   Keine Studie ist reproduzierbar. Mehrere Studien eines Patienten mit unverändertem Zustand werden immer unterschiedliche numerische Werte enthalten. Eine exakte Duplizierung ist ein unwahrscheinliches Ereignis. Trotz der unterschiedlichen numerischen Bilddaten ist in diesen Studien der gleiche Zustand repräsentiert. Eine wissensbasierte Analyse muß in allen Fällen das gleiche Resultat liefern, also invariant gegenüber natürlicher Variabilität sein.

2. Die Wissensbasis wird unter Bedingungen getestet, die teilweise den Bedingungen im späteren Einsatzbereich entsprechen:
   Die unterschiedlichen Faktoren durch die interaktive Steuerung des Aufnahmeprozesses, die neben der natürlichen Variabilität der Bilddaten eine Variabilität bei der Datenakquisition hervorrufen, dürfen keinen Einfluß auf die Zustandsbeschreibung durch die Wissensbasis besitzen. Die Invarianz des Systems gegenüber jedweder vom Zustand des aufgenommenen Objekts unabhängigen Variabilität zu prüfen, ist am besten mit den Daten durchzuführen, die diese Variabilität enthalten. Die Prüfung ist nur sinnvoll, wenn abgesicherte Befunddaten, also Zustandsbeschreibungen, vorliegen.

Die Gegensätzlichkeit der Argumentation ist absichtlich durch die scheinbar widersprüchlichen Forderungen bei der Aufzählung der Vor- und Nachteile hervorgehoben. Besonders die Variabilität der Originaldaten ist ein herausragendes Argument für die Auswahl der Testdaten aus einer Menge von realen Patientendaten. Ausschließlich Patientendaten einzusetzen, kann allgemein nicht die Forderungen nach einer bereichsabdeckenden Validierung erfüllen. Idealerweise ist die Menge der Testdaten vollständig in Bezug auf das Analysespektrum des Systems, d.h. jede mögliche Analyse ist durch einen Testfall repräsentiert. Der Aufwand für die angestrebte Abdeckung des Analysespektrums einer Wissensbasis durch Testdaten wird von der Größe des Eingabebereichs, der Komplexität der Systemstrukturen und den Möglichkeiten der Differenzierung der Aussagen beeinflußt. Je kleiner der Eingabebereich, je einfacher die Systemstrukturen und je kleiner die Variabilität der Aussagen, umso geringer ist der Aufwand für die Abdeckung des Spektrums durch Testdaten.

Wissensbasierte Systeme im medizinischen Bereich und ganz besonders in der Bildanalyse fallen nicht darunter. Das Bildanalysesystem steht zunächst in besonderer Konkurrenz zu den visuellen Fähigkeiten des Mediziners, der in der Regel über Jahre Erfahrung bei der visuellen Beurteilung von Bilddaten für die verschiedenen medizinischen Fragestellungen gesammelt hat. Hier sind nur Systeme sinnvoll, die in ihren Funktionalitäten den gleichen Anforderungen unter vergleichbaren Bedingungen wie der Experte vor Ort genügen. Simple Fragestellungen kann der Experte selbst sehr viel schneller beantworten. Erst komplexere Probleme können den Einsatz einer Entscheidungsunterstützung begründen. Es ist deutlich, daß für die Qualitätssicherung wissensbasierter Applikationen die Zusammenstellung und der Umfang der Testdatenmenge neben der Teststrategie von entscheidender Bedeutung ist.

Unter den Nachteilen von originalen Patientendaten stellt ihre eingeschränkte Verfügbarkeit aus der Sicht der Validierung eine besondere Schwierigkeit dar, um ein gesetztes Maß an Qualität zu sichern. Zur Lösung des Problems bieten sich künstliche Daten an, die als Hard- und Softwarephantome zusammengefaßt sind. Aus der Sicht der Validierung wird die Verwendung eines künstlichen Datums nur sinnvoll, sofern es in Anlehnung an die Forderungen des Validierungsverfahrens generierbar ist.

Die Verwendung künstlicher Daten für die Systemvalidierung hat sich bis heute im Bereich der Qualitätssicherung medizinischer Software nicht etablieren können. Die Hauptargumente gegen die Verwendung künstlicher Daten, wie sie beispielsweise durch Busemann Sokole et al. [BUS90a], [BUS90b] oder im COST Project B2 zur Qualitätssicherung von nuklearmedizinischer Software [BRI92a], [BRI92b] vertreten werden, sind:

- simulierte Daten und mathematische Tests besitzen keine klinische Validität, die unabhängig verifiziert werden kann
- künstliche Daten inkorporieren nicht die realen biologischen Variationen.

Diese Argumente begründen den weitverbreiteten Ansatz, bei der Qualitätskontrolle von medizinischer Software im Bereich der Bildanalyse diese ausschließlich mit realen Daten von realen Patienten mit realen Krankheiten durchzuführen. Ein großer Teil der Aktivitäten zur Verbesserung der Qualitätssicherung beschäftigt sich mit dem Aufbau einer Datenbank von realen Bilddaten mit klaren Diagnosen, die unter standardisierten Bedingungen akquiriert werden. Die Software anhand von ausgesuchten Fällen zu testen, deren Diagnose klar ist, bietet die Möglichkeiten, für die einzelnen Fälle die Korrektheit festzustellen, oder ein System zu justieren, z.B. durch Einstellungen von Schwellwerten oder Parametern von Funktionen. Offen bleibt, ob ähnliche Fälle wie der geprüfte Fall, die nicht in der Datenbank gespeichert sind, durch das System ebenso korrekt analysiert werden.

Neben den Daten, die in Befund und Diagnose zweifelsfrei sind, stellen die Daten, die in Grauzonen zwischen eindeutigen Darstellungen anzuordnen sind, die zweite große Gruppe von Eingabedaten. Existiert kein Expertenwissen für Vertreter dieser Gruppe, muß eine Klassifikation in die Menge der undefinierten Fälle erfolgen. Ist das Expertenwissen

über die Beurteilung nicht sicher, darf die Klassifikation nicht eindeutig sein, sofern in der Wissensbasis eine feine Differenzierung modelliert ist.

Eine wirtschaftlich vertretbare Validierung mit künstlichen Daten wird auch unvollständig bleiben. Geeignete Teststrategien müssen die Fehlerhäufigkeiten minimieren. Die Teststrategie richtet sich nach den zu prüfenden spezifizierten Fähigkeiten des Systems. Die Testdatenmenge muß die Testfälle liefern, die für die Teststrategie notwendig sind. Die Verfügbarkeit entsprechend den strategischen Erwägungen ist bei originalen Patientenstudien nicht kontrollierbar.

Sollen künstliche Daten eingesetzt werden, wird der Aufwand der Validierung um das Verfahren der Generierung vergrößert. Für die Validierung der Wissensbasis sind die künstlichen Daten Elemente des Eingabebereichs der Wissensbasis, d.h. sie simulieren vorverarbeitete Bilddaten. Es handelt sich nicht nur um eine andere Darstellungsform, sondern um reduzierte Information im Vergleich zu den originalen Patientendaten. Die Variabilität von realen Bilddaten ist größtenteils nicht mehr vorhanden, sondern ist durch die Low-Level-Vorverarbeitung reduziert. Ist das Ergebnis der Vorverarbeitung beispielsweise die Projektionsfläche eines Organs in einem Binärbild, ist die Variabilität der Grauwertdarstellung auf die Variabilität von Form und Größe der Projektionsfläche reduziert. Die Modellierung z.B. von statistischem Rauschen entfällt.

Durch die Simulation der Vorverarbeitungsergebnisse wird die Variabilität der Bilddaten reduziert, eine Phantomdatenbeschreibung erleichtert, und die Wissensbasis kann unabhängig von Einflüssen der Vorverarbeitung validiert werden.

Wird die Wissensbasis wieder mit einem Vorverarbeitungsmodul gekoppelt, kann das gesamte System getestet werden, wobei jetzt die Daten über die Qualität der Wissensbasis vorliegen und auftretende Fehler einfacher zu lokalisieren sind. Die Testdaten werden durch originale Patientendaten gebildet. Ist es mit der Methode der Generierung künstlicher Daten möglich, für die Teilbereiche der Eingabemenge, denen die originalen Testfälle zuzuordnen sind, künstliche Daten zu berechnen und damit die Qualität der Wissensbasis zu prüfen, sind die Ursachen für Abweichungen von erwartetem und tatsächlichem Ergebnis eingrenzbar auf die Vorverarbeitung oder die Expertise in der Wissensbasis.

Ein Fehler in der Vorverarbeitung führt dazu, daß das Segmentierungsergebnis nicht den tatsächlichen Zustand korrekt wiedergibt. Obwohl die Wissensbasis im Sinne der Spezifikation korrekt ist, kann das Ergebnis in Bezug auf den realen Fall nicht korrekt sein. Werden verschiedene Vorverarbeitungsmoduln angekoppelt oder unterscheiden sich die Akquisitionsbedingungen der Bilddaten z.B. durch verschiedene Kameratypen, so daß die Darstellungen im Eingabebereich nicht mehr mit der modellierten Expertise übereinstimmen, kommt es zu abweichenden Ergebnissen. Weitere Abweichungen können sich aus unterschiedlichen Expertenschulen ergeben, in denen eine Klassifikation in krank oder gesund bei differierenden Entscheidungsschwellen vermittelt wird oder durch die lokale Patientenpopulation, die bei ähnlichen Datenmustern zu einer entfernter lebenden Population bei einer anderen Schwelle als krank zu bezeichnen ist.

4.4 Systematik der Testfallgenerierung

Geht man davon aus, daß die Fälle aus der Testfallmenge aufgrund bestimmter Merkmale ein Verteilungsmuster im Merkmalsraum bilden, ergeben sich für die Testfallauswahl zwei Ansätze

- die Testfälle gleichmäßig bezüglich bestimmter Merkmale über den Eingabebereich der Wissensbasis zu verteilen, um objektiv alle Bereiche gleichmäßig zu repräsentieren
- die Testfallhäufigkeit lokal zu variieren, um nach subjektiven Erwägungen mehr oder weniger kritische Bereiche hinsichtlich einer korrekten Analyse zu untersuchen

Mit der ersten Maßnahme werden sowohl die Teilmengen von Testfällen, deren Analyse unkritisch ist, als auch die Teilmengen, deren Analyse einen komplexeren Hintergrund in der Methodik und Expertise besitzen, z.B. aufgrund erhöhter Unsicherheiten, in gleichem Maß behandelt. Einerseits führt es zu einem übermäßigen Validierungsaufwand, wenn die Verteilung sehr dicht gewählt wird, um lokal angehäufte Fehler zu erkennen. Andererseits können durch die Grobmaschigkeit der Verteilung in den Bereichen lokaler Anhäufung von Fehlern diese in zu geringem Maß erkannt werden.

Angestrebt wird entsprechend der zweiten Maßnahme eine partiell beliebig dicht wählbare Verteilung von Testfällen für jeden Abschnitt des Eingabebereichs der Wissensbasis. Da im Black-Box-Ansatz der Inhalt der Wissensbasis zwischen den Schnittstellen zum Ein- und Ausgabebereich unbekannt ist, werden in einem Durchgang kritische Bereiche anhand der Verteilung der Analyseergebnisse lokalisiert. Fehlerhinweise werden in Nachbarschaftsbereichen lokalisiert. Diese Bereiche können in weiteren Durchgängen mittels dichter verteilter Testfälle im Nachbarschaftsbereich detaillierter untersucht werden.

Das Ziel der Validierung ist allgemein, die Korrektheit der Analyse durch die Wissensbasis für die spezifizierten Eingaben bis zu einem bestimmten Grad zu garantieren. Um dies zu erreichen, ist es notwendig, für ausgewählte Fälle die Korrektheit der Analyse zu überprüfen und gegebenenfalls Maßnahmen zur Verbesserung oder Eliminierung von Fehlern zu veranlassen. Der Begriff der Korrektheit ist in Bezug auf eine formale Grundlage streng binär zu sehen: das Ergebnis ist korrekt oder das Ergebnis ist nicht korrekt in Bezug auf eine formale Spezifikation. Für die Validierung einer Wissensbasis für die Bildanalyse wird der Begriff relativiert. Die Wissensbasis ist korrekt, wenn die implementierte Version der qualitativen Spezifikation genügt. Für die Feststellung sind geeignete Kriterien zu definieren. Durch die Beschränkungen in Zeit, Wissen und Methoden, die die vollständige Überprüfung und die inhaltliche Übereinstimmung ver- oder behindern, kann der relativierte Begriff der Korrektheit nur approximiert werden.

