



Christian Demant
Bernd Streicher-Abel
Axel Springhoff

Industrielle Bildverarbeitung

Wie optische Qualitätskontrolle
wirklich funktioniert

3. Auflage



 Springer

Industrielle Bildverarbeitung

Christian Demant • Bernd Streicher-Abel
Axel Springhoff

Industrielle Bildverarbeitung

Wie optische Qualitätskontrolle wirklich
funktioniert

3. Aktualisierte Auflage



Springer

Dipl.-Ing. Christian Demant
Dipl.-Ing. Bernd Streicher-Abel
Dipl.-Ing. (FH) Axel Springhoff
NeuroCheck GmbH
70174 Stuttgart
Deutschland

www.neurocheck.com
fachbuch@neurocheck.de

<http://extras.springer.com/2011/978-3-642-13096-0>

Autoren der 2. Auflage: C. Demant, B. Streicher-Abel, P. Waskewitz

ISBN 978-3-642-13096-0
DOI 10.1007/978-3-642-13097-7
Springer Heidelberg Dordrecht London New York

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

© Springer-Verlag Berlin Heidelberg 1998, 2002, 2011
Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zu widerhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Einbandentwurf: WMXDesign GmbH, Heidelberg

Gedruckt auf säurefreiem Papier

Springer ist Teil der Fachverlagsgruppe Springer Science+Business Media (www.springer.com)

Vorwort

Die industrielle Bildverarbeitung (abgekürzt IBV) hat sich seit Veröffentlichung der 2. Auflage unseres Buches im Frühjahr 2001 wie im vorangehenden Jahrzehnt eines anhaltend starken Wachstums erfreut. Nachdem die Bildverarbeitung in Deutschland in 2000 erstmals die Grenze von 1 Mrd. DM Umsatz überschritten hatte, wurde 2005 die entsprechende Euro-Marke erreicht. Die durchschnittliche Wachstumsrate zwischen 2000 und 2010 lag bei 6,4 % (VDMA). Man muss schon sorgfältig suchen, um Branchen mit einer vergleichbaren Wachstumsdynamik zu identifizieren.

Aber auch dieses Hochglanz-Image hat erste Kratzer bekommen: 2009 mussten die Unternehmen im Zuge der internationalen Wirtschaftskrise erstmals in der Geschichte der Branche deutliche Einbußen hinnehmen (-21 %).

Hinzu kommt die Tatsache, dass die IBV mittlerweile die Phase „konsolidierte Branche“ im Lebenszyklus des Wirtschaftszweiges erreicht hat. Die Aufbruch-Stimmung der 90er Jahre ist weitgehend verflogen, technologische Quantensprünge sind selten und mittlerweile steht die permanente Senkung der Systemkosten im Vordergrund. Start-Ups kommen nur noch mit Hilfe massiver Subventionen in den Markt und können sich selten dauerhaft jenseits der Marke „Kleinunternehmen“ etablieren. Umgekehrt nimmt die Anzahl der Kooperationen zu, und so mancher große Anbieter wächst vor allem durch gezielte Akquisition kleinerer Unternehmen.

Dort, wo in den 90er Jahren noch ein kniffliger Algorithmus spontan überzeugen konnte, zählen heute primär die Zuverlässigkeit der Software im Dauerbetrieb und die reibungslose Einbindung in vernetzte Fertigungs-Strukturen.

Da alle Beteiligten in der Industrie unter zunehmenden Zeitdruck stehen, bekommt die intelligente, einfach zu bedienende Funktionalität eine immer größere Bedeutung. Die System-Anbieter müssen, wo immer möglich, auf leistungsfähige Hard- und Software-Standards setzen, weil die Entwicklung proprietärer Systeme vom Markt technisch und finanziell nicht mehr akzeptiert wird.

Dennoch:

Das Thema ist und bleibt faszinierend, und es gibt eine Vielzahl von Gründen, warum die IBV auch in den nächsten Jahren international weiter erfolgreich wachsen sollte.

Die Sicherung der Qualität besitzt im produzierenden Gewerbe oberste Priorität. „Sehende“ Maschinen vermessen hochpräzise Teile, führen Roboter-Arme an die richtige Position und identifizieren die Werkstücke in ihrem Produktionsfluss vom Wareneingang bis zum Versand.

Man kann es zusammenfassen: Ohne IBV ist industrielle Produktion nicht mehr denkbar! Deshalb durchdringen Sichtprüfsysteme mittlerweile Unternehmen aller Größen und Branchen.

Speziell die deutsche Industrie mit ihrer starken mittelständischen Prägung stellt die IBV immer wieder vor eine Vielzahl unterschiedlichster und anspruchsvoller Aufgaben. Deutsche Bildverarbeitungsanbieter sind dadurch in vielen Bereichen weltweit führend, vor allem dort, wo es auf Vielseitigkeit, Flexibilität und Integration in unterschiedliche Produktionsumgebungen ankommt. Herausragende Kompetenz in Bezug auf die Lösung der Bildverarbeitungsaufgabe ist allerdings absolute Grundvoraussetzung, um überhaupt als Anbieter ernsthaft in Betracht gezogen zu werden. Vor diesem Hintergrund ist eine praxisorientierte Einführung in die Bildverarbeitung aktueller denn je.

Das vorliegende Buch beruht auf jahrelangen praktischen Erfahrungen der Autoren mit der Entwicklung und Inbetriebnahme von Sichtprüf anlagen in der industriellen Fertigung. Wir haben versucht, einen anderen Weg zu gehen als in Büchern über (digitale) Bildverarbeitung allgemein üblich. Anstelle einer Vorstellung einzelner Verfahren in einer auf den mathematischen Eigenschaften dieser Verfahren begründeten Systematik zeigen wir konkrete Anwendungen, die – von einigen illustrativen Beispielen abgesehen – der industriellen Praxis entstammen. Die Lösung dieser Bildverarbeitungsaufgaben motiviert dann die Darstellung der dafür eingesetzten Algorithmen, wobei es uns weniger auf theoretische Erwägungen ankam als auf die praktischen Anwendungsmöglichkeiten dieser Methoden und auf ihr Zusammenwirken innerhalb eines durchgängigen Systementwurfs. Dabei werden die mathematischen Grundlagen natürlich nicht vernachlässigt, aber sie stehen nicht in unserem Fokus.

Auf diese Weise hoffen wir, sowohl Studenten als auch Praktikern aus der industriellen Fertigung eine Vorstellung von den Möglichkeiten der Bildverarbeitung in der industriellen Qualitätskontrolle zu vermitteln, Verständnis für die Voraussetzungen ihres Einsatzes zu wecken und ein Gefühl für die Vorgehensweise in praktischen Anwendungsfällen zu geben.

Wir möchten uns bei Frau Hestermann-Beyerle und Frau Kollmar-Thoni vom Springer-Verlag für die Gelegenheit bedanken diese dritte Auflage vorzulegen.

Für die langjährige gute Zusammenarbeit und die konstruktive Unterstützung beim Schreiben des Kapitels über Digitalkameras bedanken wir uns bei der Firma Baumer Optronic in Radeberg.

Für die Bereitstellung und Aufbereitung des anschaulichen Anwendungsbeispiels im Kapitel „Farbbildverarbeitung“ bedanken wir uns bei der Firma IVS Industrial Vision Systems Ltd., Kingston Bagpuize, UK.

Weiterhin möchten wir uns bei all jenen bedanken, die uns in den letzten Jahren auf unterschiedlichste Weise unterstützt und sich für uns und NeuroCheck engagiert haben. Ohne ihre Arbeit und ihren Einsatz wären unser Unternehmen und unsere Software nicht da, wo sie heute erfolgreich stehen. Und damit wäre auch

das vorliegende Werk in dieser Form nicht möglich gewesen. Wir bedanken uns bei:

- Dipl.-Inf. Marcellus Buchheit, Edmonds / Seattle, USA
- Dipl.-Ing. (FH) Richard Herga, Süßen
- Bernd Marquardt, Dormagen
- Prof. Dr. Konrad Sandau, Darmstadt
- Dipl.-Ing. (FH) Anton Schmidt, Bernau
- Patentanwalt Dipl.-Ing. Christoph Sturm, Wiesbaden
- Dr.-Ing. Peter Waszkewitz, Kornwestheim

Stuttgart, im Herbst 2010

*Dipl.-Ing. Christian Demant
Dipl.-Ing. Bernd Streicher-Abel
Dipl.-Ing. (FH) Axel Springhoff*

NeuroCheck GmbH

www.neurocheck.com

fachbuch@neurocheck.de

Geleitwort

Als Student der ETH Zürich bin ich Mitte der 80er Jahre erstmals mit Bildverarbeitung in Berührung gekommen. Das Thema wurde damals vor allem wissenschaftlich in der Theorie beleuchtet (Algorithmen) und hatte keinen praktischen Nutzen in der Automatisierungstechnik. Teure Spezialhardware mit schwachen, nicht-standardisierten Prozessoren, verbunden mit einer fehleranfälligen Assembler-Programmierung, führten zu einer für industrielle Prozesse mangelhaften Zuverlässigkeit und verhinderten dadurch die Verbreitung.

Vertieft habe ich die Thematik während meiner Promotion Anfang der 90er Jahre, als ich damals als einer der weltweit ersten Doktoranden am Paul Scherrer Institut in Zürich (heute Centre Suisse d'Electronique et de Microtechnique, abgekürzt CSEM) mit an zu dieser Zeit neuartigen CMOS Bildsensoren geforscht habe. Seither hat mich das Thema Bildverarbeitung in meiner beruflichen Weiterentwicklung nicht mehr losgelassen.

Mitte der 90er Jahre begann im Windschatten der rasanten Entwicklung der PC-Technologie ein Siegeszug der industriellen Bildverarbeitung, der bis heute anhält. Bildverarbeitungssysteme sind aus modernen industriellen Fertigungsprozessen nicht mehr wegzudenken. Viele Automatisierungslösungen werden sogar erst durch den Einsatz der Bildverarbeitung ermöglicht. Aus einer abstrakten Wissenschaft hat sich die industrielle Bildverarbeitung zu einer immer noch anspruchsvollen, aber zugleich extrem nützlichen Schlüsseltechnologie der modernen Automatisierungstechnik entwickelt.

Den Autoren ist zu verdanken, dass mit dem vorliegenden Buch 1998 erstmals ein Fachbuch zum Thema industrielle Bildverarbeitung zur Verfügung stand, das sowohl für Einsteiger, als auch für Fortgeschrittene gleichermaßen zum idealen Einklese- respektive Nachschlagewerk wurde. Es ist kein abstraktes akademisches Werk, sondern beleuchtet auf eine verständliche Art und Weise das methodische Vorgehen und die mathematischen Grundsätze wichtiger Bildverarbeitungsfunktionen. Ebenso behandelt es alle wesentlichen Aspekte zur Realisierung von industriellen Bildverarbeitungssystemen für die Qualitätssicherung in industriellen Fertigungsprozessen. Von der Beleuchtung, über die Optik, die Kamera und Bilderfassungshardware, die grundsätzlichen Software-Algorithmen bis hin zu Automatisierungsschnittstellen, werden die Zusammenhänge beschrieben.

Einzigartig macht dieses Buch der Praxisbezug. Mit Hilfe der von den Autoren Demant und Streicher-Abel entwickelten professionellen Bildverarbeitungs-Software NeuroCheck kann der Leser die vielen im Buch beschriebenen Praxisbeispiele per Drag&Drop am PC nachvollziehen und interaktiv neu parametrieren.

Aus Sicht meiner damaligen akademischen Tätigkeit am Institut eine absolut revolutionäre Vorgehensweise!

Daher hat sich das Buch in kurzer Zeit nach Veröffentlichung zu Recht für viele an der Bildverarbeitung Interessierte zu einem Standardwerk entwickelt. Und es hat auch heute noch seine Berechtigung, auch wenn der heutige Bildverarbeiter üblicherweise keine Algorithmen mehr entwickeln muss, da am Markt Standardsoftware verfügbar ist, mit der sogar komplexe Anwendungen realisierbar sind. Das im Buch beschriebene Verständnis für das Zusammenspiel aller Komponenten ist bis heute wichtig und hat nach wie vor Gültigkeit.

Seit der Veröffentlichung der 2. Auflage im Jahr 2001 hat sich insbesondere im Bereich der Bildverarbeitungshardware, sprich der bildgebenden Sensorik, sehr viel getan. Die Bereitstellung moderner Digital-Kameras mit immer schnelleren CCD- und CMOS-Sensoren sowie die Verfügbarkeit moderner digitaler Schnittstellen wie USB, IEEE 1394 („FireWire“) und Gigabit Ethernet, haben die Bildverarbeitung nochmals schneller und leistungsfähiger werden lassen. Moderne Multi-Core-CPU Technologie erlaubt es, Bildverarbeitungs-Software noch komfortabler und zuverlässiger zu machen – und dies bei gleichzeitig sinkenden Kosten.

Die Autoren tragen dieser Entwicklung in der nun vorliegenden und stark überarbeiteten 3. Auflage Rechnung und zeigen nicht zuletzt mit der seit September 2009 verfügbaren neuen NeuroCheck Softwareversion 6.0, wie „state of the art“ Bildverarbeitungssysteme aussehen können. Dieses Standardwerk darf auch in seiner neusten Auflage in keinem Bücherregal fehlen.

Frauenfeld (Schweiz), im Juli 2010

Dr. Oliver Vietze

CEO & Chairman
Baumer Group
www.baumer.com

Inhaltsverzeichnis

1. Einleitung	1
1.1 Warum noch ein Buch über Bildverarbeitung?.....	1
1.2 Möglichkeiten und Grenzen.....	3
1.3 Typen von Sichtprüfaufgaben.....	5
1.3.1 Einteilung nach dem Prüfziel.....	5
1.4 Aufbau von Bildverarbeitungssystemen	6
1.4.1 Hardwareaufbau.....	7
1.4.2 Signalfluss in der Prozessumgebung.....	10
1.4.3 Signalfluss innerhalb des Bildverarbeitungssystems	13
1.5 Vorgehensmodell.....	15
1.6 Einführungsbeispiel	17
1.6.1 Schrifterkennung.....	18
1.6.2 Gewindetiefe.....	21
1.6.3 Anwesenheitskontrolle.....	23
1.7 Weiteres Vorgehen	25
2. Querschnitt: Bildvorverarbeitung	29
2.1 Grauwerttransformation	30
2.1.1 Look-Up-Tabellen	30
2.1.2 Lineare Grauwertskalierung.....	33
2.1.3 Kontrastnormierung	33
2.1.4 Histogramm-Ebnung.....	34
2.1.5 Lokale Kontrastnormierung	36
2.2 Bildarithmetik	38
2.2.1 Bildaddition und Bildmittelung	39
2.2.2 Bildsubtraktion.....	39
2.2.3 Minimum und Maximum zweier Bilder	42
2.2.4 Shading-Korrektur	42
2.3 Lineare Filter	44
2.3.1 Lokale Operationen und Bildumgebungen	45
2.3.2 Prinzip linearer Filter	45
2.3.3 Glättungsfilter	48
2.3.4 Kantenfilter	54
2.4 Medianfilter	60
2.5 Morphologische Filter.....	62
2.6 Andere nichtlineare Filter	64

2.7 Koordinatentransformationen	66
2.8 Integraltransformationen.....	69
2.9 Zusammenfassung	69
3. Lageerkennung	73
3.1 Position eines Einzelobjekts	73
3.1.1 Positionierung anhand des Gesamtobjekts.....	74
3.1.2 Positionierung anhand einer Kante	76
3.2 Ausrichtung eines einzelnen Objekts.....	80
3.2.1 Orientierungsberechnung anhand der Hauptachse.....	80
3.2.2 Polarabstandprojektion.....	83
3.3 Roboterpositionierung	85
3.3.1 Aufgabenstellung	86
3.3.2 Bildverarbeitungskomponenten	87
3.3.3 Positionsermittlung an einem Objekt.....	88
3.3.4 Orientierung einer Objektgruppe	89
3.3.5 Bemerkungen zur Positions korrektur	90
3.4 Zusammenfassung	92
4. Querschnitt: Segmentierung.....	95
4.1 Arbeitsbereiche	95
4.1.1 Arbeitsbereiche und Objekte	95
4.2 Binärsegmentierung.....	96
4.2.1 Schwellwerte.....	97
4.2.2 Schwellwertbestimmung aus Histogrammanalysen.....	98
4.2.3 Grauwert histogramme	99
4.2.4 Verallgemeinerung der Binarisierung.....	102
4.3 Konturverfolgung	104
4.3.1 Punktnachbarschaft.....	105
4.3.2 Erzeugung von Objektkonturen	107
4.3.3 Konturdarstellung	108
4.4 Template Matching	108
4.4.1 Funktionsprinzip	109
4.4.2 Optimierungsmöglichkeiten.....	113
4.4.3 Bemerkungen zum Template Matching	117
4.4.4 Kantenbasierte Objektlokalisierung	118
4.5 Kantendetektion	121
4.5.1 Kanten tastung in industriellen Bildszenen	121
4.5.2 Subpixelgenaue Kantenlokalisierung.....	122
4.6 Zusammenfassung	124
5. Kennzeichnungsidentifikation.....	127
5.1 Barcode-Identifikation	127
5.1.1 Prinzip der Barcodeerkennung auf Grauwertbasis.....	128

5.1.2 Barcodetypen	129
5.1.3 Beispiele für industrielle Barcode-Identifikation	131
5.1.4 Zweidimensionale Codes	133
5.2 Klarschrifterkennung	136
5.2.1 Lasergravierte Schrift auf einem IC	136
5.2.2 Basiskonfiguration der IC-Schrifterkennung	138
5.2.3 Prinzipieller Aufbau einer Klassifikationsanwendung	141
5.2.4 Positionskorrektur am IC	147
5.2.5 Verbesserung der Zeichenqualität	152
5.2.6 Optimierung im Betrieb	153
5.3 Erkennung genagelter Schrift auf Metall	155
5.3.1 Beleuchtung	155
5.3.2 Vorverarbeitung	156
5.3.3 Segmentierung und Klassifikation	157
5.4 Blockcodierungen auf Filmrollen	159
5.5 Druckqualitätskontrolle	162
5.5.1 Vorgehensweise	163
5.5.2 Druckqualitätskontrolle in Einzelbereichen	165
5.5.3 Druckqualitätskontrolle mit automatischer Teilung	166
5.6 Zusammenfassung	167
6. Querschnitt: Klassifikation.....	171
6.1 Was ist Klassifikation	171
6.2 Klassifikation als Funktionsapproximation.....	173
6.2.1 Grundbegriffe.....	174
6.2.2 Statistische Grundlagen	176
6.2.3 Konstruktion von Klassifikatoren	177
6.3 Multireferenzenklassifikatoren	179
6.3.1 Nächster-Nachbar-Klassifikator.....	179
6.3.2 RCE-Netze	181
6.3.3 Vektorquantisierung.....	182
6.3.4 Template Matching	183
6.3.5 Bemerkungen zu Multireferenzenklassifikatoren	183
6.4 Funktionalklassifikatoren.....	184
6.4.1 Polynomklassifikator	184
6.4.2 Neuronale Netze vom Typ Multilayer-Perzeptron.....	185
6.5 Bemerkungen zum Einsatz neuronaler Netze	189
6.5.1 Zusammensetzung der Lernstichprobe	189
6.5.2 Merkmalsskalierung.....	189
6.5.3 Rückweisung.....	190
6.5.4 Abgrenzung zu anderen Klassifikatoren	191
6.6 Zusammenfassung	192

7. Vermessung.....	195
7.1 Vermessungsaufgaben	196
7.2 Einfache Vermessung	196
7.2.1 Schwerpunktsabstände.....	198
7.2.2 Konturabstände	201
7.2.3 Winkelmessungen.....	205
7.3 Formkontrolle an Elektronikstanzteilen.....	207
7.3.1 Prüfaufgabe.....	207
7.3.2 Berechnung von Ausgleichsgeraden.....	208
7.3.3 Messung des Konturwinkels.....	210
7.4 Winkelmessung an einem Zahnriemen.....	212
7.4.1 Beleuchtungsaufbau.....	212
7.4.2 Kantenerzeugung	214
7.5 Formkontrolle an Spritzgussteil.....	215
7.5.1 Bestimmung von Radien.....	216
7.5.2 Bemerkungen zur Ausgleichskreisberechnung	219
7.6 Hochgenaue Vermessung eines Gewindeflansches	219
7.6.1 Beleuchtung und Bildaufnahme.....	220
7.6.2 Subpixelgenaue Vermessung der Gewindetiefe.....	221
7.7 Kalibrierung	223
7.7.1 Kalibriermodus	223
7.7.2 Prüfungsbezogene Kalibrierungen	224
7.8 Zusammenfassung	225
8. Querschnitt: Bildaufnahme und Beleuchtung	227
8.1 Halbleitersensoren für die Bildverarbeitung	227
8.1.1 Einführung	228
8.1.2 CCD-Sensoren	230
8.1.3 CMOS-Sensoren	233
8.1.4 Sonderbauformen.....	235
8.1.5 Farbsensoren	236
8.1.6 Eigenschaften von Sensoren	238
8.2 Digitalkameras	242
8.2.1 Steuerung der Bildaufnahme.....	243
8.2.2 Aufnahme von Farbbildern	245
8.2.3 Kennwerte von Digitalkameras.....	247
8.2.4 Einsatzbedingungen in Industrieumgebungen	249
8.3 Bilddatenübertragung	250
8.3.1 CameraLink®	251
8.3.2 FireWire®	253
8.3.3 USB	255
8.3.4 Gigabit-Ethernet	256
8.4 Zeilenkameras.....	259
8.4.1 Typen von Zeilenkamera-Anwendungen.....	259

8.4.2 Ortsauflösung bei Zeilenkameras.....	261
8.4.3 Belichtung bei Zeilenkameras.....	262
8.4.4 Steuerung von Zeilenkameras	263
8.5 Optische Grundlagen	264
8.5.1 Blendenzahl	265
8.5.2 Abbildungsgesetze	266
8.5.3 Schärfentiefe	272
8.5.4 Typische Aufnahmesituationen.....	276
8.5.5 Abbildungsfehler	278
8.5.6 Objektivbestimmung.....	279
8.5.7 Besondere Objektivtypen.....	282
8.6 Beleuchtungstechnik	283
8.6.1 Leuchtmittel.....	283
8.6.2 Auflicht	286
8.6.3 Durchlicht	289
8.7 Zusammenfassung	292
9. Anwesenheitskontrolle	297
9.1 Anwesenheitskontrolle mit Schwenk-Neigekopf-Zoomkameras.....	298
9.1.1 Prüfteilgeometrie	298
9.1.2 Beleuchtung	300
9.1.3 Positionierung	300
9.1.4 Objektdetektion.....	301
9.1.5 Verifikation des Ergebnisses.....	302
9.2 Einfache Vermessung zur Montagekontrolle	304
9.2.1 Beleuchtung	304
9.2.2 Prüfkriterien	306
9.2.3 Objekterzeugung und Maßberechnung	308
9.2.4 Positionskorrektur	309
9.3 Anwesenheitskontrolle mithilfe von Klassifikatoren	311
9.3.1 Beleuchtung	312
9.3.2 Kontrolle der Verstemmung	314
9.3.3 Typkontrolle des Flansches.....	320
9.4 Kontrastlose Anwesenheitskontrolle.....	324
9.5 Anwesenheitskontrolle mit Zeilenkameras	326
9.5.1 Prüfung zylindrischer Teile mit Flächenkameras	327
9.5.2 Prüfung eines Ventilkörpers	330
9.5.3 Bemerkungen	336
9.6 Zusammenfassung	338
10. Querschnitt: Objektmerkmale	339
10.1 Einfache geometrische Merkmale.....	339
10.1.1 Umschreibendes Rechteck	339
10.1.2 Fläche und Umfang	340

10.1.3 Schwerpunktskoordinaten.....	344
10.1.4 Achsen und Radien	345
10.2 Formbeschreibende Merkmale	347
10.2.1 Konturkrümmung	347
10.2.2 Fasermerkmale.....	350
10.2.3 Euler-Zahl	351
10.2.4 Momente und Fourierdeskriptoren	351
10.3 Grauwertmerkmale	352
10.3.1 Statistik erster Ordnung	352
10.3.2 Texturmerkmale.....	353
10.4 Zusammenfassung	354
11. Farbbildverarbeitung.....	357
11.1 Farbidentifikation	358
11.1.1 Auswertestrategie	359
11.1.2 Beleuchtung und Bilddaufnahme.....	360
11.1.3 Farbklassifizierung	362
11.1.4 Auswahl des Kamerabildes für die Beschriftungserkennung	366
11.1.5 Erkennung der Beschriftung	370
11.2 Farbsegmentierung	372
11.2.1 Beleuchtung	373
11.2.2 Farbklassifikation	373
11.2.3 Segmentierung	377
11.2.4 Anwesenheitsprüfung	377
11.3 Zusammenfassung	378
12. Realisierung von Bildverarbeitungsapplikationen	381
12.1 Bildverarbeitungsprojekte.....	381
12.2 Prozessintegration.....	383
12.3 Ausblick.....	387
A. Mathematische Ergänzungen	389
A.1 Backpropagation-Training	389
A.1.1 Neuronale Netze – Begriff und Geschichte	389
A.1.2 Grundlagen.....	390
A.1.3 Backpropagation	391
A.2 Herleitung der Schärfentiefe	395
A.2.1 Grenzabstände.....	395
A.2.2 Schärfentiefe im Unendlichen.....	398
A.2.3 Abhängigkeit der Schärfentiefe von der Brennweite	400
B. Software-Download	401
C. Weblinks zur industriellen Bildverarbeitung.....	403

Literaturverzeichnis	405
Index	407

1. Einleitung

Die industrielle Bildverarbeitung ist angesichts steigender Anforderungen an Produktqualität und -dokumentation zu einer Schlüsseltechnologie geworden. Ihr Einsatz in der automatischen Fertigung ist inzwischen selbstverständlich. Trotzdem mangelt es vielfach noch an Verständnis für diese moderne Technologie. Das vorliegende Buch soll dazu beitragen, diesem Zustand abzuhelpfen, den nicht zuletzt die Bildverarbeitungsindustrie teilweise selbst herbeigeführt hat. Wie in allen Bereichen, in denen zunehmend PCs eingesetzt werden, stellt man auch in der Bildverarbeitung den Trend fest, dem Endanwender immer größere Möglichkeiten zur Systementwicklung zu geben. Damit wird es jedoch auch erforderlich, ihm das entsprechende Know-How zur Verfügung zu stellen.

In diesem einleitenden Kapitel werden wir die typischen Aufgabenbereiche von Bildverarbeitungssystemen in der Industrie und ihren grundsätzlichen Aufbau darstellen, dann das objektorientierte Modell beschreiben, das unserer Vorgehensweise zugrundeliegt, und dieses Modell an einem einfachen Beispiel illustrieren. Zuvor wollen wir aber noch ein paar Worte darüber verlieren, warum wir es für notwendig hielten, den bereits erhältlichen Werken über industrielle Bildverarbeitung ein weiteres hinzuzufügen.