Es werden künstlich generierte Testfälle eingesetzt, um die Anforderungen, die sich aus gewünschten Verteilungsmustern ergeben, zu erfüllen. Den künstlichen Testfällen liegt eine Generierungsmethode zugrunde, die als Software-Phantom bezeichnet wird. Die Spezifikation soll die Generierung von Instanzen des Software-Phantoms unterstützen. Beispielsweise

ist aus der Spezifikation des Eingabebereichs das Format der Schnittstelle zu entnehmen, aus der funktionalen Abhängigkeit die Struktur einer Teilmenge aus der Testdatenmenge hinsichtlich einer Fragestellung abzuleiten oder aus der Spezifikation des Ausgabebereichs die Art und das Format des Ergebnisses.

Das Verteilungsmuster der Testfälle basiert auf den definierten Merkmalen. Es werden für eine Fragestellung die Testfälle herangezogen, die sich in einem oder mehreren Merkmalen so unterscheiden, daß sie einer vorgegebenen Ordnung entsprechen. Die Ordnung impliziert über die funktionale Abhängigkeit ein Verteilungsmuster der Merkmale der Analyseergebnisse. Das nach der Analyse der Testfälle vorliegende Ergebnismuster wird mit dem spezifizierten Verteilungsmuster verglichen. Abweichungen werden als Fehlerhinweise interpretiert.

Merkmale, die in der Wissensbasis für die Ableitung der Beurteilung eingesetzt werden, wirken sich auf das Analyseergebnis aus. Sie selbst stellen objektive Information dar, d.h. für ihre Berechnung muß kein Beurteilungswissen (Expertise) eingesetzt werden. Variationen anderer Merkmale, die nicht in der Wissensbasis verwendet werden, besitzen keinen Einfluß, wenn sie unabhängig von den Merkmalen sind, die in die Wissensbasis integriert sind. Sind sie nicht unabhängig, ist eine Ordnungsdefinition auf ihrer Basis nur sinnvoll für die Validierung, wenn die Abhängigkeit zu den in der Wissensbasis verwendeten Merkmalen explizit ist. Das Validierungsverfahren setzt die Merkmale, die sich direkt oder indirekt auf Analysen auswirken, zur Definition von A-Ordnungen ein. Sie sind im Merkmalsvektor $(\mu_1, \mu_2, \ldots, \mu_{\mathrm{L}}) \in K$ zusammengefaßt.

Um den Einfluß mehrerer Merkmale zu reduzieren oder gar auszuschließen, unterscheiden sich Testfälle innerhalb einer Untersuchung im idealen Fall nur in einem Merkmal. Dies ist problematisch, wenn statistische Unabhängigkeit des Merkmals zu anderen Elementen des Merkmalsvektors nicht zu gewährleisten ist. Das Problem wird nicht weiter theoretisch betrachtet.

Die Wertdifferenz der variablen Merkmale, insbesondere die Dichte der Testfälle, hängt von der Genauigkeit ab, mit der Bereiche von Ausgaben zu testen sind. Der Ausgabebereich wird umso genauer getestet, je dichter die Nachbarn in der A-Ordnung des Eingabebereichs liegen.

Betrachtet man den Definitionsbereich von E-Ausprägungen eines E-Objektes, existieren E-Ausprägungen, die unter Berücksichtigung der Abhängigkeit zu anderen E-Objekten entfallen, da die Kombination nicht realistisch ist. Für die Validierung werden die Abhängigkeiten berücksichtigt. Sind beispielsweise in der Eingabemenge Darstellungen eines Objektes und Teile des Objektes, die zusammen das vollständige Objekt ergeben, so ist eine Abhängigkeit zwischen den E-Ausprägungen des ganzen Objektes und seiner Teile vorhanden. Die Testdatenmenge ist eine Untermenge aller E-Ausprägungen und der Kombinationen, in der die Abhängigkeiten zwischen E-Objekten berücksichtigt sind. Die zu einem Testfall gehörenden E-Ausprägungen sind durch den gleichen Hochindex bezeichnet. Das gilt auch für die A-Ausprägungen, um die Zusammengehörigkeit der Kombination von E-Ausprägungen zu den A-Ausprägungen über F auszudrücken.

Definition 4.10 *Sei N die Anzahl der E-Objekte, M die Anzahl der A-Objekte, dann ist ein Testfall T_i eine realistische Kombination von jeweils einer E-Ausprägung jedes E-Objektes ausgedrückt durch das Prädikat P.*

$$T_i = \{e_j^i | j = 1, \ldots, N und P = TRUE\}$$

Definition 4.11 *Die Testdatenmenge T enthält alle als Testfälle bezeichneten realistischen Kombinationen von E-Ausprägungen*

$$T = \{T_i | i = 1, \ldots\}$$

Nach der Analyse der ausgewählten Testfälle liegen die zugehörigen A-Ausprägungen zu jedem Testfall vor. Stehen die Testfälle in einer A-Ordnung, stehen die korrespondierenden A-Ausprägungen in der gleichen A-Ordnung. Gegeben sei die funktionale Abhängigkeit F

$$F(e_1^i, e_2^i, \ldots, e_N^i) = F(T_i) = (a_1^i, a_2^i, \ldots, a_M^i)$$

für die Testfälle $T_{p-1}, T_p \text{und} T_{p+1}$, die aus der Testdatenmenge T stammen.

$$\begin{aligned} F(T_{\text{p-1}}) &= (a_1^{\text{p-1}}, a_2^{\text{p-1}}, \ldots, a_M^{\text{p-1}}) && (4.19)\\ F(T_p) &= (a_1^p, a_2^p, \ldots, a_M^p) && (4.20)\\ F(T_{\text{p+1}}) &= (a_1^{\text{p+1}}, a_2^{\text{p+1}}, \ldots, a_M^{\text{p+1}}) && (4.21)\\ & && (4.22) \end{aligned}$$

Die zentrale Aussage als Basis für die Ableitung der Strategie der Testfallgenerierung und Testfallauswertung bezieht sich auf die A-Ordnung:

Sind die E-Ausprägungen der Testfälle T_{p-1} und T_{p+1} 1-Nachbarn von T_p in der A-Ordnung und ist T_{p-1} Vorgänger von T_p und T_{p+1} Nachfolger von T_p

$$e_i^{p-1} \rightarrow e_i^p \rightarrow e_i^{p+1}, \text{ für alle } \quad i = 1, \ldots, N \Rightarrow T_{p-1} \rightarrow T_p \rightarrow T_{p+1} \tag{4.23}$$

dann gilt für die A-Ausprägungen a_1^{p-1}, a_1^p und a_1^{p+1}

$$a_j^{p-1} \rightarrow a_j^p \rightarrow a_j^{p+1}, \text{ für alle } \quad j = 1, \ldots, M \tag{4.24}$$

Der Merkmalsvektor ist unabhängig vom realen oder künstlichen Ursprung der Ausprägung. Originale Patientendaten sind im Merkmalsraum nicht von künstlich generierten Testfällen zu unterscheiden. Originale und künstliche Testfälle, die die gleichen Merkmalswerte besitzen, referieren ununterscheidbar einen Punkt im Merkmalsraum. Eine A-Ordnung kann aus realen und künstlich erzeugten Fällen bestehen.

Eine Methode zur Generierung künstlicher Testfälle für ein Validierungsverfahren auf der Basis angeordneter Ausprägungen wird durch die Struktur und die Abhängigkeiten des Eingabebereichs und die Merkmale bestimmt. Um anhand der Bedingungen für eine A-Ordnung die geeigneten Fälle zu generieren, muß der Testfall durch den Merkmalsvektor eindeutig beschrieben sein.

4.5 Fehlerdetektion

Durch die Ordnung der A-Ausprägungen, die über F aus der Ordnung der E-Ausprägungen resultiert, ergibt sich ein Muster der Merkmalswerte $A_m.K(a_m^p) = \kappa_m$ eines ausgewählten Merkmals K aus der Ergebnisstruktur der A-Auspägungen a_m^p des A-Objektes A_m. Welches Muster für ein A-Objekt zu erwarten ist, wird über die A-Ordnung und F ermittelt. Stimmt das aus den Analyseergebnissen gebildete Muster nicht mit dem erwarteten Muster überein, liegen die Ursachen in der Abweichung der Implementierung der Wissensbasis von der Spezifikation.

Während der wissensbasierten Analyse werden die A-Ausprägungen auf der Basis einer Menge von Parametern, im folgenden auch Kriterien genannt, bestimmt. Für ein A-Objekt können mehrere Kriterien relevant sein. Im folgenden wird speziell das Merkmal Gewichtung betrachtet. Die Bedeutung von Gewichtung im klassifizierenden wissensbasierten System ist hier ein Ausdruck für die Übereinstimmung des aktuellen Falles mit der Klasse, die durch das A-Objekt gegeben ist. Andere Merkmale, z.B. Text, werden in diesem Rahmen nicht betrachtet, da sie für die Validierung des wissensbasierten Bildanalysesystems (Kapitel 5) nicht relevant sind.

Unterscheiden sich die E-Ausprägungen in einem Merkmal und ist dieses Merkmal direkt oder indirekt durch die Entscheidungskriterien in der Wissensbasis relevant für die A-Objekte, wird ein Einfluß auf die A-Ausprägungen in Form eines spezifizierbaren Musters erwartet. Es entsteht auf der Basis der monotonen Änderung dieses Merkmalswertes in Form einer A-Ordnung. Die entsprechenden Werte zum Merkmal Gewichtung der A-Objekte genügen dann ebenfalls partiell bestimmten Monotoniebedingungen.

Eine monotone Veränderung eines Merkmals im Eingabebereich ist durch eine A-Ordnung mit der Sortiervorschrift, die Testfälle nach aufsteigenden bzw. absteigenden Merkmalswerten anzuordnen, gegeben. Es ist ein Starttestfall festzulegen, d.h. ein Testfall, der keinen Vorgänger in der A-Ordnung besitzt. Ausgehend vom Starttestfall soll der Wert des ausgewählten Merkmals beim Nachfolger größer bzw. kleiner sein als der Merkmalswert des aktuellen Testfalls.