1.1 Warum noch ein Buch über Bildverarbeitung?

Es gibt eine Vielzahl von Büchern über digitale Bildverarbeitung, so dass die Frage berechtigt erscheint: warum noch ein weiteres? Nach unseren Erfahrungen lassen sich die verfügbaren Werke in drei Kategorien einteilen:

- Die meisten Werke stellen grundlegende Verfahren und Algorithmen der Reihe nach vor, mehr oder weniger mathematisch streng. Diese Bücher werden gewöhnlich von Hochschulangehörigen geschrieben und dokumentieren so die fortlaufende Forschung auf diesem Gebiet. Als solches sind sie für die Entwickler von Bildverarbeitungssoftware von unschätzbarem Wert. Dem Endanwender aber, der eine konkrete Sichtprüfaufgabe zu lösen hat, helfen sie nicht viel weiter. Er stellt seine Fragen notwendigerweise aus dem Blickwinkel seines Problems, nicht der vorgestellten Verfahren, die er als Nichtexperte ja gar nicht kennt. Außerdem besprechen diese Werke meist die verschiedenen Verfahren einzeln, voneinander isoliert. Die wirklichen Lösungen zeigen sich jedoch im Allgemeinen erst im Zusammenwirken der Methoden – und dabei kann es gelegentlich auch zu ganz überraschenden Ergebnissen kommen.

2 1. Einleitung

- Es gibt einige Bücher, die sich mit der konkreten Entwicklung und Implementierung von Bildverarbeitungssoftware befassen, zumeist in Form von Algorithmenbibliotheken. Wiederum für den Softwareentwickler von großer Bedeutung, helfen sie dem Endanwender vielleicht sogar noch weniger als die vorher genannten Werke, da implementatorische Details für die Lösung seiner Prüfaufgabe von höchst untergeordneter Bedeutung sind.
- Einige wenige Bücher stellen realisierte Anwendungen aus der Industrie vor, wobei aber häufig die Vorgehensweise nur sehr oberflächlich beschrieben werden kann. Das hängt damit zusammen, dass die Hersteller dieser Anlagen sich den Wettbewerbsvorteil erhalten wollen, den sie aus der Kapselung der gesamten Verfahrensweise in einer „Black Box“ gewinnen, die dem Endanwender nur ein sehr kleines Interface bietet, aus dem er keine Informationen über den internen Aufbau der Anlage gewinnen kann.

Im Gegensatz dazu sind wir der Überzeugung, dass die industrielle Bildverarbeitung die in sie gesetzten Erwartungen nur erfüllen und die Verbreitung erlangen kann, die ihrem Leistungspotential zukommt, wenn sie aus dem Status einer von wenigen beherrschten Geheimwissenschaft heraustritt und eine allgemein bekannte und beherrschte Technologie wird. Wir wollen mit diesem Buch zu einer solchen Entwicklung beitragen, indem wir funktionierende Lösungen realer Prüfprobleme beschreiben und so zeigen, wie man verschiedene wohlbekannte Algorithmen tatsächlich einsetzen kann, so dass sie sich gegenseitig unterstützen. Wir gehen dabei von einem gewissen Szenario der zukünftigen Entwicklung im Bildverarbeitungsbereich aus, das wir im Folgenden kurz erläutern wollen.

Allgemein anerkannte Standards sind eine unerlässliche Voraussetzung für die Verbreitung einer Technologie. Das wichtigste Hilfsmittel der industriellen Bildverarbeitung ist der Computer, und der am weitesten verbreitete Standard in diesem Bereich ist der PC mit Windows®-Betriebssystemen der Firma Microsoft (Redmond, WA, USA). Natürlich wird es immer Aufgaben geben, die die Leistungsfähigkeit eines PC-Systems sprengen, aber der Großteil industrieller Prüfaufgaben wird mit PCs gelöst. Dabei erweist sich die große Verbreitung von PCs im privaten Bereich genauso wie in Verwaltung und Industrie als zusätzlicher Beschleunigungsfaktor, da der Umgang mit den weitgehend standardisierten Benutzeroberflächen den meisten Menschen selbstverständlich vertraut ist.

In diesem Sinne hat sich bewahrheitet was (Jähne et al. 1995) bzgl. der Entwicklung von Bildverarbeitungssystemen vorausgesagt hat: „fallende Preise und die Möglichkeit, Bildverarbeitung auf einem gewohnten Rechnersystem durchführen zu können, werden die Bildverarbeitung zu einem ebenso allgemein eingesetzten und leicht bedienbaren Hilfsmittel machen, wie es die Messdatenerfassung schon heute ist.“

Die Bildverarbeitungssoftware ist denselben Weg wie die Software für Messdatenerfassung gegangen: hin zu interaktiven, leicht bedienbaren Systemen, die vom Endanwender parametriert und umgerüstet werden können. Damit ist auch eines der letzten Hindernisse für den Einsatz der industriellen Bildverarbeitung, vor allem in kleineren Betrieben, die meist mit häufigen Typwechseln und kleineren Se-

rien umzugehen haben, beseitigt: die erheblichen Wartungskosten, die dadurch entstanden, dass die bei einem Typwechsel unvermeidlichen Änderungen einer Prüfanlage teure Spezialisten und firmenfremdes Know-How erfordern.

Aller Fortschritt in der Bedienung darf jedoch nicht darüber hinwiegäuschen, dass die industrielle Bildverarbeitung kein einfaches Gebiet ist und dies auch in Zukunft nicht sein wird. Zu groß ist die Anzahl der Einflussgrößen, zu komplex sind die Zusammenhänge zwischen Prüfteil, Umgebung und Software. Wie stets ist auch hier Erfahrung durch nichts zu ersetzen. Besonders beim anfänglichen Aufbau einer Prüfanlage wird daher der Spezialist auch weiterhin gefragt sein. Dieses Buch soll einerseits dabei helfen, einen Einstieg in das Gebiet der Bildverarbeitung zu finden, um sich selbst mit der Zeit zu einem Spezialisten entwickeln zu können. Andererseits soll es einen Überblick geben, anhand dessen man die technischen Schwierigkeiten und die Realisierung einer Sichtprüfanlage verstehen und beurteilen kann, auch wenn man sich nicht so tief mit diesem Gebiet auseinandersetzen will, wie es nötig ist, um selbst Prüfanlagen zu erstellen.

Digitale Bildverarbeitung ist ein anschauliches Gebiet. Gerade in einem solchen Gebiet kann man am besten durch Beispiele lernen, und darum bilden Beispiele auch den Kern dieses Buches, die Motivation für theoretische Erklärungen und Algorithmenbeschreibungen. Als Download steht Ihnen als Leser alles zur Verfügung, was Sie benötigen, um auf einem handelsüblichen PC mit aktuellem Windows-Betriebssystem diese Beispiele unmittelbar nachzuvollziehen, und zwar – und das dürfte derzeit einzigartig sein – mit genau dem Programmsystem, das auch für die industrielle Realisierung dieser Prüfaufgaben eingesetzt wurde. Diesem beispielorientierten, eher „spielerischen“ Ansatz entsprechend werden Sie keine bis in die letzten Einzelheiten gehenden Beschreibungen von Algorithmen in diesem Buch finden. Natürlich werden die wesentlichen Verfahren und ihre mathematischen Grundlagen vorgestellt, aber unser Ziel ist es, den Sinn und Zweck, den Einsatz und die Wirkung dieser Verfahren zu verdeutlichen, nicht ihre mathematische Begründung.

Die Fragestellung, die wir mit diesem Buch beantworten wollen, ist also nicht „Was ist ein Hammer, wie stelle ich ihn her und wie schlage ich damit einen Nagel ein?“ sondern „Ich habe einen Werkzeugkasten mit Hammer, Nägeln und anderen Werkzeugen; wie baue ich damit ein Möbelstück oder gar ein Blockhaus?“ Dabei ist es unvermeidlich, dass wir gelegentlich vorgreifen müssen, Verfahren benutzen müssen, die erst im weiteren Verlauf ausführlich vorgestellt werden, aber im Interesse realistischer Beispiele hielten wir dies für gerechtfertigt.

1.2 Möglichkeiten und Grenzen

Gegenüber einigen anderen, verwandten Gebieten befindet sich die industrielle Bildverarbeitung in einer wenig beneidenswerten Lage, und dies ausgerechnet aufgrund der größeren Anschaulichkeit ihres Gegenstandes. Die meisten potentiellen Anwender automatischer Prüfsysteme sind ohne weiteres bereit, die Schwie-

rigkeiten anzuerkennen, die bei der Interpretation endloser Messreihen unterschiedlicher Größen entstehen. Selbst bei Klangdaten – für die der Mensch ja selbst einen Sinn besitzt – werden diese mathematischen Schwierigkeiten gewöhnlich durchaus gewürdigt. Der Hersteller von Bildverarbeitungssystemen ist hingegen nur zu oft mit dem Argument konfrontiert „Das sieht man doch!“ Ja, natürlich, ein Mensch sieht das sofort, aber für eine Maschine ist das, was der Mensch in Jahrmillionen der Evolution gelernt hat, normalerweise ganz und gar nicht trivial. Darin besteht dann auch das Hauptproblem bei der Realisierung automatischer Sichtprüf anlagen: sich klarzumachen, auf welche Weise die Maschine „sieht“, und Bedingungen zu schaffen, unter denen sie optimal arbeiten kann.

In direktem Zusammenhang mit diesem Problem tritt eine weitere Schwierigkeit bei der Einführung von Bildverarbeitungsanlagen auf: Sie werden unweigerlich mit Menschen auf dem Gipfel ihrer Leistungsfähigkeit verglichen. Es ist natürlich richtig, dass Menschen im Allgemeinen Schriftzeichen fehlerfrei erkennen können, bei entsprechender Übung auch schwer leserliche Handschriften. Man kann daher durchaus, bei entsprechender Konzentration, von einer hundertprozentigen Trefferquote sprechen. Über einen ganzen Arbeitstag hinweg ist diese Leistung jedoch keineswegs aufrechtzuerhalten. Obwohl gedruckte Schriftzeichen prinzipiell leichter zu lesen sind, kann man davon ausgehen, dass die Fehlerquote bei derartigen Sichtprüfungen in der Industrie eher noch höher liegen wird, da die Arbeit sehr monoton abläuft und daher die Konzentration stärker nachlässt.

Über die Möglichkeiten des menschlichen visuellen Systems und die Unterschiede zur computerisierten Verarbeitung von Bildinformationen ließen sich ohne weiteres mehrere Bücher schreiben. Dies kann aber nicht die Aufgabe dieser praxisorientierten Einführung sein, daher wollen wir nur kurz die Kernaussage zusammenfassen: Automatische Sichtprüf anlagen sind in der Lage, konstant und zuverlässig ausgezeichnete Erkennungsleistungen zu liefern, die im zeitlichen Mittel nicht hinter der von Menschen zurückbleiben und sie in bestimmten Bereichen sogar deutlich übertreffen, vorausgesetzt man beachtet folgende Grundregeln:

- Die Prüfaufgabe wurde detailliert und präzise beschrieben, in einer den Eigenarten des maschinellen „Sehens“ angemessenen Weise.
- Alle zulässigen Varianten von Prüfteilen (hinsichtlich Form, Farbe, Oberflächenbeschaffenheit etc.) und Fehlern wurden erfasst.
- Die Umgebungsbedingungen (Beleuchtung, Bildaufnahme, mechanisches Umfeld etc.) wurden so gestaltet, dass die festzustellenden Fehler in maschinell erfassbarer Weise hervortreten.
- Diese Umgebungsbedingungen werden stabil gehalten.

Man muss sich unbedingt darüber klar sein, dass für eine automatische Sichtprüf anlage *wie für jede andere Maschine* bestimmte Spezifikationen gelten, außerhalb derer man kein einwandfreies Funktionieren erwarten kann. Es ist erstaunlich, wie oft diese einfache Grundregel bei Systemen, die wesentlich auf Software basieren, missachtet wird. Niemand würde von einer Bohrmaschine mit einem Holzbohrer erwarten, Stahlbeton zu durchschlagen, aber von einem Programm wird erwartet,

dass es mit Eingangsdaten zurechtkommt, die mit seinem ursprünglichen Einsatzzweck nicht das Geringste zu tun haben. Das hängt natürlich auch damit zusammen, dass die Aufgaben der Anlage und die möglichen Zustände der zu prüfenden Teile in der Regel seitens des Anwenders nicht hinreichend präzise und detailliert vorgegeben werden – ungeachtet der Mitwirkungspflicht des Auftraggebers einer solchen Anlage.

1.3 Typen von Sichtprüfaufgaben

Man kann verschiedene Typen von Sichtprüfaufgaben unterscheiden, einmal nach dem damit verfolgten Ziel, andererseits nach der Struktur der Vorgehensweise.

1.3.1 Einteilung nach dem Prüfziel

(Steger et al. 2008) unterteilen die Aufgaben für Bildverarbeitungssysteme in der industriellen Produktion in folgende Klassen:

- Objekterkennung
- Lageerkennung
- Vollständigkeitsprüfung
- Form- und Maßprüfung
- Oberflächeninspektion

Im Wesentlichen folgen auch wir dieser Einteilung. Es ist allerdings zu berücksichtigen, dass Objekterkennung ein Bestandteil vieler Anwendungen ist, ohne jedoch das eigentliche Ziel der jeweiligen Sichtprüfaufgabe zu sein. Wir sind deshalb an dieser Stelle von der Einteilung nach Aufgabenklassen abgewichen und haben uns mehr an der zugrundeliegenden Technologie orientiert, mit der ein Objekt speziell zum Zwecke der Identifikation markiert werden kann. Die Auflistung ergänzen wir durch zwei Bereiche, die durch die rasante technische Entwicklung der letzten Jahre in den Fokus gerückt sind: die Farbbildverarbeitung und die 3D-Bildverarbeitung. Dabei handelt es sich nicht um andere Aufgaben, aber um andere Arten von Informationen, deren Auswertung bzw. Erfassung besondere Methoden erfordern. Außerdem haben wir den Bereich *Bild- und Objektvergleiche* hinzugefügt, da sich bestimmte Aufgaben der Vollständigkeitsprüfung so leichter beschreiben lassen. Damit kommen wir zu folgender Einteilung:

- Lageerkennung
- Kennzeichnungsidentifikation
- Form- und Maßprüfung bzw. Vermessung
- Vollständigkeitsprüfung

6 1. Einleitung

- Farbverarbeitung
- Bild- und Objektvergleich
- Oberflächeninspektion
- 3D-Bildverarbeitung

Die Aufgabenbereiche sind hier in der Reihenfolge aufgeführt, in der sie in den folgenden Kapiteln besprochen werden. Wir beginnen mit der Lageerkennung, da sie strukturell eine der einfachsten Aufgabe darstellt: Wenn das Objekt erst einmal gefunden ist, muss nur noch ein Schritt durchgeführt werden, eben die Bestimmung der Position. Die Vollständigkeitsprüfung haben wir demgegenüber weiter nach hinten verschoben, weil dies, so einfach der Begriff zunächst klingt, im Einzelfall eine sehr komplexe Anwendung sein kann.

Im Interesse einer geschlossenen Darstellung der Vorgehensweise und um den geplanten Rahmen nicht zu sprengen, beschränken wir uns auf die ersten fünf Aufgabenbereiche, die auch die typischen Anwendungen von PC-basierten Bildverarbeitungssystemen darstellen. Der Bildvergleich kommt in Form der Druckqualitätsprüfung im Zusammenhang mit Kennzeichnungidentifikation zur Sprache. Einen kurzen Einblick in die Oberflächeninspektionen werden wir im Kapitel über Anwesenheitskontrolle geben. In den letzten Jahren ist die 3D-Bildverarbeitung Gegenstand großer Aufmerksamkeit geworden, dennoch ist dieses Gebiet nach wie vor geprägt von einer Vielfalt von Erfassungstechniken, jede mit ihren spezifischen Vor- und Nachteilen, von denen wir eines Exemplarisch im Querschnittskapitel über Bildaufnahme und Beleuchtung darstellen werden. Die Auswertung der 3D-Bilddaten erfolgt oft mit den Methoden der klassischen zweidimensionalen Bildverarbeitung, wobei die Helligkeitsinformationen durch Entfernungsinformationen ersetzt werden, so dass die in diesem Buch dargestellten Auswertungsmethoden im Allgemeinen auch für 3D-Bilder verwendet werden können.

Zwischen die einzelnen Anwendungskapitel haben wir Querschnittskapitel eingeschoben, die einen bestimmten Aspekt aus dem vorangegangenen Anwendungskapitel aufgreifen und ausführlicher beleuchten. Dort werden dann auch die Verfahren, die in den Anwendungskapiteln zumeist einfach nur verwendet werden, näher erläutert.

1.4 Aufbau von Bildverarbeitungssystemen

Dieser Abschnitt gibt einen kurzen Überblick über den grundsätzlichen Aufbau von Bildverarbeitungssystemen in der industriellen Fertigung. Diese Übersicht ist nur als erste Orientierung gedacht und geht darum auch nicht auf Einzelheiten wie die Realisierung von Beleuchtungssystemen, die Eigenschaften der verwendeten Kameras oder die Kommunikation mit der übergeordneten Fertigungssteuerung ein. Diese Aspekte werden in den Kapiteln 8 und 11 ausführlich besprochen.

1.4.1 Hardwareaufbau

Nahezu jedes Bildverarbeitungssystem zerfällt grob in drei Blöcke: Sensoren, Rechner und Kommunikationseinheiten, wie die schematische Darstellung in Abbildung 1.1 zeigt. Dabei wurde der für den Erfolg jeder Bildverarbeitungsanwendung entscheidende Bereich der Beleuchtung vorläufig ausgeklammert, da er für eine geschlossene Beschreibung des Systemaufbaus einfach zu schwer zu verallgemeinern ist. Wir werden versuchen, dies im Kapitel 8 nachzuholen.

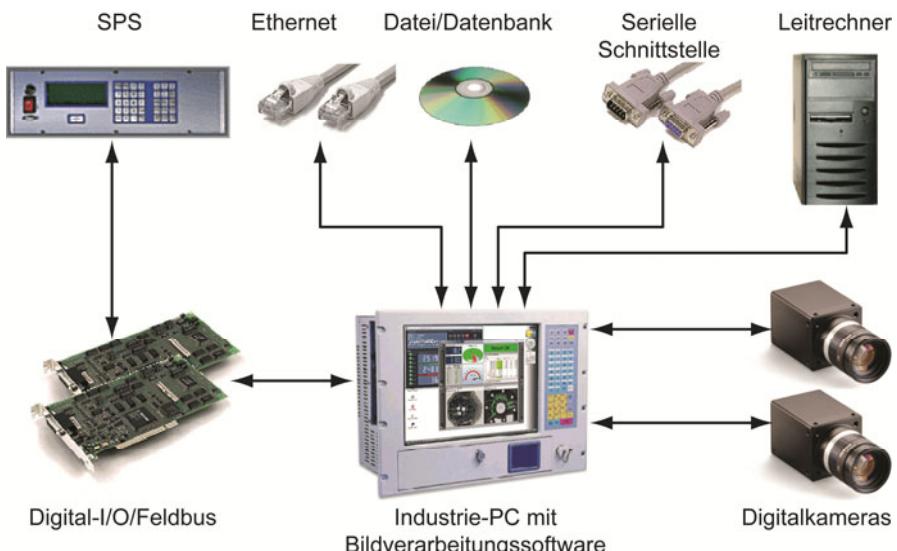


Abbildung 1.1: Industrielles Bildverarbeitungssystem

Sensoren: Bei einem System für die industrielle Qualitätskontrolle, wie es schematisch in Abbildung 1.1 dargestellt ist, sind die Sensoren in der Regel Kameras, wenngleich auch andere bildgebende Sensoren Verwendung finden können, wie z. B. Laserabtast- und Ultraschallsensoren. Scanner, wie sie im grafischen Gewerbe oder bei der Analyse fotografischen Materials, etwa bei der Satellitenbilddauswertung, üblich sind, trifft man im industriellen Einsatz schon aus Geschwindigkeitsgründen praktisch nicht an. Kapitel 8 beschäftigt sich näher mit Fragen der Kameratechnik.

Die Verbindung zwischen Sensoren (also Kameras) und Rechner wird gewöhnlich über digitale Übertragungsmedien wie FireWire, Gigabit-Ethernet oder USB hergestellt. Diese aus dem PC Massenmarkt kommenden Technologien haben sich in den letzten Jahren in der industriellen Anwendung durchgesetzt, womit das im PC-Bereich schon länger zu beobachtende Phänomen, dass die Umsätze aus dem Massenmarkt dazu dienen, die technologische Gesamtentwicklung voranzutreiben, seine logische Fortsetzung fand.

Rechner: Je nach Aufgabenstellung können ganz verschiedene Rechnertypen zum Einsatz kommen: Bei den sehr datenintensiven Auswertungen kontinuierlicher Fertigungsprozesse, wie etwa der Herstellung von Stahl, Papier oder Textilien trifft man nach wie vor spezialisierte Parallelrechner an, da die dabei anfallenden Datenraten sowohl vom Speicherdurchsatz als auch von der Rechenzeit her weder von Workstations noch von PC-Systemen zu bewältigen sind. Der Großteil industrieller Prüfaufgaben ist jedoch mit handelsüblichen PCs und Standardkomponenten zu bewältigen. Durch die Verwendung von modernen Multi-Core-CPUs ergibt sich ein weiterer Quantensprung in der Leistungsfähigkeit von PC-Systemen, von dem gerade die Bildverarbeitung massiv profitiert. Denn insbesondere die zeitaufwändigen Berechnungen auf Bilddaten können, eine entsprechende intelligente „Multi-Threaded“ Implementierung vorausgesetzt, auf die Kerne des Prozessors verteilt werden, was häufig zu deutlich verkürzten Auswertezeiten führt.

Bis Mitte der 90er Jahre waren PC-Systeme vor allem aufgrund des unzureichenden Datendurchsatzes ihrer Bussysteme in der Bildverarbeitung kaum konkurrenzfähig, VME-Bus-Systeme und Spezialprozessoren beherrschten den Markt. Kein Sektor der Informationstechnologie hat sich jedoch in den letzten Jahrzehnten so rasant entwickelt wie der PC-Bereich. Die Leistungssteigerungen bei gleichzeitigem Preisverfall ermöglichen die Lösung auch anspruchsvoller Bildverarbeitungsaufgaben auf PC-Basis. Man erlebt dabei einen Selbstbeschleunigungseffekt: Die weite Verbreitung der PC-Architektur sorgt dafür, dass sich aufwändige Hard- und Softwareentwicklung lohnt, wodurch neue Leistungsbereiche erschlossen werden, die wiederum die Attraktivität und damit die Verbreitung der PC-Plattform erhöhen. Auch die hohe Standardisierung nicht nur von Hardware-, sondern auch von Softwareschnittstellen, trug mit dazu bei, dass PC-Systeme heute eine bedeutende Rolle in allen industriellen Bereichen, von der Fertigungssteuerung bis zur Qualitätskontrolle, spielen. Auch im Bereich der speicherprogrammierbaren Steuerungen (SPS) wird häufig auf PC-Technologie zurückgegriffen.

Kommunikation: Ein Bildverarbeitungssystem zur industriellen Qualitätskontrolle muss im Gleichtakt mit dem Produktionsprozess arbeiten, d. h. es muss sich von außen steuern lassen, und es muss seine Ergebnisse auch nach außen in einer Form melden können, die in den automatischen Fertigungsanlagen und Qualitätssicherungssystemen verarbeitet werden kann. Das System muss also in der Lage sein, mit anderen Geräten zu kommunizieren.

Für die Fernsteuerung und die unmittelbare Auswertung von Endergebnissen (Test bestanden oder nicht) werden Bildverarbeitungssysteme häufig über Digitalschnittstellen oder Feldbus mit speicherprogrammierbaren Steuerungen verbunden. Genauso gut ist aber auch die direkte Verbindung zu Leitrechnern über Netzwerke oder serielle Kommunikation denkbar. Beide Kommunikationsformen können auch durchaus nebeneinander existieren, wobei dann die SPS unmittelbar für die Synchronisation mit dem Produktionsprozess verantwortlich ist, während dem Leitrechner die globale Steuerung und die Erfassung von Qualitätsdaten obliegt. Qualitätsrelevante Daten, Messwerte und dergleichen können natürlich auch vom Bildverarbeitungssystem selbst direkt in Dateien zur weiteren Auswertung

festgehalten werden. Durch die Verwendung standardisierter Dateiformate lassen sich diese Daten praktisch überall weiterverarbeiten, worin sich ein weiterer Vorteil der Verbreitung von PC-Systemen zeigt. Wenn man dies noch einen Schritt weiterdenkt, kommt man zum Konzept der Fernwartung von Anlagen direkt über das Internet. Diese Form der Kundenunterstützung hat eine große Bedeutung bei der Betreuung von Bildverarbeitungssystemen gewonnen, da sie dem Lieferant die Möglichkeit eröffnet, seinen Kunde innerhalb von Minuten auch über große Entfernnungen online zu unterstützen. Trotz dieser offensichtlichen Vorteile werden aber aufgrund von Sicherheitsbedenken immer wieder Sichtprüfauflagen ohne externen Netzwerkzugang erstellt, bei denen also auf die Möglichkeit der Fernwartung aktuell noch bewusst verzichtet wird. Aufgrund der massiven wirtschaftlichen Vorteile ist aber davon auszugehen, dass mittelfristig die meisten Systeme mit der Option zur Fernwartung ausgerüstet werden.

Intelligente Kameras: An dieser Stelle ist noch eine Bemerkung zu dem seit Mitte des Jahrzehnts zu beobachtenden Trend zu intelligenten Kameras angebracht. Prinzipiell folgen auch diese dem oben angedeuteten Hardwareaufbau, nur dass der Rechner im Kameragehäuse integriert ist. Vorteile dieser Systeme liegen in ihrer geringen Baugröße und dem typischerweise niedrigen Anschaffungspreis, der sie besonders für kleine und mittlere Unternehmen als Einstieg in die Bildverarbeitung attraktiv erscheinen lässt. Andererseits sind die Rechenleistungen und vor allem die Speicherkapazität dieser Kameras gerade wegen der geringen Baugröße begrenzt, so dass sie sich nur für relativ einfache Anwendungen eignen. Je nach Modell muss man sie entweder regelrecht programmieren, meist in C, oder hat nur wenige einstellbare Parameter zur Verfügung. Es handelt sich dabei praktisch um eine Miniaturisierung des alten Konzeptes der „Black-Box“, die dem Anwender nur sehr wenig Informationen gibt. Der Aufbau leistungsfähiger, objektorientierter Prüfanwendungen ist so kaum möglich. Zudem können diese Systeme den Prüfprozess und die Prüfergebnisse für das Bedienpersonal oft nur eingeschränkt visualisieren.

Der oft genannte zusätzliche Vorteil der einfacheren Bedienung gegenüber einem PC-System wird allerdings dadurch relativiert, dass die Konfiguration von Prüfanwendungen nicht direkt an der Kamera erfolgen kann, sondern oft einen zusätzlichen PC als Terminal voraussetzt. Die Prüfanwendung wird dann, sei es als C-Programm, sei es als Parametrierung vorgefertigter Routinen, die im Rechner der Kamera zur Verfügung stehen, auf dem PC konfiguriert und meist seriell in die Kamera übertragen. Das hat zur Folge, dass häufiges Umkonfigurieren, wie es stets in der Anlaufphase eines Produktionsprozesses anfällt, aber auch später aufgrund von Änderungen in der Produktpalette oder einfache Prozessdrift der Fertigung, recht umständlich ist.

Damit soll den intelligenten Kameras nicht ihre Berechtigung abgesprochen werden. Man muss sich nur genau darüber klar sein, wozu ein solches System imstande ist und auf welche Leistungen, die bei PCs heute selbstverständlich geworden sind, man verzichten muss. Die Berechnung der Wirtschaftlichkeit von intelligenten Kameras muss man üblicherweise wiederholen, wenn mehr als zwei bis

drei Kameras zum Einsatz kommen. So kann ein PC, der mit der entsprechenden Software ausgestattet ist, problemlos die Bilder von einem Dutzend Kameras oder mehr auswerten.