Die A-Objekte stellen im wissensbasierten Bildanalysesystem die Interpretation des segmentierten Bildes dar. Sie werden durch die Erfüllung eines Kriteriums, auf dem sie basieren, mehr oder weniger bestätigt. Den Grad der Erfüllung beschreibt die Gewichtung. Da bis auf eine feste Teilmenge der Merkmalsmenge, die als variable Merkmale im Wert verändert werden, alle anderen Merkmale im Wert konstant sind, muß die A-Ordnung von

E-Ausprägungen mit monoton steigenden Merkmalswerten über dem gesamten Werteintervall zu einem Verlauf der Gewichtung führen, der dem Eindruck

$$\text{untypisch} \rightarrow \text{unsicher} \rightarrow \text{typisch} \rightarrow \text{unsicher} \rightarrow \text{untypisch}$$

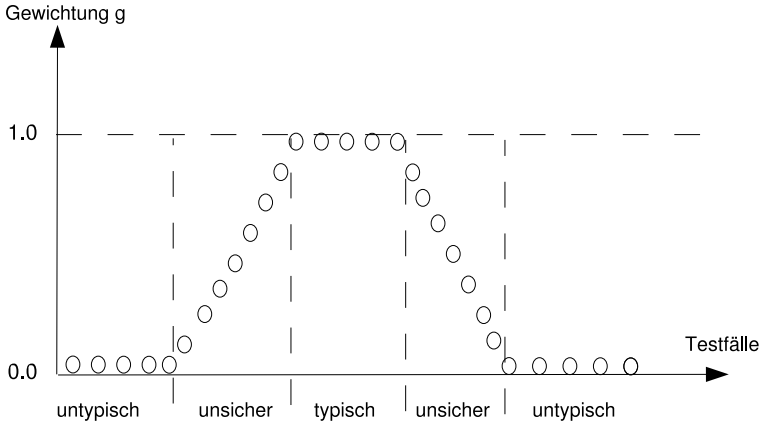

Abbildung 4.5: Typisches Beispiel des erwarteten Gewichtsverlaufs eines A-Objektes. Die Gewichtung ist ein Merkmal des Ergebnisses der Analyse von Testfällen in der A-Ordnung ansteigener Merkmalswerte, dargestellt in der Achse **Testfälle**. **o** repräsentiert die Gewichtung eines Testfalls. Im unsicheren Bereich wurde die Veränderung im Beispiel linear gewählt.

entspricht (Abbildung 4.5). Ausnahmen bilden Spezialfälle, beispielsweise bei denen die Gewichtung am Anfang des Verlaufs den Eindruck typisch vermittelt und weiter über unsicher nach untypisch verläuft. Ist die Gewichtung $g \in [0.0, 1.0]$, entsteht der Eindruck untypisch bei einer Gewichtung von 0.0, während die Gewichtung 1.0 den Eindruck typisch vermittelt. Die Gewichtungen zwischen 0.0 und 1.0 drücken die Unsicherheit für eine eindeutige Aussage aus. Ist das Gewicht $g \in [g_{\min}, g_{\max}]$, dann gilt für

$$\text{untypisch:} g = g_{\min}, \text{unsicher:} g \in (g_{\min}, g_{\max}), \text{typisch:} g = g_{\max}$$

Der Grundtyp des Verlaufsmusters ist monoton wachsend in den aneinander angrenzenden Bereichen von untypisch über unsicher bis typisch. Der Verlauf ist monoton fallend in den aneinander angrenzenden Bereichen von typisch über unsicher bis untypisch.

Je nach Wahl des Starttestfalls, der Anzahl der Testfälle, ihres Abstands D, eventuell vorhandener Abhängigkeiten der variierten Merkmale zu den konstanten Merkmalen und der funktionalen Beschreibung, ergeben sich Spezialformen des Verlaufs, die abschnittsweise dem Grundtyp entsprechen (Abbildung 4.6).

Ist der Starttestfall so gewählt, daß sein Merkmalswert **typisch** für das A-Objekt ist

($g = g_{\max}$), dann wird der Verlauf der Gewichtung im typischen Bereich beginnen. Führt der Merkmalswert des Starttestfalls zu dem Eindruck **untypisch** für das A-Objekt ($g = g_{\min}$), wird es entweder nachfolgend zu einem Gewichtungsverlauf nach dem Grundtyp oder zu keiner weiteren Gewichtungsveränderung kommen. Ähnlich verhält es sich mit einem Beginn im unsicheren Bereich. Welchem Abschnitt das Ende des Verlaufs zuzuordnen ist, hängt von der Wahl des Endtestfalls in der A-Ordnung ab.

Abbildung 4.6 zeigt aus dem Grundtyp abgeleitete Spezialtypen des Gewichtsverlaufs. Es sind idealisierte Darstellungen in Bezug auf Dichte und Form. Die Dichte hängt von der Anzahl der Testfälle ab, die für eine Testphase ausgewählt wurden. Die Linearität oder Nichtlinearität des Verlaufs wird für die Validierung der Wissenbasis als Information nicht herangezogen. Untersucht wird lediglich die Einhaltung abschnittweiser Monotonie. Tritt der Fall auf, daß ein Verlauf zwischen den ansteigenden und abfallenden Abschnitten nicht den Wert $g_{\max}$ erreicht, sind weitere Schritte notwendig, beispielsweise um festzustellen, ob die Ursache in der Wahl der Belegungen der konstant gewählten Merkmale liegt, und ob es eine Wertekombination der konstanten Belegungen gibt, bei denen der Verlauf $g_{\max}$ erreicht.

Definition 4.12 *Eine Testphase umfaßt die Menge der Aktivitäten, die zur Ermittlung eines Verlaufsmusters des ausgewählten Merkmals ausgewählter A-Objekte notwendig sind. Zu den Aktivitäten zählen die Definition der A-Ordnung, die Auswahl der Testfälle (Starttestfall, Dichte, Endtestfall), die Analyse der Testfälle durch das Validierungsobjekt, die Auswahl der A-Objekte und die Verlaufsdarstellung der Merkmalswerte für jedes ausgewählte A-Objekt.*

Definition 4.13 *Ein Testblock umfaßt mehrere Testphasen, die sich nur durch die Variation eines zweiten Merkmals unterscheiden.*

Der Ansatz zur Fehlerdetektion basiert auf der Annahme, daß ein Fehler, sofern er bei der Analyse aktiviert wird, zu einer Abweichung des erwarteten abschnittweise steigend oder fallend monotonen Gewichtungsverlaufs führt (Abbildung 4.7). Ist die Wissensbasis durch den Merkmalsvektor vollständig abgedeckt, d.h. sind alle die Entscheidungsfindung beeinflussenden Merkmale im Merkmalsvektor, wird die für den Fehler typische Erscheinung durch die Analyse mindestens einer A-Ordnung aktiviert. Die Abweichungen sind größtenteils als Oszillationen erkennbar, deren Anzahl und Amplitude von der Fehlerhäufigkeit, der Fehlerart und ihrem Wirkungsgrad abhängen. Ein Fehler kann auch zu einem Gewichtungsverlauf führen, der über ein größeres Intervall konstant ist. Aus der Oszillation oder dem konstanten Verlauf kann nicht auf Art und Ort des Fehlers geschlossen werden. Die Erscheinungen werden nur als Hinweise auf Fehler interpretiert. Für die Lokalisierung sind weitere MaÃnahmen (Kapitel 4.6) zu ergreifen.

Der Bereich, in dem das variierte Merkmal bei der Entscheidungsfindung durch die Belegung der konstant gewählten Merkmale dominiert wird, enthält möglicherweise eine

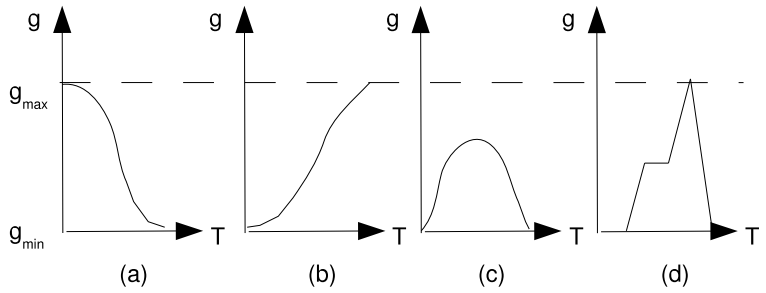

Abbildung 4.6: Aus dem Grundtyp abgeleitete Spezialtypen des Gewichtungsverlaufs von g für die Testfälle T, die der Monotoniebedingung genügen.
(a) monoton fallende Gewichtung
(b) monoton steigende Gewichtung
(c) Stauchung des Grundtyps mit $g < g_{max}$ für alle g
(d) linear monoton steigende und fallende Gewichtung mit einem konstanten Bereich im steigenden Zweig und sehr kleinem typischen Bereich

verdeckte Fehlerquelle. Durch Variationen der konstanten Belegungen in verschiedenen Testdurchläufen können die Fehlerhinweise deutlich werden.

Der in Abbildung 4.7a dargestellte Gewichtsverlauf zeigt typische Fehlerhinweise. Es handelt sich um einen deutlichen Einbruch im typischen Bereich (1), über in ihrer Amplitude kleine, aber in der Anzahl häufige Oszillationen im unsicheren Bereich des fallenden Zweiges (2) bis zu einem Ausbruch im untypischen Bereich (4). Trotz der Erhaltung der Monotoniebedingung kann ein konstanter Bereich mit positiver Gewichtung am Ende des fallenden Zweiges (3) verdächtig sein, wenn dies nicht spezifiziert ist.

Um möglichst alle Fehler aufzudecken, muß für jeden der Fehler mindestens ein Repräsentant in einer A-Ordnung im Eingabebereich existieren, der eine auffällige Erscheinung im Gewichtungsverlauf verursacht. Werden die Testdurchläufe ungeeignet festgelegt, können Fehler völlig unentdeckt bleiben. Die Abbildung 4.7b zugrundeliegende Testphase unterscheidet sich von der in Abbildung 4.7a durch eine geringere Anzahl von Testfällen mit geringerer Dichte. Es wurde nur jede zweite Ausprägung aus (a) betrachtet. Dabei entfielen drei auffällige Erscheinungen (bei (1) und (2)).

Ein Ansatz, die Rate aufgedeckter Fehler zu erhöhen, ist, Testfälle mit geringem Abstand $D < \varepsilon$ auszuwählen, wobei ε die Schranke ist, bei der alle Fehler aufgedeckt werden. Dieser Ansatz ist nicht praktikabel. Es ist abzuwägen, welche Bereiche durch dichte Abdeckungen mit Testfällen zu untersuchen sind, und welche weniger kritisch sind, so daß auf einen hohen Aufwand verzichtet werden kann. Die Abwägung ist eine qualitative Entscheidung auf der Basis subjektiven Wissens, beispielsweise durch Entwickler oder Bereichsexperten.

Um Hinweise durch Oszillationen zu lokalisieren, wird der Verlauf der Gewichtungen

von links nach rechts anhand der Vorzeichen der Gewichtungsänderungen (Abbildung 4.7) verfolgt. Beginnt der Verlauf vor dem typischen Bereich, wird eine Zunahme der Gewichtungen erwartet, dargestellt als (+). Der typische Bereich enthält keine Gewichtungsänderung. Geht der Verlauf über den typischen Bereich hinaus, ist die Gewichtungsänderung negativ und wird als (-) dargestellt. Beginnt der Verlauf im typischen Bereich, wird nur eine Abnahme der Gewichtungen erwartet. Beginnt und endet der Verlauf in einem der zwei unsicheren Bereiche oder im typischen Bereich, liegt keine Gewichtungsänderung vor. Der Fall eines Verlaufs mit einer Abnahme und darauffolgender Zunahme der Gewichte wird nicht betrachtet. Er tritt beispielsweise bei negierten Aussagen auf. Hier wird von der Annahme ausgegangen, daß durch die vollständige Abdeckung des Diagnosespektrums durch A-Objekte für jeden Fall eine zutreffende positive Aussage abgeleitet werden kann.

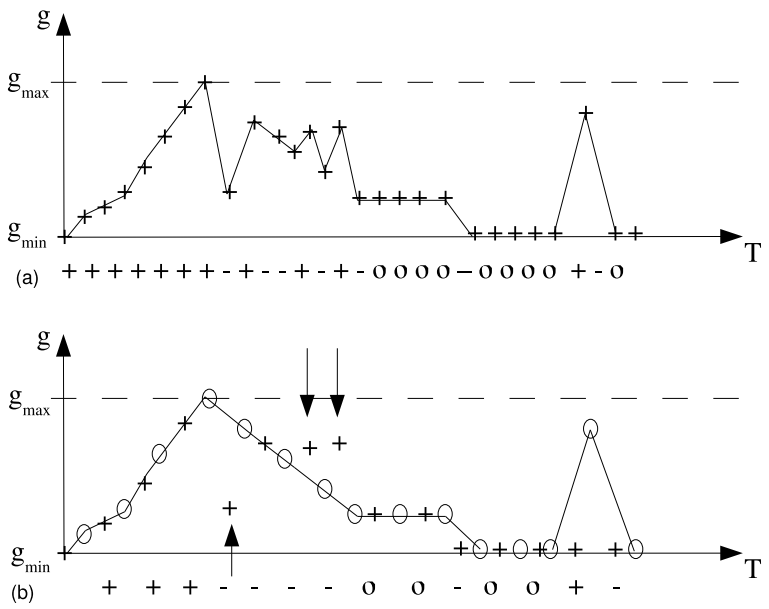

Abbildung 4.7: Verlauf von Gewichtungen + in (a) und o in (b). Die diskreten Werte zur Verdeutlichung des Verlaufstyps sind verbunden. Die Folgen von +,- und 0 unter der T-Achse sind die Vorzeichen der Bewertungsänderung zwischen direkt aufeinanderfolgenden Gewichtungen von links nach rechts. Der Verlauf in (a) besitzt 4 Fehlerhinweise, an den Positionen (1),(2) und (4) als Beispiel für Oszillationen und (3) frr konstantes Verhalten. In (b) basiert der Verlauf o auf einer A-Ordnung, die nur aus jedem zweiten Testfall aus (a) besteht. Die entfallenen Gewichte aus (a) sind durch ein + gekennzeichnet. Bei der Verdünnung entfielen 3 durch Pfeile gekennzeichnete Hinweise.

Der steigende und fallende Verlauf darf bei der Suche nach Oszillationen durch konstante Bereiche unterbrochen sein, die nicht als Fehlerhinweise interpretiert werden. Die

zulässigen Vorzeichenfolgen umfassen nach Ausschluß der negierten Aussage drei erwartete Verlaufsformen. Die eckigen Klammern fassen die Änderungsbezeichnungen zusammen, die in beliebiger Anordnung aufeinander folgen dürfen. Die Zeichen sind durch — getrennt. Die Häufigkeit des Auftretens ist mindestens einmal ($*$) oder beliebig oft ($\wedge$) beschrieben:

$(* : 1, \ldots, n), (\wedge : 0, \ldots, n)$
1. Fall:$[+ * |0\wedge] * [- * |0\wedge]\wedge$ z.B. ++++-0-00
2. Fall:$[- * |0\wedge]*$ z.B. 000-0—0
3. Fall:$[0]*$

Um einen Fehlerhinweis zu detektieren, wird geprüft, ob die erwarteten Vorzeichenfolgen gestört sind. Jede Gewichtung korrespondiert entsprechend ihrer Position im Verlauf mit dem Testfall an der gleichen Position in der A-Ordnung. Ob der Fehler durch den hinweisgebenden Testfall aktiviert wurde oder durch benachbarte Fälle, wird bei der Fehlerlokalisation untersucht. Das Verfahren zur Fehlerdetektion schränkt den Suchraum für die Fehlerlokalisation ein, liefert der Suche einen Startpunkt.

Der Bereich, in dem das variierte Merkmal bei der Entscheidungsfindung durch die Belegung der konstanten Merkmale dominiert wird, ist als mögliche verdeckte Fehlerquelle zu berücksichtigen.

Die bisher durchgeführte Fehlerdetektion beschränkt sich auf die Untersuchung eines einzigen Ausgabeobjektes. Bei diesem Ansatz ist die Bewertung der Entscheidungsfähigkeit des Systems nicht möglich. Entscheidungsfähigkeit bezieht sich auf die Eigenschaft des Systems, eine sichere Aussage zu treffen. Eine sichere Aussage ist mit der maximalen Bewertung 1.0 belegt, liegt demnach im typischen Bereich des Gewichtungsverlaufs. Angenommen, es existiert eine Teilmenge von A-Objekten, die für bestimmte Testfälle, die nicht sicher zu beurteilen sind, mögliche alternative Aussagen darstellen. Um die Unsicherheit auszudrücken, daß keine einzelne Aussage durch eine höhere Gewichtung hervorzuheben ist, ist eine gleiche positive Gewichtung $g < 1.0$ für die alternativen Aussagen sinnvoll. Wird allerdings in dieser Situation den alternativen A-Objekten die gleiche sichere Bewertung 1.0 zugeordnet, ist es unsinnig. Tritt der Fall auf, ist es ein Hinweis auf eine redundante Wissensdefinition in der Wissensbasis, die schlechte Expertise ausdrückt. Die Redundanz ist in den Gewichtungsfunktionen und ihren Parametern zu vermuten oder wird durch die Regeln, die auf den Gewichtungsfunktionen basieren, erzeugt.

Die Entscheidungsfähigkeit wird nicht durch gleiche Bewertungen im unsicheren Bereich ($0.0 < g_1 = g_2 < 1.0$) in Frage gestellt. Im Gegenteil soll gerade hier die Unmöglichkeit der sicheren Zuordnung verdeutlicht werden. Die Übereinstimmung der Gewichtung liefert dabei deutliche Hinweise auf die Lokalisierung der Eingabeobjekte im Grenzbereich zwischen Aussagen. Die Höhe der übereinstimmenden Bewertung wird vom AusmaÃ der Überlappung der Gewichtungsfunktionen beider Aussagen mit beeinflußt. Ob eine gemeinsame Gewichtung nahe 1.0 bereits als Entscheidungsunfähigkeit zu betrachten ist, ist eine jeweils für das System zu treffende spezielle Entscheidung. Dies sollte bereits in der Spezi-

fikation berücksichtigt werden.

Um die Bereiche zu detektieren, in denen die Wissensbasis entscheidungsunfähig ist, werden für einen Testfall die Gewichtungen der alternativen A-Objekte verglichen. Sind zwei oder mehr A-Objekte mit der sicheren Bewertung 1.0 belegt, können die Ursachen anhand des zugehörigen Testfalles untersucht werden.

Ein weiterer durch die lokale Betrachtung eines A-Objekts nicht sichtbarer Fehler ist die Unvollständigkeit von Wissen, die die Monotonieeigenschaften der Gewichtungsverläufe nicht verletzt. Sie wird beispielsweise durch Gewichtungsfunktionen bei den genannten alternativen A-Objekten verursacht. Alternative A-Objekte besitzen Gewichtungsfunktionen, die auf demselben Kriterium basieren, aber in verschiedenen Teilintervallen des Kriteriumswertes positiv definiert sind. Existiert nun ein Teilintervall, in dem für keines der alternativen A-Objekte die Gewichtung positiv definiert ist, ist das Wissen unvollständig. Dann existieren Testfälle, die bei allen alternativen A-Objekten nur eine Gewichtung 0.0 erhalten, also im untypischen Bereich liegen. Soll nach der Spezifikation das Analysespektrum vollständig sein, sollen also die alternativen A-Objekte das gesamte Wertespektrum des Kriteriums abdecken, dann sind die Analyseverläufe zu untersuchen, bei denen die Testfälle in den alternativen A-Objekten keine positive Gewichtung erhalten.

Sind die E-Ausprägungen eine Manifestation unrealistischer Gegebenheiten, beispielsweise als Ergebnis einer fehlerhaften Bildvorverarbeitung, ist die durchgehende Gewichtung mit 0.0 korrekt. Ist die Wissensbasis vollständig bezüglich der Abdeckung des möglichen Ausgabespektrums durch die A-Objekte, existiert ein A-Objekt, das mit der Aussage undefiniert korrespondiert. Dieses A-Objekt muß in den Fällen unrealistischer Gegebenheiten die sichere Gewichtung 1.0 erhalten.

Decken die A-Objekte das gesamte Analysespektrum ab, können lokal für ein A-Objekt die Fehlerhinweise mit Ausnahme des Hinweises auf Unvollständigkeit oder Redundanz des Wissens durch die Verfolgung des Gewichtungsverlaufs erkannt werden. Sind durch die Unvollständigkeit der A-Objekte Testfälle nicht zu klassifizieren, können Fehlerhinweise nicht systematisch erkannt werden. Die spezifizierte Vollständigkeit ist Voraussetzung für die Vollständigkeit der Fehlerüberprüfung. Die vollständige Fehlersuche wird durch die Betrachtung von zwei Arten von Kontext gestützt. Ein Kontext bezieht sich ausgehend von einem A-Objekt auf die Nachbarschafts- und Anordnungseigenschaft der Testfälle. Der zweite Kontext bezieht sich ausgehend von einem Testfall auf die Gewichtung der zugehörigen alternativen A-Objekte.

4.6 Fehlerlokalisation

Die detektierten Fehlerhinweise verweisen direkt oder indirekt auf die fehleraktivierenden Testfälle. Kandidaten für die Fehlerlokalisierung sind im Fall eines indirekten Fehlerhinweises die n-Nachbarn ($n \geq 1$), wie es in Abbildung 4.8 dargestellt ist. Welcher Testfall für die Überprüfung des Hinweises auszuwählen ist, ist interaktiv zu entscheiden. Der Ansatz

der Automatisierung des Verfahrens zur Fehlersuche wird nicht weiter verfolgt. Ist der fehleraktivierende Testfall ermittelt, ist zur Optimierung der Fehlersuche die Erfahrung der Entwickler, ihre detaillierte Kenntnis der implementierten Wissensbasis, nützlich.

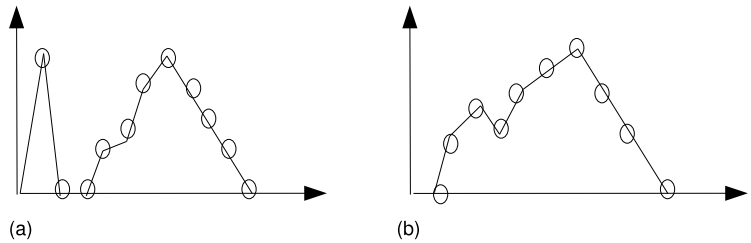

Abbildung 4.8: Gewichtungsverläufe mit Fehlerhinweis (2) und dem Gewicht (1), das mit dem fehleraktivierenden Testfall korrespondiert: (a) Der Fehlerhinweis wird durch den vorhergehenden 3-Nachbarn des fehleraktivierenden Testfalls (2) erzeugt, (b) Fehleraktivierender und hinweisgebender Testfall sind identisch

Obwohl das Verfahren zur Fehlerdetektion in Abbildung 4.8a durch die Gewichtung des Testfalls an Position (2) einen Fehlerhinweis erzeugt, erscheint es sinnvoller, den Testfall zu untersuchen, der den Peak am Beginn des Verlaufs erzeugt, wenn nach der Spezifikation der positive Verlauf erst später beginnen soll. Ist trotz zusätzlicher Informationen die Auswahl eines Testfalles als Fehlerrepräsentant nicht möglich, ist die in Frage kommende Gruppe zu überprüfen. Obwohl der Suchaufwand durch Erhöhung der Anzahl der zu betrachtenden Testfälle vergrößert wird (wenn der Fehler nicht schon bei der ersten Fallüberprüfung gefunden wurde), ist doch die Fehlersuche zunächst kanalisiert.

Für die Lokalisierung des Fehlers wird die Wissensbasis nicht mehr als Black-Box betrachtet. Anhand des oder der Testfallkandidaten für die Fehleraktivierung werden die während der Analyse durchlaufenen deklarativen Strukturen, aktivierten prozeduralen Anteile und generierten Werte überprüft. Die Suche beschränkt sich auf die Systembereiche, die für die Berechnung der Gewichtung der auffälligen Ausprägungen durchlaufen wurden. Andere aktivierte Bereiche werden ausgeklammert.

Wird eine A-Ausprägung im Gewichtungsverlauf auffällig, kann ausgehend von ihr der Einstieg in die Wissensbasis gefunden werden. Die auffällige A-Ausprägung korrespondiert mit der Instanz des Objektes der Wissensbasis im Schnittstellenbereich der Ausgabe, da die A-Ausprägung das Ergebnis einer Merkmalsextraktion aus der Ergebnisstruktur eines Objektes ist. Das Objekt der Wissensbasis korrespondiert wiederum mit dem A-Objekt. (Abbildung 4.9). Die E-Objekte und die E-Ausprägungen korrespondieren in gleicher Weise mit den Objekten und Instanzen der Wissensbasis im Schnittstellenbereich der Eingabe. Über diese Abhängigkeiten kann der Analyseverlauf untersucht werden.

Die während einer Analyse eingesetzten oder erzeugten Elemente der Wissensbasis, die zur A-Ausprägung führen, können in Langzeit-, Kurzzeit- und temporäres Kurzzeitwissen

unterteilt werden

Langzeitwissen: Deklaratives und prozedurales Wissen, das nach Beendigung der Analyse und nach Beendigung des Prozesses (Systemprozeß) noch vorliegt (z.B. auf der Festplatte). Über das Benutzer-Interface kann vor und nach der Analyse darauf zugegriffen werden.

Kurzzeitwissen: Instanzen, die nach Beendigung der Analyse vorliegen (im Hauptspeicher). Über das Benutzer-Interface kann darauf zugegriffen werden. Mit der Beendigung des Prozesses wird es gelöscht.

temporäres Kurzzeitwissen: Temporäre Speicher- und Variablenbelegungen, die nur während der Analyse vorliegen und auch nicht protokolliert werden.