1.4.2 Signalfluss in der Prozessumgebung

Der Zweck eines industriellen Bildverarbeitungssystems, dessen Grundaufbau im vorigen Abschnitt kurz beschrieben wurde, besteht darin, aus einer Bildszene, d. h. einem in der realen Welt existierenden Gebilde, eine Qualitätsaussage zu gewinnen. In größtmöglicher Vereinfachung stellt sich demnach der Signalfluss eines Bildverarbeitungssystems wie in Abbildung 1.2 dar. Die Abbildung 1.2 zeigt, dass ein Bildverarbeitungssystem über wenigstens zwei Schnittstellen mit der Außenwelt verbunden ist. Weitere Schnittstellen, über die das System ferngesteuert oder manuell bedient werden kann, sind natürlich denkbar; diese zwei sind jedoch für den Einsatz eines industriellen Bildverarbeitungssystems unabdingbar: am Eingang des Systems wird die Realweltszene in ein Bild übersetzt, das vom Rechner bearbeitet werden kann, am Ausgang wird das Ergebnis dieser Bearbeitung in Form einer Qualitätsaussage an die Umgebung übermittelt.

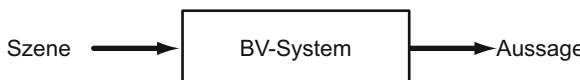


Abbildung 1.2: Schematischer Signalfluss eines industriellen Bildverarbeitungssystems

Ausgangsschnittstelle: Die Qualitätsaussage kann dabei sowohl inhaltlich als auch signaltechnisch ganz verschiedene Formen annehmen; sie kann ein Zahlenwert sein oder eine Gut/Schlecht-Aussage, eine Zeichenkette oder etwas völlig anderes; sie kann über eine Datenleitung übertragen, ausgedruckt, in einer Datei gespeichert oder als Lichtsignal dargestellt werden. Dies alles ist gänzlich von der Aufgabe selbst und der Prozessumgebung abhängig. In jedem Fall muss aber der Übermittlung einer Aussage an die Außenwelt irgendeine Art von symbolischer Darstellung im Bildverarbeitungssystem vorangehen.

Eingangsschnittstelle: Auf der Eingangsseite können, wie schon im vorigen Abschnitt erwähnt, ganz verschiedene Sensoren zur Bereitstellung der Bildinformation verwendet werden, das Ergebnis ist aber in aller Regel stets von derselben Form: ein digitales Bild, in dem die Helligkeitswerte der Bildszene als Zahlenwerte codiert sind (das gilt auch für Farbbilder, allerdings wird dann für jede der Grundfarben Rot, Grün und Blau ein eigener Helligkeitswert vorhanden sein, die gemeinsam eine Farbe ergeben). Ein digitales Bild formt also eine Matrix von Helligkeitswerten. Dieses Bild stellt natürlich nur in gewissen Grenzen eine getreue Abbildung der Realität dar. Zwei Faktoren sind dabei von besonderer Bedeutung: Rasterung und Quantisierung.

Rasterung: Digitale Bilder verfügen nur über eine begrenzte Anzahl von Bildpunkten¹. Die reale Bildszene wird folglich bei der Digitalisierung in ein begrenztes Raster von Bildpunkten hineingepresst. Kapitel 8 wird sich näher mit den Eigenarten dieses Vorgangs befassen, sowohl in mathematischer als auch in technischer Hinsicht. An dieser Stelle möge der Hinweis genügen, dass der unvermeidlich eintretende Informationsverlust sich wie stets in der digitalen Signalverarbeitung nach dem Shannonschen Abtasttheorem richtet, dessen Interpretation hier sehr einfach ist: Das Abtastintervall, d.h. der Abstand zwischen zwei Punkten im digitalen Bild, darf höchstens halb so groß sein wie das kleinste relevante Detail der Bildszene (Sonka et al. 2008). Dieser Umstand ist vor allem für die Lösung von Vermessungsaufgaben von großer Wichtigkeit, da dadurch die theoretische Messgenauigkeit begrenzt wird.

Abbildung 1.3 zeigt prinzipiell den Effekt der Rasterung bei der Überführung des Bildes eines Buchstabens in ein digitales Bild begrenzter Auflösung. Beachten Sie, dass das Originalbild nur zwei Helligkeitsstufen enthält, ein mittleres Grau als Hintergrund, Schwarz für den Buchstaben, das Ergebnis jedoch Zwischentöne aufweist, die durch die Verteilung des Lichts auf verschiedene Zellen des Sensors entstehen. Dieser Effekt erlaubt eine teilweise Wiedergewinnung der verlorenen Information, was im Kapitel 7 ausführlicher dargestellt wird.

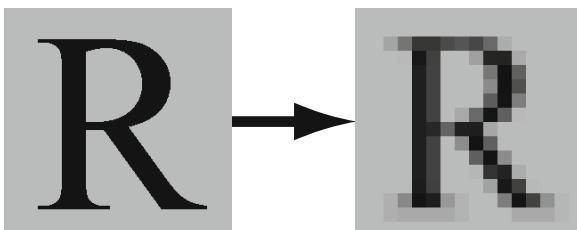


Abbildung 1.3: Effekt der Rasterung in einem digitalen Bildverarbeitungssystem

Quantisierung: Ein Punkt in einer realen Bildszene kann jede beliebige Helligkeit aufweisen. Innerhalb gewisser Grenzen, gegeben durch das Grundrauschen und die Sättigung des Sensors, gilt dies auch noch innerhalb der Kamera (wenn wir einmal als gegeben annehmen, dass in der industriellen Bildverarbeitung Bilder üblicherweise mit CCD- oder CMOS-Kameras aufgenommen werden). Im Rechner aber werden die Helligkeiten der Bildpunkte als Zahlen dargestellt, und der Wertevorrat eines Computers ist zwangsläufig begrenzt. Aus verschiedenen Gründen, wie z. B. Speicherplatzbegrenzungen, Rechenzeit, bequemere Verarbeitung oder auch einfach, weil die Aufgabe keinen höheren Aufwand erfordert, verwendet man in der industriellen Bildverarbeitung sogar noch eine deutlich kleinere Menge von Zahlenwerten, als theoretisch möglich wäre. Heutige Schwarzweiß-

¹ Tatsächlich gilt das für jedes Bild, auch solche, die vom menschlichen Auge oder einem Fotoapparat aufgenommen wurden, allerdings ist deren Auflösung um ein Vielfaches höher, so dass sie als kontinuierlich empfunden werden.

Kameras können zwar Auflösungen von zehn oder zwölf Bit liefern, also 1024 bzw. 4096 Helligkeitswerte erzeugen, allerdings wird in der industriellen Bildverarbeitung auch heute noch typischerweise mit einer Auflösung von acht Bit gearbeitet. Das ursprüngliche Helligkeitssignal wird also üblicherweise in 256 Stufen quantisiert. Farbkameras können entsprechend zehn oder zwölf Bit Helligkeitsauflösung für jede Grundfarbe erzeugen, von denen aber in der industriellen Bildverarbeitung meistens ebenfalls nur acht Bit, also 256 Abstufungen, je Grundfarbe verwendet werden. Bei Spezialanwendungen, wie etwa in der Spektroskopie oder der Untersuchung von Entfernungs-Bildern werden Bilder mit deutlich höherer Graustufenauflösung verwendet, allerdings muss dafür ein erhöhter technischer Aufwand getrieben werden. Die Verwendung von 256 Graustufen ist übrigens auch deshalb so beliebt, weil sie der typischen Speicherorganisation heutiger Computer entspricht, die in einem Byte, d. h. einer Speicherzelle, gerade 256 verschiedene Werte darstellen können.

Die Reduzierung einer unendlichen Anzahl möglicher Helligkeiten auf 256 Graustufen klingt allerdings dramatischer als sie ist. Tatsächlich erreicht das menschliche Auge selbst keine bessere Helligkeitsauflösung. Nach (Russ 2007) können auf einem Videomonitor oder einer Fotografie kaum mehr als 30 Graustufen visuell unterschieden werden. Typische Ausgabegeräte, wie Laser- oder Tintenstrahldruckern, unterliegen derselben Begrenzung. Abbildung 1.4 zeigt – stark übertrieben, damit es trotz der erwähnten Begrenzungen bei der drucktechnischen Wiedergabe sichtbar wird – den Effekt der Graustufenquantisierung anhand des gerasterten Buchstabens aus Abbildung 1.3.

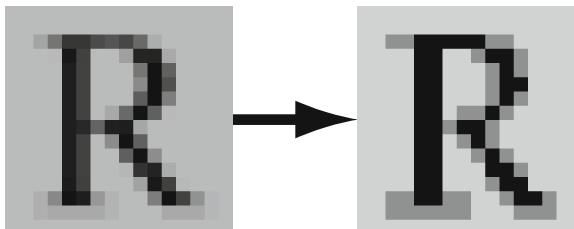


Abbildung 1.4: Effekt der Quantisierung in einem digitalen Bildverarbeitungssystem.

Um noch deutlicher zu machen, welche Bedeutung der Informationsverlust durch Rasterung und Quantisierung in einer Bildverarbeitungsanwendung haben kann, zeigt Abbildung 1.5 das Bild einer Zündkerze, wie sie im Einführungsbeispiel in Abschnitt 1.6 verwendet wird. Von links nach rechts sehen Sie das Originalbild, das Ergebnis einer Rasterung, und das einer Quantisierung des gerasterten Bildes. Nach der Rasterung ist es bereits unmöglich, die kleineren Buchstaben auf der Zündkerze zu lesen oder einzelne Spitzen des Gewindes exakt zu erkennen. Nach der Quantisierung lässt sich an einigen Stellen das Objekt bereits nicht mehr vom Hintergrund trennen. Man sieht also, dass das Zusammenspiel von ausreichender Orts- und Helligkeitsauflösung entscheidend dafür ist, ob sich eine Prüfaufgabe

lösen lässt. Informationen, die an dieser Stelle nicht mehr vorhanden sind, lassen sich nur in begrenztem Umfang wiedergewinnen.

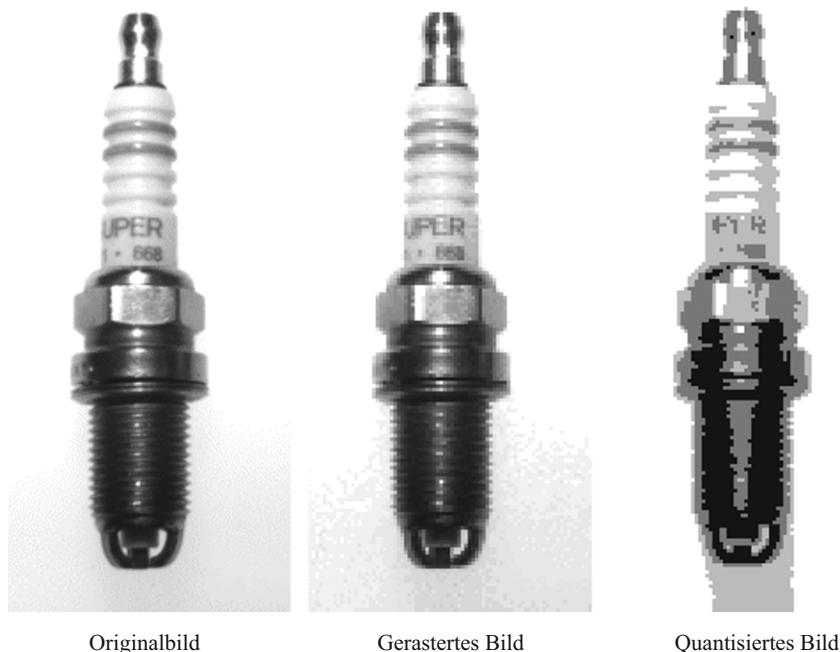


Abbildung 1.5: Effekt von Rasterung und Quantisierung auf ein Prüfbild

1.4.3 Signalfluss innerhalb des Bildverarbeitungssystems

Wie in Abschnitt 1.4.2 dargestellt, beginnt die Verarbeitung innerhalb des Bildverarbeitungssystems mit einem gerasterten und quantisierten Bild, wie es vom Sensor geliefert wird, und endet mit einer symbolischen Darstellung, die als Aussage an die Peripherie übermittelt werden kann. Für den oben schon gezeigten Fall eines einzelnen Buchstabens kann man sich das etwa wie folgt vorstellen:

- Am Anfang steht das gerasterte und quantisierte Bild des Buchstabens. Dies bezeichnet man auch als die *ikonische*² Ebene, da hier nur ein Bild ohne weitere Informationen über den Inhalt zur Verfügung steht.
- Da der Computer nicht wie ein Mensch in der Lage ist, „mit einem Blick“ zu erkennen, dass es sich hier um einen bestimmten Buchstaben handelt, wird dieses Bild einer Reihe von Bearbeitungsschritten unterzogen, an deren Ende die-

² griechisch *εικων* (eikon): Bild

ser Buchstabe durch ein Objekt repräsentiert wird. Der Begriff *Objekt* ist hier sehr stark im Sinne der Informatik gebraucht, d. h. aus der (datentechnisch) unstrukturierten Darstellung des Buchstabens als Matrix aus Helligkeitswerten ist eine Datenstruktur geworden, die Eigenschaften eines in der Bildszene aufgefundenen Objekts beschreibt, wie etwa seine Position, Größe, Form etc. Diesen Vorgang, aus der anfänglich unstrukturierten Bildfläche identifizierbare Objekte herauszulösen, bezeichnet man als *Segmentierung*.

- Schließlich wird aus den Eigenschaften eines oder mehrerer solcher aus der Bildszene segmentierter Objekte die Information gewonnen, die als Aussage an die Umgebung übermittelt wird. In diesem Fall handelt sich um die Bezeichnung des dargestellten Buchstabens, in anderen Fällen könnte es sich um ein Längenmaß oder eine Objektanzahl handeln. In jedem Fall aber handelt es sich um eine Information, die in Symbolen dargestellt werden kann, die für Menschen verständlich sind. Daher spricht man hier auch von der *symbolischen* Ebene der Bildverarbeitung.