4.7 Qualitätsbeschreibung

Von der Qualitätsbeschreibung wird die Vermittlung eines Eindrucks von der Leistungsfähigkeit der Wissensbasis erwartet. Die Information, die die Qualitätsbeschreibung liefert, soll eine Grundlage für die Entscheidung zur Fortführung oder zur Beendigung des Validierungsverfahrens sein oder potentiellen Anwendern Hinweise auf die Richtigkeit der Analyseergebnisse liefern. Die Qualitätsbeschreibungen als Ergebnis qualitativer Methoden sind nur eingeschränkt nutzbar. Besteht beispielsweise die qualitative Methode darin, einen Bereichsexperten zu bitten, für ausgewählte Verläufe die Übereinstimmung des Verlaufsmusters der Gewichtungen mit dem erwarteten Muster abzuwägen, ist der Nutzen sowohl für die Validierung als auch für Anwender fraglich. Eine quantitative Methode mit reproduzierbaren Werten liefert konkretere Hinweise.

Im folgenden werden Qualitätsmaxiße für die Beschreibung des Verlaufsmusters relativ zum erwarteten Verlauf und für die Richtigkeit der Analysen eingeführt. Wissensbasisinterne Informationen werden nicht verwendet, sondern ausschlieÃlich die Informationen aus dem Ein- und Ausgabebereich und aus den A-Ordnungen. Um eine Qualitätskenngröße als Qualitätsmaß bezeichnen zu können, muß sie verschiedene Gütekriterien erfüllen. Itzfeld et al. [ITZ84] nennen die folgenden Gütekriterien für ein Software-Qualitätsmaß:

Objektivität: Ein Maß ist objektiv, wenn kein subjektiver Einfluß bei der Messung durch die messende Person möglich ist.

Zuverlässigkeit: Ein Maß ist zuverlässig, wenn bei der Wiederholung der Messung unter denselben Meßbedingungen dieselben Meßergebnisse erzielt werden.

Validität: Ein Maß heißt valide, wenn die Meßergebnisse einen eindeutigen und unmittelbaren Rückschluß auf die Ausprägung der Qualitätseigenschaft zulassen.

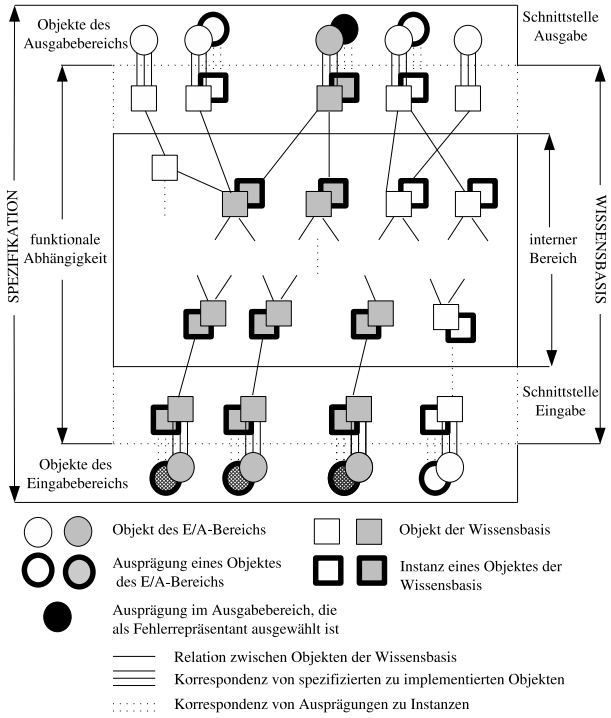

Abbildung 4.9: Überblick über Objekte und Ausprägungen des Ein- und Ausgabebereichs und die Beziehungen zu den Objekten und Instanzen der Wissensbasis. Die gesamten bei der Analyse berührten Objekte besitzen eine Instanz, die in Bezug zu einer A-Ausprägung steht. Die für die Fehlerlokalisation zu prüfenden Teilstrukturen sind schraffiert dargestellt. Ausgehend von der hinweisgebenden A-Ausprägung werden über die Korrespondenz zu der Instanz und von dort über die internen Relationen die internen Bereiche bis zum Eingabebereich abgesucht, die potentielle Fehlerquellen darstellen.

Normierung: Um ein Maß zu normieren, ist eine Skala notwendig, auf die die Meßergebnisse eindeutig abbildbar sind. Ein Maß wird als normiert bezeichnet, wenn es eine Vergleichbarkeitsskala gibt.

Vergleichbarkeit: Ein Maß ist vergleichbar, wenn es zu anderen Maßen in Relation gesetzt werden kann.

Ökonomie: Ein Maß ist ökonomisch, wenn die Messung mit geringen Kosten verbunden ist.

Nützlichkeit: Ein Maß ist nützlich, wenn mit der Messung praktische Bedürfnisse erfüllt werden.

Mit der Messung ist der Wunsch nach der Aufdeckung von Stärken und Schwächen

verbunden, unabhängig von subjektiven Einflüssen auf die Messung. Die Verbesserung der Qualität und die Vertrauensbildung für das applizierte System sind zwei grundlegende Motivationen für den Aufwand der Qualitätsmessung. Dahinter stehen die zwei Gruppen der Entwickler und Benutzer. Für die Entwicklung bedeutet Qualitätsmessung, Fehler abzuschätzen, um sie entweder zu eliminieren oder als Restfehler bei erzielter Qualitätsvorgabe zu akzeptieren. Für die Benutzergruppe bedeutet es, Informationen über die Auswirkungen der Restfehler zu erhalten, die eine Akzeptanzbeurteilung relativ zu alternativen Verfahren unterstützt. Hier werden weitere externe Informationen eingesetzt.

Während der Validierungsphase und für den Einsatz im Anwendungsbereich ist objektive Information über die Qualität der Ergebnisse der wissensbasierten Analyse nachgefragt. Als objektive Information wird die Messung von Qualität nach objektiven Verfahren mit reproduzierbaren Ergebnissen verstanden, im Gegensatz zur subjektiven Information, die auf dem Eindruck eines menschlichen Experten basiert, der die Ergebnisse zu beurteilen hat. Subjektive Information setzt gleiche oder ähnliche Erfahrungen voraus, um sie adäquat interpretieren zu können. Sie enthält bereits persönliches Beurteilungswissen. Objektive Information enthält beurteilungsfreie Werte. Sie basiert auf Verfahren, die ohne gemeinsame Erfahrungsgrundlagen nachvollziehbar sind.

Im folgenden Kapitel 4.7.1 wird die Messung der Eigenschaft der Homogenität eingeführt. Die Homogenität drückt die Übereinstimmung der implementierten Version zur Spezifikation auf der Basis der Gewichtungsverläufe aus. Das darauf folgende Kapitel 4.7.2 behandelt die Messung der Zuverlässigkeit der Wissensbasis unabhängig vom Validierungsverfahren auf der Basis eines Qualitätsmodells.

4.7.1 Spezifikationsbezogene Qualitätsbeschreibung

Die im Merkmalsraum eine A-Ordnung bildenden Testfälle werden für die spezifikationsbezogene Qualitätsbeschreibung verwendet. Es handelt sich um die in den Testdurchläufen analysierten Testfälle. Die Abweichung eines in einer Testphase erzeugten Verlaufsmusters von dem erwarteten Muster ist im Kontext der Testphase durch die Oszillationen gekennzeichnet. Der Parameter Homogenität ist ein Maß für die Oszillationen in einem Verlauf.

Die Homogenität wird anhand der Gewichtungsverläufe von Testfällen ermittelt, die in einer A-Ordnung sortiert sind, also im Merkmalsvektor in einer festen Anzahl von Elementen variieren, während alle anderen Elemente konstant sind. Die A-Ordnung der Testphase besteht aus τ Testfällen T_i. i ist der Positionsindex des Testfalls in der A-Ordnung

$$T_1 \rightarrow T_2 \rightarrow \cdots \rightarrow T_\mathrm{i} \rightarrow \cdots \rightarrow T_\tau$$

Die funktionale Abbildung F eines Testfalles T_i resultiert in der Menge der Ausprägungen der M A-Objekte, deren Merkmal K = Gewichtung weiter betrachtet wird.

$$F(T_\mathrm{i}) = (a_m^i | m = 1, \ldots, M) \tag{4.25}$$

$$A_m.\text{Gewicht}(a_m^i) = g_m^i \in [g_{\min}, g_{\max}], m = 1, \ldots, M \tag{4.26}$$

Der Gewichtungsverlauf des Ergebnismusters der Testphase für jedes A-Objekt entspricht der A-Ordnung der korrespondierenden Testfälle

$$g_m^1 \rightarrow g_m^2 \rightarrow \cdots \rightarrow g_m^i \rightarrow g_m^\tau \tag{4.27}$$

Zur Bewertung der Monotonieeigenschaft wird ein Maß für die Oszillationen, die Homogenität H eingeführt. Eine Oszillation besteht aus zwei Werteänderungen mit entgegengesetzten Vorzeichen, die innerhalb eines Verlaufs aufeinanderfolgen. Innerhalb eines Verlaufs, der der Monotonieeigenschaft genügt, ist eine Oszillation zulässig, nämlich der Anstieg in den typischen Bereich, gefolgt von dem Abfall in den untypischen Bereich. Das Maß liefert den Wert 1.0 im idealen Fall, wenn die Monotonieeigenschaft voll erfüllt ist. Um die Spezialtypen des Gewichtsverlaufs (Kapitel 4.5) mitzuerfassen, werden zwei weitere Gewichtungen eingeführt, die zu der erlaubten Oszillation führen, ohne die Monotonieeigenschaft zu stören

$$g_m^0 = g_m^{\tau+1} = g_{\min} \tag{4.28}$$

Die Homogenität $H(m)$ eines Gewichtungsverlaufs des m-ten A-Objektes analog der A-Ordnung der korrespondierenden τ Testfälle ist dann

$$H(m) = \begin{cases} 1 & , \quad wenn \ \underset{i=1,\ldots,\tau}{max}(g_m^i) = g_{\min} \\ g_{\text{norm}} * \sum_{i=0}^{\tau} |g_m^i - g_m^{i+1}| & , \quad sonst \end{cases} \tag{4.29}$$

$$H(m) = H_1(m), \text{wenn } g_{\text{norm}} = g_{\text{norm1}} \tag{4.30}$$

$$H(m) = H_2(m), \text{wenn } g_{\text{norm}} = g_{\text{norm2}} \tag{4.31}$$

$$g_{\text{norm}} = g_{\text{norm1}} = \frac{1}{2 * (\underset{i=0,\ldots,\tau+1}{max} g_m^i - \underset{i=0,\ldots,\tau+1}{min} g_m^i)} \tag{4.32}$$

Damit gilt für g_{norm1}

$$g_{\text{norm}} = g_{\text{norm1}} = \frac{1}{2 * (g_{\max} - g_{\min})} \tag{4.33}$$

Eine Variante ist die Normierung auf das gesamte Intervall von g

$$g_{\text{norm}} = g_{\text{norm2}} = \frac{1}{2 * (g_{\max} - g_{\min})} \tag{4.34}$$

Wird über den Faktor g_{norm1} der Ausdruck auf die Amplitude des Verlaufs normiert, ist $H_1 = 1.0$, wenn außer der zulässigen Oszillation keine weiteren Oszillationen auftreten. Wird im gesamten Verlauf der maximal mögliche Wert $g_{\max}$ nicht erreicht, hat es keinen Einfluß auf die Homogenität. Wird über den Faktor g_{norm2} normiert, enthält der Wert der Homogenität H_2 zusätzlich die Information über die Ausnutzung des Werteintervalls. Die Bedeutung dieses Wertes ist nur relativ zu H_1 zu sehen.

4.7.2 Zuverlässigkeitsmessung im medizinischen Anwendungsbereich

Die Zuverlässigkeit, mit der eine Wissensbasis richtige Aussagen generiert, pathologische und normale Fälle erkennt, ist eine entscheidende Information, um beispielsweise die Effizienz im Vergleich mit alternativen Verfahren aufzuzeigen. Im medizinischen Bereich werden in der Regel die Qualitätsmaße Sensitivität und Spezifität herangezogen, um die Zuverlässigkeit zu beschreiben.

Um diese Maße für eine Wissensbasis zu übernehmen, müssen die spezifischen Bedingungen der Wissensbasis benannt werden. Es wird hier von einer klassifizierenden Wissensbasis ausgegangen, die gewichtete Antworten ($g \in [0.0, 1.0]$) ableitet. Sie liefert keine eindeutige Antwort und ihre Klassen schließen sich nicht gegenseitig aus. Eine nicht eindeutige Antwort liegt vor, wenn mehrere alternative Ergebnisklassen mit einer positiven Gewichtung belegt sind. Die Klassen schließen sich nicht gegenseitig aus, wenn mehrere Ergebnisklassen auf denselben Fall zutreffen. Hat beispielsweise ein Patient Fieber und Darmkrebs und diagnostiziert ein System diese Krankheiten, dann können keine Annahmen über den gegenseitigen Ausschluß der entsprechenden Ergebnisklassen getroffen werden.

Das hier eingeführte Qualitätsmodell basiert auf Vorschlägen von Indurkhya und Weiss [IND89] für die Performanzmessung von medizinischen Expertensystemen. Bereits mit SEEK2 [POL84], [WEI88] wird Performanzmessung für ungewichtete Anworten zur Optimierung einer Wissensbasis eingesetzt. Hier wird die Messung zur Visualisierung von Performanz unter festen Bedingungen verwendet. Zunächst wird die Qualitätsmessung eines Systems beschrieben. Unter den Bedingungen, von denen das Verfahren zur Qualitätsmessung abhängt, werden externe wie interne Entscheidungsschwellen oder Parameter für die Entscheidungsfindung zusammengefaßt. Der Einfluß von Änderungen der Bedingungen auf die Qualitätsmeßwerte des Systems wird besonders untersucht. Die Ergebnisse werden als ROC (receiver operating characteristic)-Kurven dargestellt (Abbildung 4.13), die eine Beurteilung der Abhängigkeit von den Bedingungen und einen Ansatz zur Optimierung der Qualität des Systems bieten.

Es wird ein Qualitätsmodell auf der Basis von Sensitivitäts- und Spezifitätsberechnung erarbeitet. Für das Qualitätsmodell wird die untersuchte Wissensbasis auf die 2-Klassen-Situation vor und nach der Analyse projiziert. Die zwei Klassen jeweils vor und nach der Analyse müssen vollständig sein bezüglich des Eingabe- bzw. Ausgabebereichs, z.B. Klasse 1 umfaßt die Menge A, Klasse 2 umfaßt die Menge ¬ A im Eingabe- bzw. Ausgabebereich. Die Aufteilung der Mengen leitet sich aus den zu untersuchenden Ergebnisklassen ab. Diagnostiziert die Wissensbasis unter anderem Krankheit X, und soll die Qualität der Diagnosefindung der Krankheit X durch die Wissensbasis beschrieben werden, dann werden vor der Analyse die Fälle, die Krankheit X repräsentieren, der Menge A, alle übrigen Fälle der Menge ¬ A im Eingabebereich zugeordnet. Die Zuordnung richtet sich somit nach dem Diagnosespektrum der Wissensbasis. Der Vorgang wird als Vorklassifikation bezeichnet, die Methode als Vorklassifizierer.

Die Definition der Vorklassifikation basiert auf dem Merkmalsvektor, der die Wissensbasis vollständig abdeckt. Es können die Merkmale selbst oder daraus abgeleitete Merkmale herangezogen werden, um die Schwellwerte zur Klassifikation in A und ¬ A zu bestimmen. Um die Vorklassifikation durchzuführen, muß zu den Eingabeobjekten der Wissensbasis vor der Analyse das Wissen über die Zuordnung der Testfälle zur Menge A oder ¬ A vorliegen. Das Wissen kann durch Experten, Falldokumentation, Untersuchungsergebnisse oder sonstige Literatur eingebracht werden.

Die Ausprägungen von *A*-Objekten eines Testfalles, die durch ähnliche Gewichtung alternative Aussagen repräsentieren können, werden durch das Verfahren der Nachklassifikation auf eine eindeutige Aussage bei sich gegenseitig ausschließenden Ausgabeobjekten abgebildet. Die Situation, die sich dem Verfahren der Nachklassifikation darstellt, ist eine Menge alternativer *A*-Objekte, deren Ausprägungen positive Gewichtungen besitzen. Aus diesen ist die Ausprägung eines *A*-Objekts für die eindeutige Antwort auszuwählen. Ein Ansatz zur Nachklassifikation ist, die Ausprägung mit der höchsten Bewertung und bei gleicher Bewertung die Ausprägung, die die schwerwiegendere Entscheidung repräsentiert, zu wählen. Eine Variation der Nachklassifikation wäre durch ein verändertes Auswahlkriterium gegeben, beispielsweise in bestimmten Situationen die zweithöchst bewertete Ausprägung zu wählen.

Zusätzlich beinhaltet die Nachklassifikation die Zusammenfassung der Ausgabeobjekte in 2 Klassen, die sich nach der Vorklassifikation richten. Die Klasse, die der Menge A entspricht, kann mehr als ein Ausgabeobjekt enthalten, da für eine zu diagnostizierende Krankheit mehrere sich nicht gegenseitig ausschlieÃende Ausgabeobjekte vorliegen können. Die Qualitätsbeschreibung bei mehreren Ausgabeobjekten, die der Menge A zugeordnet sind, wird durch verschiedene Modelle unterstützt, die in [IND89] beschrieben sind. Die Modelle beziehen sich auf Systeme mit ungewichteten und eindeutigen Antworten :

das Fall-Korrektheits-Modell: einfache Zählung der inkorrekten Fälle und prozentuale Angabe in Bezug auf die gesamte Anzahl untersuchter Fälle

das partielle Korrektheits-Modell: differenzierte Betrachtung der Ergeb-

nisklassen, die die teilweise Korrektheit bei nur einigen übereinstimmenden Klassen wiedergibt

das diagnostische Performanz-Modell: Spezifizierung der Systemleistung relativ zu einer Klasse

Mit der Vor- und Nachklassifikation sowie der Behandlung der sich nicht gegenseitig ausschließenden Ausgabeobjekte sind die externen Bedingungen für das Qualitätsmodell gegeben. Variationen der Bedingungen resultieren aus Änderungen von Entscheidungsschwellen in den klassifizierenden Methoden oder Modifikationen der Klassifizierungsmethoden.

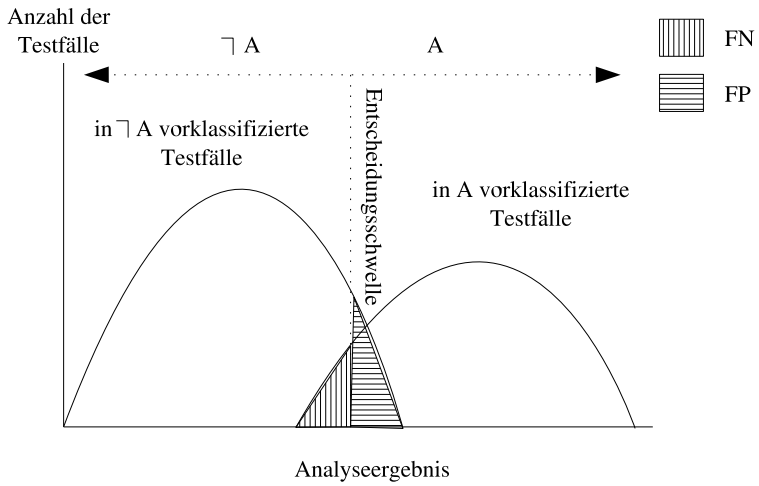

Abbildung 4.10: Verteilung der Analyseergebnisse einer Wissensbasis für Testfälle, die in die Mengen A und ¬ A vorklassifiziert sind. Die Variation der Entscheidungsschwelle durch Variation interner (Wissensbasis) und externer (Nachklassifikation) Bedingungen, verändert die Werte für die falsch negativen (FN) und die falsch positiven (FP).

Die Visualisierung der Auswirkung von Variationen externer und interner Bedingungen liefert die Basis dafür, die Abhängigkeit der Wissensbasis von statischen Parametern der Expertise zu lösen und die Wissensbasis an die lokalen Bedingungen einer Systemapplikation anzupassen. Beispielsweise läßt sich bei einer veränderten Entscheidungsschwelle für die Vorklassifikation die optimale Parametereinstellung oder eine optimale Entscheidungsstrategie ermitteln. Britton et al. führen in [BRI92b] eine Untersuchung nuklearmedizinischer Auswertungssoftware in britischen Krankenhäusern an, die beispielhaft für die Variabilität der für die Diagnose zugrundeliegenden Schwellwerte ist. Die Auswertung ergab für die unteren Normalwerte der linksventrikulären Auswurffraktion eine Schwankungsbreite von 35% bis 75%. Solange die Standardisierung der Datenakquisition, der Datenvorverarbeitung und der Schwellwerte für die Parameterklassifikation für den Problembereich nicht existieren, wird zumindest der verbreitete Einsatz des wissensbasierten Systems nicht behindert.

Da das Verfahren nach Beendigung der Prüfung der Übereinstimmung von Spezifikation und Implementation angesetzt wird, muß die Gefahr, redundante und unvollständige Wissensdefinitionen durch Parameteränderungen erneut zu generieren, besonders bedacht werden.

Eine ideale Wissensbasis bildet die in die Menge A vorklassifizierten Fälle auf die der Menge A zugeordneten Aussagen unter den *A*-Objekten und die in die Menge ¬ A vorklassifizierten Fälle auf die der Menge ¬ A zugeordenten Aussagen unter den *A*-Objekten ab. Wenn eine Überlappung in der Verteilung der Analyseergebnisse bei Testfällen aus A und ¬ A existiert, werden einige Fälle aus A als ¬ A zugehörig und einige Fälle aus ¬ A als A zugehörig interpretiert (Abbildung 4.10). Die vier Gruppen, die sich nach Vor- und Nachklassifikation gebildet haben, werden mit folgenden Termen bezeichnet:

TP (true positive - richtig positiv) ist ein Analyseergebnis mit der Aussage A zugehörig, bei einem Testfall, der als A zugehörig vorklassifiziert ist

TN (true negative - richtig negativ) ist ein Analyseergebnis mit der Aussage ¬ A zugehörig, bei einem Testfall, der als ¬ A zugehörig vorklassifiziert ist

FP (false positive - falsch positiv) ist ein Analyseergebnis mit der Aussage A zugehörig, bei einem Testfall, der als ¬ A zugehörig vorklassifiziert ist

FN (false negative - falsch negativ) ist ein Analyseergebnis mit der Aussage ¬ A zugehörig, bei einem Testfall, der als A zugehörig vorklassifiziert ist

Ist ein Wert für die Entscheidungsschwelle gewählt, kann die Leistung der Wissensbasis, nämlich Fälle, die zur Menge A gehören, von Fällen, die zur Menge ¬ A gehören, zu unterscheiden, in der 4-Felder-Tafel dargestellt werden. Die 4-Felder-Tafel enthält die Anzahl der Testfälle für jede Gruppe TP, TN, FP und FN (Abbildung 4.11). Die Summe der Testfälle in der ersten Spalte (TP+FN) ist die Anzahl der als zur Menge A gehörig vorklassifizierten Testfälle, die Summe der zweiten Spalte (FP+TN) ist die Anzahl der als zur Menge ¬ A vorklassifizierten Testfälle. Die Summe der ersten Zeile (TP+FP) ist die Anzahl der Testfälle, die nach der Analyse der Menge A zugeordnet sind, die Summe der zweiten Zeile (TN+FN) gibt die Anzahl der Testfälle an, die nach der Analyse der Menge ¬ A zugeordnet sind .