Abbildung 1.6 illustriert diesen Verarbeitungsprozess. Der Übergang von der rein ikonischen Ebene zur Ebene der Objektbeschreibung mittels Segmentierung wird dadurch symbolisiert, dass nur mehr der Buchstabe selbst, einheitlich schwarz und ohne Hintergrund dargestellt wird, während die Gewinnung der symbolischen Information durch die Verwendung eines echten Buchstabens aus einem Computerzeichensatz symbolisiert ist.

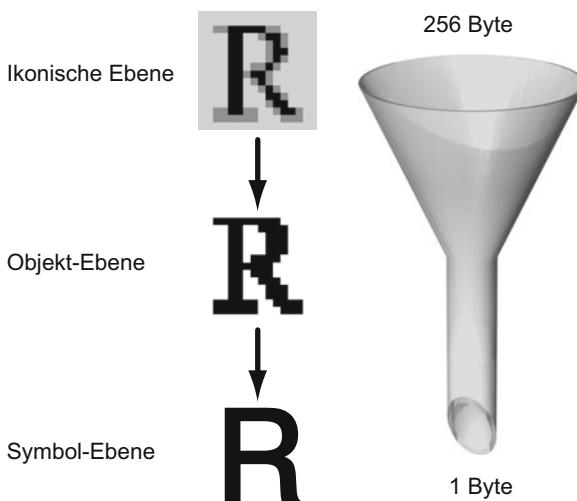


Abbildung 1.6: Signalfluss in einem Bildverarbeitungssystem

Abbildung 1.6 zeigt außerdem, dass in diesem Verarbeitungsprozess eine erhebliche Informationsreduktion stattfindet. Am Anfang steht das Bild eines Zeichens mit insgesamt 256 Punkten in einem Raster der Größe 16×16 . Wenn die verbreitete Darstellung jedes Punkts durch ein Byte (mit 256 möglichen Helligkeitswerten)

ten) verwendet wird, entspricht dies einer Informationsmenge von 256 Byte. Am Ende der Kette steht die symbolische Information, dass es sich um den Buchstaben „R“ handelt. Diese Information kann in gängigen Computerzeichensätzen auch wiederum in einem Byte dargestellt werden. Der Informationsgehalt wurde also um einen Faktor 256 reduziert. Gleichzeitig fand aber ein *Wissensgewinn* statt, denn nachdem die gesamte Verarbeitung durchgeführt wurde und das Bildverarbeitungssystem uns sein Ergebnis mitgeteilt hat, wissen wir, dass auf dem Bild ein bestimmter Buchstabe zu sehen war. Dies ist ein sehr typisches Phänomen in der Bildverarbeitung: Die große Menge an unstrukturierter Information, die in einem Bild enthalten ist, wird schrittweise strukturiert und verdichtet, bis eine auf sehr kleinem Raum darstellbare symbolische Information entsteht.

1.5 Vorgehensmodell

Es gibt verschiedene grundlegende Ansätze für den Entwurf der Abfolge von Algorithmen für die Lösung einer Bildverarbeitungsaufgabe. Welche Vorgehensweise man verwendet, hängt einmal von der Aufgabenstellung und dem Bildmaterial ab: Der Einsatz digitaler Bildverarbeitung für die Auswertung von Satellitenbildern und für die industrielle Qualitätskontrolle unterliegt völlig verschiedenen Voraussetzungen. Die Beschaffenheit der Bilder, ihr typischer Inhalt und die Zielsetzung sind verschieden. Des Weiteren wird die Vorgehensweise auch von den verwendeten Werkzeugen bestimmt. Bis zu einem gewissen Grad wird jedes Bildverarbeitungssystem dem Anwender eine bestimmte Arbeitsweise aufprägen – außer er unterzieht sich der Mühe, jede einzelne Funktion und den gesamten Anwendungsrahmen selbst zu programmieren. Und schließlich ist es auch eine Frage des persönlichen Geschmacks, was man daran erkennen kann, dass verschiedene Autoren unterschiedliche Vorgehensweisen favorisieren.

Wir haben hier weder den Platz auf alle diese unterschiedlichen Möglichkeiten einzugehen, noch entspräche dies der Zielrichtung dieses Buches. Natürlich bevorzugen auch wir einen bestimmten Ansatz, den wir, nach unserer Erfahrung, für eine große Zahl industrieller Anwendungen für angemessen halten und der sich entsprechend auch in der von uns eingesetzten Software widerspiegelt. Da die im Folgenden vorgestellten Beispiele durchgängig mit dieser Software und daher auf der Basis dieses Denkansatzes entwickelt wurden, erscheint es angebracht, diesen Ansatz zu Beginn als Orientierungshilfe einmal grob zu skizzieren.

Der Zweck einer industriellen Sichtprüfung besteht im Allgemeinen darin, die Übereinstimmung eines Prüfteils mit bestimmten Vorgaben zu überprüfen, z. B. hinsichtlich vorgeschriebener Abmessungen, Seriennummern, vorhandener Komponenten etc. Oft lässt sich die Gesamtaufgabe in mehrere voneinander unabhängige Teile zerlegen, in denen jeweils ein bestimmtes Kriterium geprüft wird. Diese Einzelprüfungen genannten unabhängigen Teile verlaufen dann meist nach folgendem Schema:

1. Bildaufnahme
2. Bildvorverarbeitung
3. Setzen von einem oder mehreren (manuellen) Arbeitsbereichen
4. Erzeugung von Objekten
5. Berechnung von Objekteigenschaften
6. Entscheidung über die Korrektheit der vorhandenen Objekte

Voraussetzung für die Analyse einer Bildszene ist natürlich die Aufnahme eines Bildes, ggf. auch mehrerer Bilder, beispielsweise wenn bewegte Vorgänge zu untersuchen sind. In vielen Fällen sind diese Bilder nicht unmittelbar für die weitere Untersuchung geeignet und werden daher einer Vorverarbeitung unterzogen, die bestimmte Charakteristika ändert, z. B. den Kontrast verstärkt, Störungen unterdrückt, bestimmte Strukturen hervorhebt etc. Sie werden in den verschiedenen Beispielen der folgenden Kapitel eine Reihe dieser sogenannten Vorverarbeitungsmethoden kennenlernen. Ausführlich werden sie in Kapitel 2 besprochen.

In den meisten Fällen wird zumindest annähernd bekannt sein, welche Bereiche des Bildes näher zu untersuchen sind, also z. B. wo sich ein zu lesender Schriftzug befindet oder in welchem Gebiet sich ein Bauteil befindet, dessen Anwesenheit sicherzustellen ist. Daher werden Arbeitsbereiche gesetzt, die oft als *Area of Interest* (abgekürzt *AOI*) oder *Region of Interest* (abgekürzt *ROI*) bezeichnet werden. Im Einzelfall kann ein solcher Arbeitsbereich natürlich auch einmal das gesamte Bild umfassen. Die Begrenzung der weiteren Bildanalyse auf diese Bereiche dient sowohl zur Beschleunigung der Verarbeitung, da nicht mehr die gesamte Bildinformation betrachtet werden muss, als auch zur Erhöhung der Flexibilität. Es kann ja durchaus vorkommen, dass in den verschiedenen Gebieten des Bildes nach ganz unterschiedlichen Informationen gesucht wird und sie sich in ihren Charakteristika so sehr unterscheiden, dass sie sich nicht mit denselben Verfahren bearbeiten lassen. So könnte man z. B. in einem Gebiet einen hellen, in einem anderen einen dunklen Schriftzug gleichzeitig lesen lassen, da für jeden Bereich individuelle Verarbeitungsparameter gesetzt werden können.

Bei der industriellen Bildverarbeitung muss man in der Regel Eigenschaften bestimmter Objekte ermitteln, die in der Bildszene enthalten sind. Im nächsten Schritt sind also diese Objekte zu isolieren, ein Vorgang, den man als *Segmentierung* bezeichnet. Wegen der zentralen Bedeutung dieses Verarbeitungsschrittes werden die verschiedenen Segmentierungsmethoden in einem eigenen Kapitel (Kapitel 4) besprochen. Nachdem die einzelnen Objekte isoliert wurden, können dann charakteristische Merkmale berechnet werden, wie etwa Fläche, Umfang, Position, Orientierung, Abstände voneinander, Ähnlichkeit zu vorgegebenen Mustern (z. B. zur Schrifterkennung). Abschließend wird dann die Übereinstimmung dieser Merkmale mit den Vorgaben der Prüfaufgabe bewertet.

Es sei noch vermerkt, dass es gelegentlich Anwendungen gibt, in denen keine Segmentierung erforderlich ist. Ein Beispiel ist die Oberflächenprüfung, bei der man oft einfach bestimmte Merkmale (wie z. B. die Helligkeitsverteilung) eines Bildbereiches bewerten kann. In einem solchen Fall können die charakteristischen Merkmale also direkt aus den manuell gesetzten Arbeitsbereichen berechnet wer-

den. Aus diesem Grund macht der von den Autoren verfolgte Ansatz keinen prinzipiellen Unterschied zwischen manuell gesetzten Arbeitsbereichen und segmentierten Objekten. Natürlich verhalten sich diese beiden Typen von Strukturen in gewisser Hinsicht verschieden, und nicht jede Operation ist für beide Typen sinnvoll (so sind z. B. Merkmale wie Position und Größe für einen manuellen Arbeitsbereich von vornherein bekannt und unveränderlich), im Allgemeinen kann man sie aber als äquivalent betrachten. Wir möchten diesen Punkt ganz besonders betonen, da dies eine ungewöhnliche Art des Herangehens ist, die aber ein sehr flexibles Arbeiten ermöglicht, wie Sie später in den Beispielen noch sehen werden. In unserem Modell ist es also einerseits möglich, alle für (reale) Bildobjekte verfügbaren Merkmale auch für die manuell an bestimmten Stellen des Bildes definierten Arbeitsbereiche zu berechnen. Andererseits kann man jedes aus der Bildszene segmentierte Objekt wie einen Arbeitsbereich benutzen, d. h. nachfolgende Operationen auf diesen Bereich begrenzen. So lassen sich in der Technik häufig vorkommende hierarchische Strukturen, wie etwa Schrauben in einem Gehäuse, Bauteile innerhalb der Begrenzungen einer Platine, leicht in der Struktur des Bildverarbeitungssystems abbilden.

1.6 Einführungsbeispiel

Das in diesem Abschnitt vorgestellte Beispiel soll Sie kurz mit den wesentlichen Begriffen der industriellen Bildverarbeitung vertraut machen. Die Realisierung von Sichtprüf anlagen erfordert es, viele verschiedene Aspekte parallel zu berücksichtigen, wie es der parallelen Natur der visuellen Informationsverarbeitung entspricht. Ein Buch ist mit seiner sequentiellen Struktur nicht unbedingt das angemessene Medium hierfür, und daher wird es bei den anspruchsvolleren Beispielen der folgenden Kapitel gelegentlich erforderlich sein, Methoden zu benutzen, die erst später ausführlich behandelt werden können. Aus diesen Gründen wollen wir Ihnen hier einen schnellen Überblick über verschiedene Gebiete und Verfahren geben, damit wir in den folgenden Kapiteln auf diese Begriffe zurückgreifen können.

Das Einführungsbeispiel in diesem Abschnitt demonstriert drei der häufigsten und grundlegendsten Anwendungsbereiche industrieller Bildverarbeitung:

- Schrifterkennung als Spezialfall für die Verwendung von Mustererkennungsverfahren in der Identifikationstechnologie
- Vollständigkeitsprüfung
- Vermessung

Dabei werden wir hier nicht auf irgendwelche Details der angewandten Verfahren eingehen, vielmehr soll es ausschließlich um die Einführung von Begriffen und die Herangehensweise an eine Sichtprüfungsaufgabe gehen.

Prüfaufgabe: An einer Zündkerze sind folgende Punkte zu überprüfen:

1. Dreistellige Typnummer auf dem Keramikkörper
2. Gewindetiefe
3. Vorhandensein zweier Unterlegscheiben

Abbildung 1.7 zeigt die Zündkerze unter zwei verschiedenen Beleuchtungen, einmal im für die Schrifterkennung erforderlichen Auflicht, dann im für die Vermessung vorteilhaften Durchlicht.

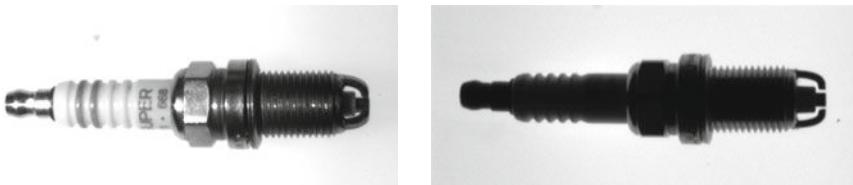


Abbildung 1.7: Zündkerze in Auf- und Durchlicht

Programmhinweis: Sie finden dieses Beispiel im Download im Verzeichnis Beispiele\Einfuehrung\.

1.6.1 Schrifterkennung

Strukturell betrachtet ist die Erkennung der Typnummer die Teilaufgabe mit der klarsten Lösung: Man finde zunächst die Zeichen im Bild, präsentiere diese dann einem Klassifikator, der in der Lage ist, dem optischen Bild eines Zeichens den richtigen Namen zuzuordnen, und vergleiche dann diese Namen mit der Vorgabe für die Typnummer. Entsprechend unserem Vorgehensmodell aus Abschnitt 1.5 vollzieht sich die Lösung der Aufgabe dann in folgenden Schritten:

1. *Bildaufnahme:* Hier wird das Bild von der Kamera digitalisiert und in den Hauptspeicher übertragen. Das Ergebnis ist in unserem Beispiel die Abbildung 1.7.
2. *Arbeitsbereiche definieren:* Für die Erkennung der Typnummer kann der Bereich, in dem Zeichen zu suchen sind, auf einen kleinen Ausschnitt des Keramikkörpers begrenzt werden. Neben der für die Suche erforderlichen Zeit, die natürlich um so geringer ist, je kleiner der abzusuchende Bereich wird, spielt hier auch die Sicherheit eine Rolle: Auch andere Bereiche des Bildes könnten Zeichen oder jedenfalls ähnliche Strukturen enthalten, die die Zuordnung erschweren würden. Abbildung 1.8 zeigt den hier verwendeten Suchbereich.
3. *Erzeugung von Objekten:* Innerhalb des definierten Suchbereiches wird nun jedes einzelne Zeichen als ein separates Bildobjekt erzeugt. Dieser Vorgang wird als *Segmentierung* bezeichnet. Auf welche Weise dies erfolgt, soll uns hier vorläufig nicht interessieren, das Querschnittskapitel 4 befasst sich ausführlich mit Segmentierungsmethoden.