Die Messung der Leistung der Wissensbasis erfolgt mit den Termen der 4-Felder-Tafel durch die Berechnung der Sensitivität und Spezifität. Die Sensitivität ist durch das Verhältnis der Anzahl der Testfälle, die in A vorklassifiziert sind und nach der Analyse der Menge A zugeordnet sind, zu der Gesamtanzahl der in A vorklassifizierten Testfälle gegeben.

$$\text{Sensitivität} = \frac{\#\text{ Testfälle aus A, die nach der Analyse in A klassifiziert sind}}{\#\text{ Testfälle aus A}} = \frac{\text{TP}}{\text{TP+FN}}$$

		Vorklassifikation		Zeilen-summen
		A	**¬ A**	
Analyseergebnis	**A**	**TP**	**FP**	TP+FP
	¬ A	**FN**	**TN**	FN+TN
	Spalten-summen	TP+FN	FP+TN	

Abbildung 4.11: 4-Felder-Tafel für die Einordnung der Analyseergebnisse in TP (true positive), FP (false positive), FN (false negative) und TN (true negative)

Die Spezifität ist durch das Verhältnis der Anzahl der Testfälle, die in ¬ A vorklassifiziert sind und nach der Analyse der Menge ¬ A zugeordnet sind, zu der Gesamtanzahl der in ¬ A vorklassifizierten Testfälle gegeben.

$$\text{Spezifität} = \frac{\#\text{ Testfälle aus } \neg \text{ A, die nach der Analyse in } \neg \text{ A klassifiziert sind}}{\#\text{ Testfälle aus } \neg \text{ A}} = \frac{\text{TN}}{\text{TN+FP}}$$

Die berechneten Werte der Sensitivität und Spezifität hängen von der Wissensbasis und der Nachklassifikation ab, da die Entscheidungsschwelle für die Zuordnung während und nach der Analyse (Abbildung 4.10) durch die abgeschlossene Wissensbasis und die Nachklassifikation bestimmt wird.

Für die Behandlung mehrerer Ausgabeobjekte für die Qualitätsmessung (kein gegenseitiger Ausschluß) wird das diagnostische Performanzmodell zugrundegelegt. Da das Modell sich auf ungewichtete und eindeutige Antworten bezieht, werden in der Nachklassifikation die Gewichtungen und die Alternativen bei den Ausgabeobjekten so behandelt, daß eine eindeutige und ungewichtete Antwort gegeben ist.

Für die Qualitätsmessung nach dem diagnostischen Performanzmodell werden jedem Testfall die p Ausgabeobjekte zugeordnet, die als korrekte Antwort erwartet werden. Diese Ausgabeobjekte bilden die Menge A. Jeder Testfall wird in p-Fälle unterteilt, entsprechend der Anzahl der korrekten Ausgabeobjekte für den Testfall. Somit ergibt sich für jeden p-Fall eine korrekte Schlußfolgerung. Die Qualität wird für jeden p-Fall gemessen. Die Wissensbasis analysiert einen p-Fall, der in die Menge A vorklassifiziert ist, korrekt, wenn dieser nach

der Analyse der Menge A zugeordnet ist. Mit diesem Ansatz werden die Ausgabeobjekte der Wissensbasis als Objekte, die sich gegenseitig ausschließen, betrachtet.

Sei beispielsweise die Menge $A \cup \neg A$ = Objekt1, Objekt2, Objekt3 mit den in Abbildung 4.12 a aufgeführten Vorklassifikationen und Analyseergebnissen nach Nachklassifikation. Die p-Fälle des Testfalles 1 mit p=2 bilden die Menge A = Objekt1, Objekt2. Die p-Fälle des Testfalles 2 bilden die Menge A = Objekt1, Objekt3. Für jeden p-Fall kann nur ein Objekt im Analyseergebnis mit der korrekten Antwort übereinstimmen. Das Ergebnis der Überprüfung kann auf die Terme der 4-Felder-Tafel abgebildet werden, wie die Abbildung 4.12 b verdeutlicht. Sie zeigt die nach den untersuchten Klassen aufgelisteten Ergebnisse der Qualitätsmessung.

a

Testfall	Vorklassifikation	Analyseergebnis
1	Objekt1, Objekt2	Objekt1, Objekt3
2	Objekt1, Objekt3	Objekt2, Objekt3

b

Objekt	TP / (TP+FN)	FP
Objekt1	1 / 2	0
Objekt2	0 / 1	1
Objekt3	1 / 1	1
Gesamt	2 / 4	2

Abbildung 4.12: Qualitätsmessung nach dem diagnostischen Performanzmodell am Beispiel einer hypothetischen Wissensbasis mit den Ausgabeobjekten Objekt1, Objekt2 und Objekt3

Die Wissensbasis steht in ihrer Gesamtheit deklarativer und prozeduraler Bestandteile für die Qualität der Entscheidung. Interessant sind die Elemente, über die die Qualität ohne strukturelle Änderungen zu beeinflussen ist. Wird die Entscheidungsfindung durch verschiedene Parameter geprägt, hängt die Qualität von der Wahl der Parameterwerte ab. Die Parameter können einfache Schwellen oder funktionsbeschreibende Parameter sein. Bei einer Wissensbasis, deren Entscheidungsfindung auf Gewichtsfunktionen für die Begriffe und Aussagen basiert, wird die Qualität der Entscheidung durch die Wahl der Parameter der Gewichtsfunktionen beeinflußt. Eine Veränderung der Parameter, die eine Erhöhung der Schwelle zur Folge hat, würde die Anzahl falsch-positiver Ergebnisse signifikant verringern, gleichzeitig auch die Anzahl der falsch-negativen Ergebnisse erhöhen. Die Wissensbasis würde mehr spezifisch und weniger sensitiv sein. Wo die Schwelle letztendlich gesetzt wird, also welche Parameterwerte festgeschrieben werden, ist eine Entscheidung, die von verschiedenen Erwägungen getragen wird. Ist beispielsweise die Krankheit, die durch die Wissensbasis diagnostiziert wird, sehr schwerwiegend, aber therapierbar, dann ist es erstrebenswert, die Anzahl der falsch-negativen Analyseergebnisse zu minimieren. Im anderen Extremfall, wenn die Krankheit nicht sehr schwerwiegend ist, jedoch die Therapie Gefahren in sich birgt, dann ist die Minimierung der Anzahl der falsch-positiven Analyseergebnisse das Ziel.

Die Auswirkungen der Variierung der internen Bedingungen bei demselben Testdatensatz spiegelt sich in der Veränderung von Spezifität und Sensitivität wider. Wie sich Spezifität und Sensitivität relativ zueinander verändern, kann durch die ROC (receiver operating characteristic)-Kurve visualisiert werden. In der ROC-Kurve ist die Sensitivität der Wissensbasis gegenüber 1-Spezifität bei variierten internen Bedingungen graphisch dargestellt (Abbildung 4.13).

Aus dem Verlauf der ROC-Kurve können nicht nur die Veränderungen von Sensitivität relativ zu 1-Spezifität beobachtet werden. Ein direkter Vergleich mit ROC-Kurven alternativer Wissensbasen ist denkbar, sofern das gleiche Qualitätsmodell zugrundeliegt. Die Bedingungen für die Vergleichbarkeit werden hier nicht weiter untersucht, sondern es soll lediglich die Möglichkeit aufgezeigt werden. Wie in Abbildung 4.13 gezeigt ist, wäre bei einer Vergleichbarkeit die Aussage möglich, daß eine Wissensbasis besser als eine konkurrierende Wissensbasis ist, wenn ihre ROC-Kurve oberhalb der Kurve der konkurrierenden Wissensbasis liegt. In der Abbildung ist die Wissensbasis B besser als die Wissensbasis A, da die Sensitivität von B für alle Werte der Spezifität größer als die Sensitivität von A ist.

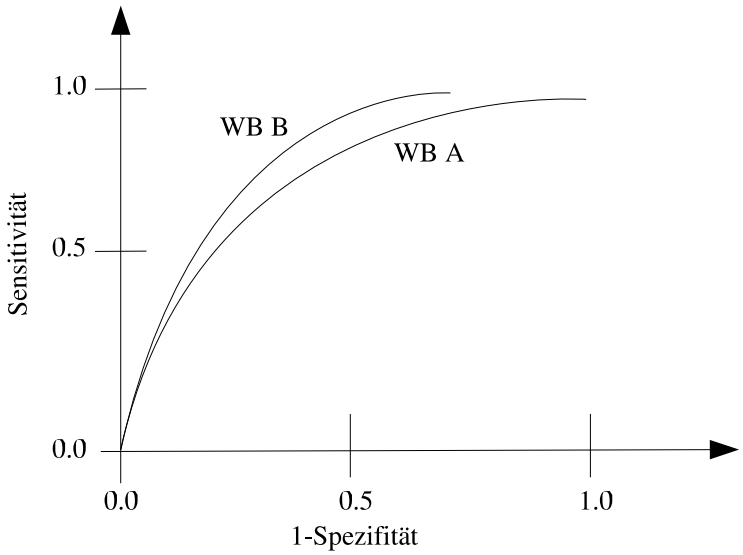

Abbildung 4.13: *Receiver operating characteristic* (ROC)-Kurve für zwei hypothetische Wissensbasen. Wissensbasis B ist besser als Wissensbasis A, da die Sensitivität für alle Spezifitätswerte höher ist

4.8 Zusammenfassung

Es wurde eine Teststrategie entwickelt, die durch die Betrachtung der Testfälle im Kontext anderer Testfälle auch das nicht formale vage und heuristische Expertenwissen quantitativ erfaßt. Durch die Abbildung auf einen Merkmalsraum können Strukturen definiert werden, die durch vages und heuristisches Wissen nicht verändert werden und über die funktionale Beschreibung auch für die Ergebnisse gelten.

Die Beschreibung der Eigenschaften der Ein- und Ausgabedaten, wie sie für die medizinischen Bilddaten typisch sind, ist relevant für die Definition des Ergebnismusters, das sich aus dem Kontext und der Struktur der Eingabedaten durch die funktionale Beziehung ergibt.

Die Erfüllung der Bedingungen, die sich durch die Anforderungen der Teststrategie ergeben, sind auch ein Argument für die Notwendigkeit von Generierungsmethoden für künstliche Testfälle. Originale Daten allein erfüllen nur zufällig die strukturellen Voraussetzungen für die Überprüfung eines Ergebnismusters. Erst mit einer geeigneten Methode zur Erzeugung künstlicher Daten wird das Validierungsverfahren praktikabel. Es erlaubt, beliebige Teilbereiche der spezifizierten Ergebnisse beliebig detailliert zu untersuchen.

Mit der Spezifizierbarkeit eines Ergebnismusters läßt sich ein Vergleich von erwartetem und erhaltenem Muster durchführen. Bei der Feststellung von Abweichungen liegen die Ursachen in Fehlern in der Wissensbasis. Im Vergleich zu herkömmlichen Verfahren impliziert die kontextbezogene Darstellung auch das Expertenwissen. Die Abweichungen repräsentieren somit auch mangelhaft modellierte Expertise. Die Fehlerlokalisation erfordert weitere nur schwer zu verallgemeinernde Maßnahmen. Sie hängen von den speziellen Bedingungen der Wissensbasis ab und erfordern in der Regel die Erfahrung der Entwickler oder ein spezielles Werkzeug für das *Debugging* der Wissensbasis. Mit dem Ausblick auf den Einsatz des Verfahrens für ein wissensbasiertes System für die Analyse medizinischer Bilddaten mit gewichteten alternativen Antworten ist die Erläuterung der Fehlerlokalisation auf die Ergebnismuster von Gewichtungsverläufen ausgerichtet, wie sie für diese Fälle gelten. Eine weitere Verallgemeinerung ist nicht vorgesehen.

Die quantifizierte Beschreibung des Analyseverhaltens der Wissensbasis ist die Grundlage für QualitÃ?tsmessungen, die für die entwicklungsinterne oder benutzerexterne Darstellung verwendbar sind. Die Zuverlässigkeitsmessung für den medizinischen Anwendungsbereich ist unabhängig von strukturierten Eingabedaten. Sie kann ebenso mit originalen Testdaten durchgeführt werden.

Literaturverzeichnis

[ADR82] Adrion, W.R., Branstad, M.A., Cherniavsky, J.C. Validation, Verification and Testing of Computer Software. *Computing Surveys* 1982; 14(2):159-192.

[AYE91] Ayel, M., Laurent, J.-P. (eds.): *Validation, Verification and test of knowledge-based systems.* Chichester: John Wiley & Sons; 1991.