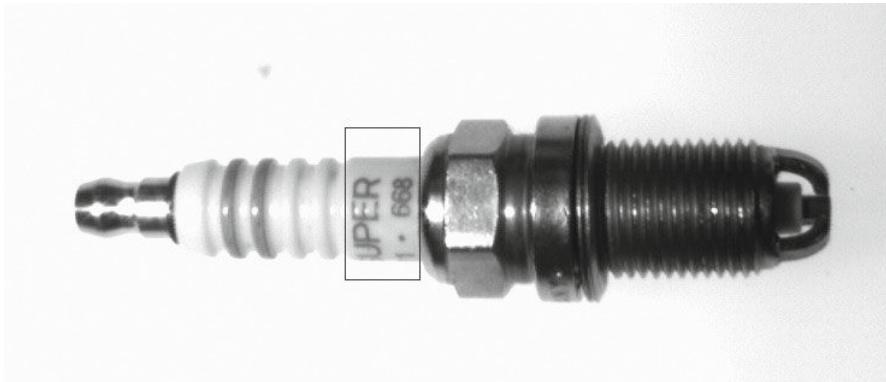


Abbildung 1.8: Arbeitsbereich für die Suche nach der Typnummer

4. *Berechnung von Objekteigenschaften:* Für einen Menschen ist das Bild einer Ziffer unmittelbar als solche erkennbar, die Maschine muss hingegen aus dieser mehr oder weniger zufälligen Anhäufung heller und dunkler Punkte ableiten, um welche Ziffer es sich handelt, d. h. sie muss die einzelnen segmentierten Objekte klassifizieren. Kapitel 6 wird sich ausführlich mit *Klassifikatoren* befassen. Hier möge es genügen, einen Klassifikator als eine Black-Box zu definieren, die gewisse Eingangssignale erhält und als Ausgangsignal eine Klassennummer liefert. In unserem Beispiel heißt das konkret, dass der Klassifikator mit bestimmten Eigenschaften der einzelnen Ziffernobjekte „gefüttert“ wird und aus diesen Eigenschaften berechnet, um welche Klasse, d. h. welche Ziffer es sich handelt.

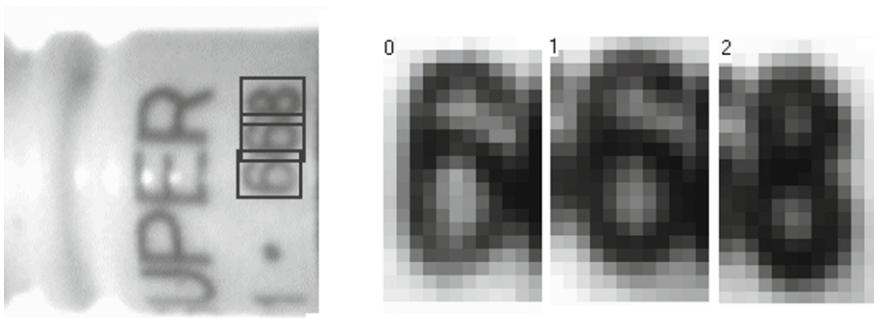


Abbildung 1.9: Segmentierte und größennormierte Ziffern

Bei der Erkennung gedruckter Zeichen wird man sehr häufig das Bild des Zeichens direkt als Eingangssignal verwenden, oft aber kann man dem Klassifikator durch bestimmte Zusatzmerkmale die Aufgabe erheblich erleichtern. Aber

auch wenn nur das Bild selbst als Eingangssignal dienen soll, wird man um eine gewisse Verarbeitung desselben nicht herumkommen; da das Bild des Zeichens aus sehr vielen Einzelwerten besteht – einem Helligkeitswert für jeden Bildpunkt – muss man dafür sorgen, dass eventuelle Größenunterschiede der Zeichen ausgeglichen werden, da Klassifikatoren gewöhnlich eine feste Anzahl von Eingängen aufweisen. In diesem Schritt werden wir daher eine solche Normierung der Zeichen vornehmen. Abbildung 1.9 zeigt die nach dem Segmentierungsschritt durch Rahmen gekennzeichneten Objekte und die zugehörigen größennormierten Bilder (letztere wurden der leichteren Erkennbarkeit halber in die normale Schriftlage gedreht; für den Klassifikator spielt die Lage keine Rolle).

5. *Entscheidung:* Im letzten Schritt werden dann die segmentierten Objekte anhand des soeben berechneten normierten Bildes klassifiziert und die Übereinstimmung der erkannten Klassen mit den Vorgabewerten festgestellt. Das Ergebnis dieser Klassifikation sehen Sie in Abbildung 1.10.

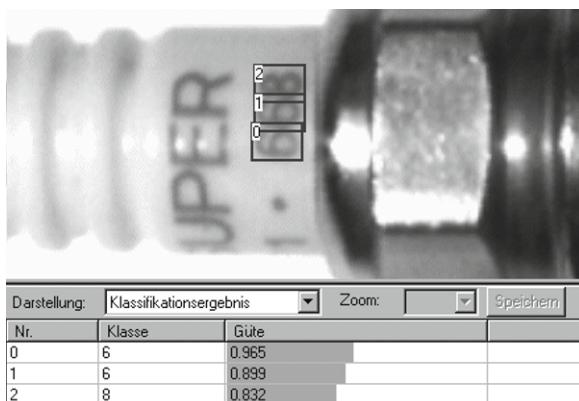


Abbildung 1.10: Ergebnis der Klassenauswertung

Sie werden sicher bemerkt haben, dass in dem oben dargestellten Ablauf einer der Schritte aus Abschnitt 1.5 fehlt, nämlich die Bildvorverarbeitung. Wenn möglich, wird man Vorverarbeitung nur wenig einsetzen. Denn da Vorverarbeitungsalgorithmen normalerweise einem relativ einfachen Ansatz folgen und auf das gesamte Bild wirken, sind sie oft verhältnismäßig rechenaufwändig. Vor allem aber sollte Vorverarbeitung keinesfalls als Ersatz für sorgfältig optimierte Beleuchtung und Bildaufnahme herhalten müssen, denn was im Bereich der Sensorik verloren geht, lässt sich hinterher praktisch nicht wiedergewinnen. Eine Tatsache, die von vielen, die noch keine oder nur wenig Erfahrung mit Bildverarbeitung haben, oft unterschätzt wird.

Die Abgrenzung zwischen Vorverarbeitung und Bildanalyse, die in unserem Modell mit der Erzeugung von Objekten beginnt, ist in vielen Fällen durchaus fließend. Zur Segmentierung von Objekten anhand ihrer Helligkeit erfolgt zum Beispiel in den meisten Büchern zunächst die Erzeugung eines Binärbildes, eines

Bildes mit nur zwei Helligkeitsstufen, Schwarz und Weiß, deren eine idealerweise den gesamten Hintergrund repräsentiert, während die andere alle Objekte kennzeichnet. Definitionsgemäß handelt es sich hier um eine Vorverarbeitung, da ein neues, ähnliches Bild erzeugt wird. Man kann aber die Objekte auch unmittelbar aus dem Originalbild erzeugen, ohne zwischendurch ein echtes Binärbild zu erzeugen, so dass eine Vorverarbeitungsstufe entfällt.

Der oben dargestellte Ablauf lässt natürlich einige Feinheiten vermissen, die bei der ausführlichen Besprechung der verschiedenen Anwendungsbereiche noch im Einzelnen behandelt werden. Wenden wir uns also nun der zweiten Teilaufgabe zu.

1.6.2 Gewindetiefe

Bei der Prüfung der Gewindetiefe müssen zwei verschiedene Fehlerzustände unterschieden werden. Einerseits könnte das Gewinde zu tief geschnitten sein, andererseits nicht ausreichend tief oder sogar gar nicht vorhanden sein. Diese beiden Fehlerzustände gilt es nun so zu beschreiben, dass sie einer optischen Messung zugänglich werden, denn man kann einem Computer ja nicht einfach anweisen „Schau' nach, ob das Gewinde die richtige Tiefe hat!“

Beginnen wir mit dem Fall eines zu tief geschnittenen Gewindes. Wenn man in diesem Fall das Gewinde in der Mitte aufschneidet und zwei Linien parallel zur Achse der Zündkerze so auf die Querschnittsfläche zeichnet, dass sie gerade die Spitze des tiefsten Gewindeganges auf jeder Seite berühren, dann würde ein zu geringer Abstand dieser beiden Linien eine zu große Gewindetiefe anzeigen.

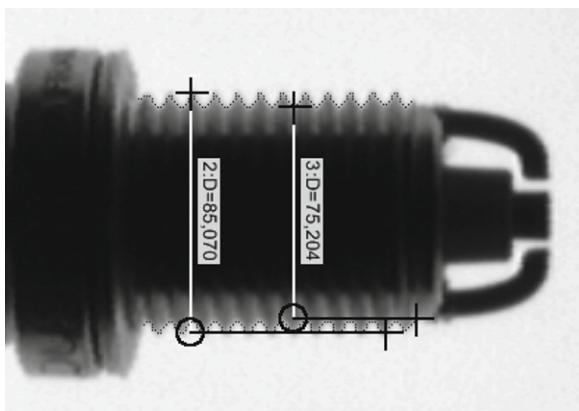


Abbildung 1.11: Prinzip der Gewindetiefenmessung

Im zweiten Fehlerfall, dem gar nicht oder nicht ausreichend tief geschnittenen Gewinde, wird hingegen der Unterschied zwischen dem größten Außendurchmesser und dem Innendurchmesser des Gewindes zu gering sein. Ganz entsprechend kann man auch hier einen Querschnitt durch das Gewinde erzeugen und zwei Li-

nien parallel zur Achse legen, die aber diesmal nicht die innersten, sondern die äußersten Spitzen des Gewindes berühren. Der Innendurchmesser ist schon aus dem vorigen Fehlerfall bekannt, so dass sich der Unterschied nunmehr leicht bestimmen lässt. Da sich sowohl Außendurchmesser als auch Innendurchmesser innerhalb enger Toleranzen bewegen müssen, ist es nicht einmal nötig, die Differenz tatsächlich zu berechnen, man kann einfach jedes Maß einzeln prüfen. Abbildung 1.11 zeigt das Prinzip.

Der Ablauf der Prüfung ist prinzipiell wie gehabt, jedoch unterscheidet sich die Objektsegmentierung recht erheblich von der bei der Teilaufgabe Schrifterkennung verwendeten. Natürlich wäre es grundsätzlich möglich, das gesamte Gewinde als Objekt zu segmentieren und seine minimale und maximale Breite zu messen, es geht jedoch erheblich schneller, lediglich von beiden Seiten her die Außenkontur anzutasten.

1. *Bildaufnahme:* Wieder wird das Bild aufgenommen und in den Hauptspeicher übertragen. Bei hinreichender Kameraauflösung kann man alle Teilaufgaben auch im gleichen Bild durchführen, so dass sich die mehrfache Bildaufnahme erübrigt.
2. *Arbeitsbereiche definieren:* Entsprechend der Aufgabenstellung, minimalen und maximalen Abstand der beiden Gewindekonturen zu vermessen, werden zwei Linien als Suchbereiche verwendet, die jeweils von oben und von unten die Gewindekontur kreuzen. Die Kontur ist dann sehr leicht zu finden, indem entlang dieser Linien nach einem Wechsel von hell nach dunkel gesucht wird. Sobald dieser Wechsel stattfindet, ist ein Punkt der Kontur gefunden.
3. *Objekte erzeugen:* Nachdem der jeweils erste Konturpunkt entlang der Suchlinien gefunden wurde, können die beiden Randlinien leicht durch einen Algorithmus erzeugt werden, der die Kontur so verfolgt, dass stets auf einer Seite helle, auf der anderen Seite dunkle Bildpunkte liegen (wie eine solche Kontur im Einzelnen beschrieben und verfolgt wird, ist in Kapitel 4 erläutert). Eine Begrenzung zu beiden Seiten der Suchlinien verhindert dabei, dass der Verfolgungsalgorithmus um die ganze Zündkerze herumläuft. Bild 1.12 zeigt die Suchlinien mit ihren Begrenzungen und die erzeugten Konturlinien.
4. *Objekteigenschaften berechnen:* In diesem Schritt werden hier die minimalen und maximalen Abstände der gefundenen Konturlinien senkrecht zur Achse der Zündkerze berechnet. Dazu werden (rein rechnerisch, versteht sich) Parallelen zu dieser Achse in das Bild gelegt und so lange verschoben, bis sie jeweils gerade noch die äußersten, bzw. innersten, Punkte der Konturlinien berühren. Da die Zündkerze genau waagerecht liegt, kann einfach die Differenz der y-Koordinaten zur Messung benutzt werden³.
5. *Entscheidung:* Dieser Punkt ist hier sehr einfach; die berechneten Abstände werden mit den Vorgabewerten verglichen, d. h. es wird geprüft, ob die Abstände innerhalb eingestellter Toleranzen um die Sollwerte liegen.

³ Die Kapitel 3 und 7 erläutern die Behandlung von Lageabweichungen und Messungen mit höherer Genauigkeit als ganze Bildpunkte.