[BRI92a] Britton, K.E., Bardoux, C.: Quality Assurance on Nuclear Medicine Software. *COST Project B2, 3rd Annual Report 1991.* Commissionof the European Communities; 1992.

[BRI92b] Britton, K.E., Busemann Sokole, E.: COST B2: why and wherefore. *Eur J Nucl Med* 1992; 19:563-568.

[BUS90a] Busemann Sokole, E.: *Quality Assurance in Nuclear Medicine Imaging.* Amsterdam: Rodopi; 1990.

[BUS90b] Busemann Sokole, E., Cradduck, T.D., Erickson, J.J.: Experience with gated cardiac software phantoms for quality control of application programs. *Eur J Nucl Med* 1990; 17:106-110.

[CEN87] EN29000-29004 *Übernahme der internationalen Normen ISO 9000-9004 [ISO87b-f] als europäische Norm.* Europäisches Komitee für Normung; 1987.

[CLA84b] Clancey, W.J., Shortliffe, E.H.: Introduction: Medical Artificial Intelligence Programs. In: Clancey, W.J., Shortliffe, E.H. (eds.): *Readings in Medical Artificial Intelligence.* Reading Massachusetts: Addison-Wesley; 1984:1-17.

[DAV79] Davis, R.: Interactive transfer of expertise: acquisition of new inference rules. *Aritificial Intelligence* 1979; 12:121-157.

[DEU82] Duetsch, M.S. *Software Verification and Validation.* Prentice Hall; 1982

[DIN87] DIN 55350 Teil 13. *Begriffe der Qualitätssicherung und Statistik; Begriffe zur Genauigkeit von Ermittlungsverfahren und Ermittlungsergebnissen.* Deutsches Insitut für Normung e.V.; 1987

[DIN90] DIN 66285. *Anwendungssoftware. Gütebedingungen und Prüfbestimmungen.* Deutsches Insitut für Normung e.V.; 1990.

[DIN92] DIN ISO 8402, 9000-9004, 9000-3. *Übernahme der internationalen Normen ISO 8402, 9000-9004, 9000-3 [ISO87a-f] und [ISO91] als deutsche Norm.* Deutsches Institut für Normung e.V.; 1992.

[DUD88] *Duden Informatik.* Mannheim: Bibliographisches Institut; 1988.

[DUN87] Duncan, J.S. Knowledge Directed Left Ventricular Boundary Detection in Equilibrium Radionuclide Angiocardiography. *IEEE Transactions on Medical Imaging.* 1987; 6(4):325-336.

[DUR84] Duran, J.W., Ntafos, S.C.: An Evaluation of Random Testing. *IEEE Transactions on Software Engineering.* 1984; 10(4):438-444.

[END77] Endres, A., *Analyse und Verifikation von Programmen.* München-Wien; 1977.

[EVA87] Evans, M.W., Marciniak, J.J. *Software Quality Assurance and Management.* Chichester: John Wiley & Sons; 1987

[FOX90] Fox, M.S.: AI and Expert Systems: Myths, Legends, and Facts. *IEEE Expert* 1990;February:8-20.

[GAS83] Gaschnig, J., Klahr, P., Pople, H., Shortlieffe, E., Terry, A.: Evaluation of Expert Systems: Issues and Case Studies. In: Hayes-Roth, F., Waterman, D.A., Lenat, D.B. (eds.) *Building expert systems.* London: Addison-Wesley; 1983:241-280.

[GOO75] Goodenough, J.B., Gerhart, S.L.: Toward a Theory of Test Data Selection. *IEEE Transations on Software Engineering.* 1975; 1(2):156-173.

[GOU83] Gourley, J.S.: A Mathematical Framework for the Investigation of Testing. *IEEE Transactions on Software Engineering.* 1983; 9(6):686-709.

[HOE92] Hoenen, M., Kloth, M., Steven, E. Verifikation oder Überprüfbarkeit? *KI* 1992;4:67-70

[HUA81] Huang, J., Experience with Use of Instrumentation Techniques in Software Testing. In: *Proceedings NSIA National Conference of Software Technology and Management.* 1981:D1-D10.

[HWA86] Hwang, V., Shang-Shouq, Davis, L.S., Matsuyama, T. Hypothesis Integration in Image Understanding Systems. *Computer Vision, Graphics, and Image Processing.* 1986; 36:321-371

[IEEE83] IEEE Std 729. *Glossary of Software Engineering Terminology.* 1983.

[IND89] Indurkhya, N., Weiss, S.M.: Models for Measuring Medical Expert System Performance. *Artificial Intelligence in Medicine* 1989; 1:61-70.

[ISO87a] ISO 8402. *Quality-Vocabulary.* International Organization for Standardiasation; 1987.

[ISO87b] ISO 9000. *Quality management and quality assurance standards - guidelines for selection and use.* International Organization for Standardiasation; 1987.

[ISO87c] ISO 9001 *Quality systems - Model for quality assurance in design/development, production, installation and servicing.* International Organization for Standardiasation; 1987.

[ISO87d] ISO 9002 *Quality systems - Model for quality assurance in production and installation.* International Organization for Standardiasation; 1987.

[ISO87e] ISO 9003 *Quality systems - Model for quality assurance in final inspection and test.* International Organization for Standardiasation; 1987.

[ISO87f] ISO 9004 *Quality management and quality system elements - Guidelines.* International Organization for Standardiasation; 1987.

[ISO91] ISO 9000-3. *Quality management and quality assurance standards; Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software.* International Organization fo

[ITZ84] Itzfeld, W., Schmidt, M., Timm, M.: Spezifikation von Verfahren zur Validierung von Software-Qualitätsmaßen. *Angewandte Informatik* 1984; 1:12-21.

[KUL82] Kulikowski, C., Weiss, S.M. Representation of expert knowledge for consultation: the CASNET and EXPERT projects. In: Szolovits, P. (ed.): *Artificial Intelligence in Medicine.* Boulder, CO: Westview Press; 1982.

[LAU90] Laufmann, St.C., DeVaney, D.M., Whiting, M.A.: A Methodology for Evaluating Potential KBS Applications. *IEEE Expert* 1990; December:43-61.

[LIE89] Liedtke, C.E., Ender, M. *Wissensbasierte Bildverarbeitung.* Berlin: Springer; 1989.

[MAR88] Mars, Miller, P.L.: Knowledge Acquisition and Verification Tools for Medical Expert Systems. In Miller, P.L. (ed.): *Selected Topics in Medical Artificial Intelligence.* New York: Springer; 1988:141-152.

[MCC76] McCabe, T.: A Complexity Measure. *IEEE Transactions on Software Engineering* 1976; 2(4):308-320.

[MES92] Meseguer, P. Towards a Conceptual Framework for Expert System Validation. *AICOM* 1992; 5(3), September: 119-135.

[MIL82] Miller, R.A., Pople, H.E., Myers, J.D. Internist-1, an experimental computer-based diagnostic consultant for general internal medicine. *N Engl J Med* 1982; 307:468-476.

[MIL84] Miller, R.A., Pople, H.E., Myers, J.R., Myers, J.D. INTERNIST-1, An Experimental Computer-Based Diagnostic Consultant for general Internal Medicine. In: Clancey, W.J., Shortliffe, E.H. (eds.) *Readings in Medical Artificial Intelligence.* Reading, Massachusetts: Addison-Wesley; 1984:190-209

[MIL88] Miller, P.L.,: Artificial Intelligence in Medicine: An Emerging Discipline. In: Miller, P.L. (ed.): *Selected Topics in Medical Artificial Intelligence.* New York: Springer; 1988:1-9.

[MYE79] Myers, G.J., *The Art of Software Testing.* Chichester: John Wiley & Sons; 1979.

[MYE89] Myers, G.J., *Methodisches Testen von Programmen.* München: Oldenbourg; 1989.

[NAZ84] Nazif, A.M., Levine, M.D. Low Level Image Segmentation: An Expert System. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 1984; 6(5):555-577.

[NIE87] Niemann, H., Bunke, H. *Künstliche Intelligenz in Bild- und Sprachanalyse.* Stuttgart: B.G. Teubner; 1987.

[NGU85] Nguyen, T.A., Perkins, W.A., Laffey, T.J., Pecora, D.: Checking an expert systems knowledge base for consistency and completeness. In: Aravind, J. (ed.): *IJCAI85 Proceedings of the Ninth International Joint Conference on Artificial Intelligence.* Los Altos California: Morgan Kaufmann Publishers; 1985:375-378.

[NGU87] Nguyen, T.A., Perkins, W.A., Laffey, T.J., Pecora, D.: Knowledge base verification. *AI Magazine* 1987;2:69-75.

[OKE87] O'Keefe, R.M., Balci, O., Smith, E.P.: Validating Expert System Performance. *IEEE Expert* 1987; Winter:81-89.

[OLE87] O'Leary, D.E.: Validation of Expert Systems. *Decisions Sciences* 1987; Summer, 18.

[OLE90] O'Leary, D.E.: Soliciting Weights or Probabilities From Experts for Rule-Based Systems. *International Journal of Man-Machine Studies* 1990; 32:293-301.

[OLE91] O'Leary, D.E.: Design, Development and Validation of Expert Systems: A Survey of Developers. In: Ayel, M., Laurent, J.P. (eds.): *Validation, Verification and test of knowledge-based systems.* Chichester: John Wiley & Sons; 1991:3-19.

[PAV77] Pavlidis, T.: *Structural Pattern Recognition.* Berlin: Springer-Verlag; 1977.

[POL84] Politakis, P., Weiss, S.: Using empirical analysis to refine expert system knowledge bases. *Artificial Intelligence* 1984; 22:23-48.

[PRA78] Pratt, W.K. *Digital image processing.* New York; John Wiley & Sons; 1978.

[RAM82] Ramamoorthy, C.V., Dong, S.L., Ganesh, S.L., Jen, C.H., Tsai, W.T.: Techniques in Software Quality Assurance. In: Sneed, H.M., Wiehle, H.R. (eds.) *Software-Qualitätssicherung, Tagung 1982 des German Chapter of the ACM.* Stuttgart; 1982:11-34.

[SAG85] Sagerer, G. *Darstellung und Nutzung von Expertenwissen für ein Bildanalysesystem.* Informatik Fachberichte 104, Berlin: Springer; 1985.

[SCH83] Schmitz, P., Bons, H., van Megen, R. *Software-Qualitätssicherung, Testen im Software-Lebenszyklus* Braunschweig/Wiesbaden: Vieweg; 1983.

[SCH86] Schefe, P. *Künstliche Intelligenz.* Reihe Informatik Band 52. Mannheim: Bibliographisches Institut; 1986.

[SCH90] Schill, K.: *Medizinische Expertensysteme.* München: Oldenbourg; 1990.

[SHO76] Shortliffe, E.H. *Computer-Based Medical Consultations: MYCIN.* New York; Elsevier; 1976.

[SHO81] Shortliffe, E.H., Scott, A.C., Bischoff, M.B., Campbell, A.B., van Melle, W., Jacobs, C.D.: ONCOCIN: an expert system for oncology protocol management. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence.* Los Altos CA: Morgan Kaufmann Publishers; 1981:876-881.

[SHW89] Shwe, M.A., Tu, S.W., Fagan, L.W.: Validating the Knowledge Base of a Therapy Planning System. *Meth Inform Med* 1989; 28:36-50.

[STA86] Stansfield, S.A. ANGY: A Rule-Based Expert System for Automatic Segmentation of Coronary Vessels From Digital Subtracted Angiograms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1986; 8(2):188-199.

[SUW82] Suwa, M., Scott, A.C., Shortliffe, E.H.: An approach to verifying completeness and consistency in rule-based expert system. *AI Magazine* 1982; 2:16-21.

[WAL90] Wallmüller, E. *Software-Qualitätssicherung in der Praxis.* München Wien: Carl Hanser; 1990.

[WEI88] Weiss, S.M., Politakis, P., Ginsberg, A.: Empirical Analysis and Refinement of Expert System Knowledge Bases. In Miller, P.L. (ed.): *Selected Topics in Medical Artificial Intelligence.* New York: Springer-Verlag; 1988:153-166.