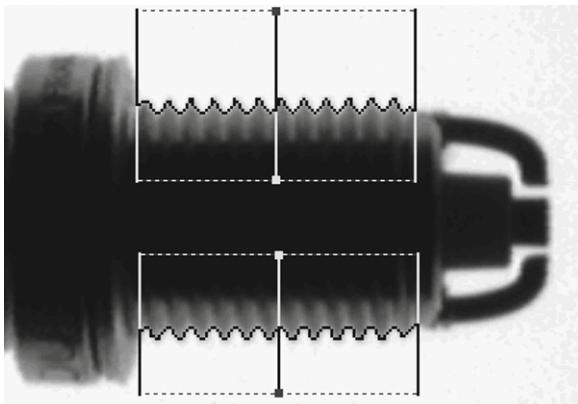


Abbildung 1.12: Suchlinien und segmentierte Gewindekonturen

1.6.3 Anwesenheitskontrolle

Sichtprüfaufgaben des Typs *Anwesenheitskontrolle* sind häufig ebenso einfach zu beschreiben wie schwierig zu lösen. In diesem Fall lautet die Aufgabe, festzustellen, ob die beiden Unterlegscheiben vorhanden sind, die Sie in Abbildung 1.7 auf dem Abschluss unten am Gewinde erkennen können. Die Schwierigkeit liegt darin, dass es praktisch unmöglich ist, diese Unterlegscheiben als einzelne Objekte von der Gewindebasis zu trennen. Für einen Menschen mit seiner Fähigkeit, unterbrochene Kanten auch unter ungünstigen Lichtverhältnissen mühelos zu ergänzen, ist es ein Leichtes, die Teilung zwischen den Unterlegscheiben und der Gewindebasis zu erkennen. Ein Bildverarbeitungssystem kann dies praktisch nur unter Einbringung erheblichen Vorwissens über den Aufbau der Bildszene leisten. Es gibt aber erfreulicherweise einen erheblich einfacheren Lösungsweg. Man macht sich dazu die scheinbar störende Tatsache zunutze, dass Gewindebasis und Unterlegscheiben als ein Objekt erscheinen. Wenn die Unterlegscheiben fehlen, ändert sich die Dicke dieses Objekts, d. h. seine horizontale Ausdehnung. Anhand dieses Maßes ist also die Anwesenheit der Unterlegscheiben leicht feststellbar.

Dieses Vorgehen findet man sehr häufig bei Aufgaben vom Typ der Anwesenheits- oder Vollständigkeitskontrolle: Statt alle erforderlichen Objekte einzeln zu finden und zu zählen, überprüft man die Einhaltung bestimmter Eigenschaften, die sich im Fehlerfall signifikant ändern. Da die Segmentierung von Objekten oftmals den schwierigsten Schritt in einer Bildverarbeitungsanwendung darstellt, erreicht man so eine deutlich höhere Zuverlässigkeit, als wenn man unter allen Umständen die menschliche Sichtweise nachzuvollziehen versuchte.

Die Lösung der Prüfaufgabe stellt sich dann wie folgt dar:

- Bildaufnahme:* Wie in den vorherigen Teilaufgaben wird das Bild aufgenommen und in den Hauptspeicher übertragen.
- Arbeitsbereiche definieren:* Entsprechend der Aufgabenstellung, die Dicke der Gewindebasis (einschließlich Unterlegscheiben) zu messen, werden zwei Linien als Suchbereiche verwendet, die jeweils von rechts nach links die Gewindebasis schneiden. Man durchbohrt also gewissermaßen das Objekt auf jeder Seite mit einer Messlinie, wie es in Abbildung 1.13 dargestellt ist.



Abbildung 1.13: Gewindebasis mit Suchlinien

An dieser Stelle muss unbedingt darauf hingewiesen werden, dass diese Vorgehensweise nur funktioniert, wenn die Positionierung der Zündkerze vor der Kamera immer absolut gleich ist. Davon kann man in realen Anwendungen nicht ausgehen, so dass man praktisch immer Lagekorrekturen vornehmen muss. Aus diesem Grund werden wir uns gleich in Kapitel 3 mit der Lageerkennung befassen.

- Objekte erzeugen:* Die Gewindebasis lässt sich ebenso vom Hintergrund trennen wie das Gewinde selbst im vorigen Abschnitt, jedoch sind wir hier nicht an der Außenkontur interessiert, sondern daran, auf welcher Länge die Suchlinien „im Innern“ der Gewindebasis verlaufen, also wie lang der Teil der Suchlinien ist, der sich über dunklen Bildbereichen befindet. Nachdem also der erste dunkle Punkt auf einer der beiden Linien gefunden ist, verfolgt das Suchverfahren nicht die Außenkontur, sondern tastet sich weiter an der Linie entlang, bis es wieder auf helle Punkte stößt. Der vollständig dunkle Bereich wird als Objekt erzeugt. Abbildung 1.14 zeigt die beiden liniengleichartige Objekte, die dadurch entstehen. Sie verlaufen natürlich genauso wie die definierten Messlinien, aber nur innerhalb des dunklen Bereichs, der durch die Gewindebasis und die Unterlegscheiben umgrenzt wird.
- Objekteigenschaften berechnen:* In diesem Schritt wird für jedes der erzeugten Linienobjekte die Länge berechnet, die ja das entscheidende Kriterium hier darstellt. Danach werden alle Objekte entfernt, deren Länge nicht den Vorgaben entspricht. Die Objekte dürfen weder zu kurz sein (wodurch das Fehlen der Unterlegscheiben angezeigt würde), noch zu lang (was auf ein Segmentierungsproblem schließen ließe).



Abbildung 1.14: Ergebnis der Anwesenheitskontrolle mit Schnitten durch Gewindegroßbasis und Unterlegscheiben

5. *Entscheidung:* In diesem Schritt wird hier die Anzahl der Objekte festgestellt, die die korrekte Länge hatten. Nur wenn auf jeder Seite genau ein solches Objekt vorhanden ist, wird davon ausgegangen, dass die Objekte korrekt segmentiert wurden und die Unterlegscheiben vorhanden sind. In diesem Fall ist das Prüfteil dann schließlich „in Ordnung“.

1.7 Weiteres Vorgehen

Im weiteren Verlauf dieses Buches werden wir jeweils Anwendungen eines bestimmten Typs vorstellen. Mit Ausnahme einiger illustrativer Beispiele sind alle diese Anwendungen reale Prüfaufgaben aus der Industrie, die auch im Wesentlichen in der beschriebenen Form gelöst wurden. Die Einschränkung „im Wesentlichen“ machen wir deshalb, weil nicht alle Details einer Fertigungslösung für die Aufbereitung in einem Einführungsbuch geeignet erscheinen.

In einer Reihe von Fällen sind die beteiligten Firmen genannt. Leider war das nicht immer möglich, meistens aus Gründen der Geheimhaltung, teilweise aber auch aus räumlichen Gründen. Eine solche Veröffentlichung setzt eine sehr enge Zusammenarbeit voraus, was über große Entfernung nur schwer herzustellen ist.

Die verschiedenen Anwendungen verwenden meist ein breites Spektrum von Verfahren, was auch der Grund dafür ist, dass wir uns entschlossen haben, von der hergebrachten Systematik abzuweichen, isolierte, nach bestimmten mathematischen Gesichtspunkten geordnete Algorithmen zu beschreiben. Bildverarbeitungsaufgaben lassen sich nur durch das Zusammenwirken verschiedener Methoden lösen und unser Anliegen war es, genau dieses Zusammenwirken darzustellen. Dennoch hat jede Anwendung ihren besonderen Schwerpunkt, der dann in einem nachfolgenden Querschnittskapitel vertieft wird.

Wir haben uns bemüht, die Kapitel so eigenständig wie möglich zu halten, so dass es auch einfach möglich ist, mal ein Kapitel über Vermessung zu lesen, zu einem anderen Zeitpunkt sich über Klassifikationsverfahren zu informieren. Abhängigkeiten ließen sich dennoch nicht ganz vermeiden, ohne in endlose Wiederholungen zu verfallen. Wir haben aber versucht, mehr Antworten als Querverweise zu geben.

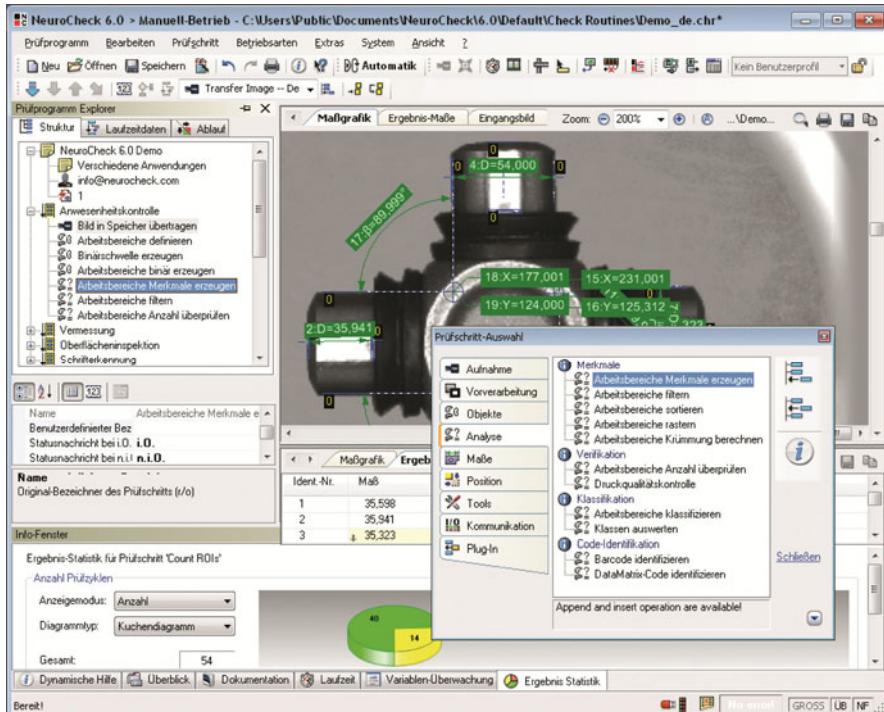


Abbildung 1.15: Benutzeroberfläche für die Konfiguration einer Bildverarbeitungslösung

In diesem Zusammenhang möchten wir auch noch darauf hinweisen, dass ein solches Buch natürlich keinen Anspruch auf Vollständigkeit erheben kann. Im Bereich der digitalen Bildverarbeitung ist in den vergangenen Jahren eine Vielzahl von Verfahren und Algorithmen entwickelt worden, die zum Teil an sehr spezielle Anforderungen angepasst sind. Hierauf einzugehen würde den Rahmen eines Einführungswerkes bei weitem sprengen. Wir haben uns daher auf bewährte und für eine breite Palette industrieller Anwendungen einsetzbare Verfahren konzentriert. Für den an speziellen Verfahren oder tiefer gehenden mathematischen Hintergründen interessierten Leser haben wir im Literaturverzeichnis eine Auswahl weiterführender Werke zusammengestellt. Wir sind jedoch der Meinung, dass eine Anhäufung von Verweisen auf Aufsätze in theoretisch orientierten Forschungsjournalen dem Charakter eines Einführungsbuches nicht angemessen ist und der Leserschaft, die wir hier ansprechen wollen, nicht zu großem Nutzen gereicht.

Daher führen wir im Literaturverzeichnis bewusst hauptsächlich Standardwerke auf, die eine breite Auswahl an Verfahrenn besprechen und dann, wie es der eher mathematischen Ausrichtung dieser Werke angemessen ist, eine Vielzahl von Literaturstellen aus der Welt der Bildverarbeitungsforschung nennen.

An verschiedenen Stellen werden Sie Absätze finden, die als **Programmhinweise** gekennzeichnet sind. Diese Vermerke beziehen sich auf die Realisierung der dort vorgestellten Verfahren oder Anwendungen mit der als Download zur Verfügung stehenden Bildverarbeitungssoftware **NeuroCheck**. Mit dieser Software wurden die in diesem Buch vorgestellten Anwendungen realisiert, d. h. die Prüflösung wurde in der Software interaktiv konfiguriert und anschließend von der Software vollautomatisch in der Fertigung ausgeführt. Die interaktive Konfiguration umfasst dabei nicht nur die Erstellung der Bildverarbeitungslösung, wie sie in den folgenden Kapiteln beschrieben wird, sondern auch das Einrichten der Kommunikation mit der Anlagensteuerung und die Gestaltung der Visualisierung für das Bedienungspersonal der Fertigungsanlage. Abbildung 1.15 gibt einen Eindruck von der Benutzeroberfläche für die interaktive Konfiguration. Ein Beispiel für eine Visualisierung für das Bedienpersonal zeigt Abbildung 1.16. Mehr über den Download erfahren Sie im Anhang B.



Abbildung 1.16: Frei konfigurierbare Visualisierung für das Bedienpersonal

2. Querschnitt: Bildvorverarbeitung

Vorverarbeitungsalgorithmen bilden häufig den ersten Arbeitsschritt nach der Bildaufnahme und werden auch in vielen Beispielen der nachfolgenden Kapitel benötigt. Daher gibt dieses erste Querschnittskapitel eine Einführung in die Bildvorverarbeitung. (Gonzalez und Woods 2008) geben eine umfassende Übersicht. Um eine möglichst klare Vorstellung von der Wirkungsweise der einzelnen Operationen zu geben, verwenden wir in diesem Kapitel sehr einfache, synthetische Beispielbilder. In den Anwendungsbeispielen der übrigen Kapitel sind viele der Verfahren dann im industriellen Einsatz zu sehen.

Im Allgemeinen versteht man unter Bildvorverarbeitung einen Verarbeitungsschritt, der ein Bild in ein neues Bild überführt, das dem Eingangsbild grundsätzlich ähnlich ist, sich aber in bestimmter Hinsicht, z. B. durch einen verbesserten Kontrast, von diesem unterscheidet. Eine Ausnahme hiervon stellen die Integraltransformationen dar, die in Abschnitt 2.8 kurz besprochen werden. Dort hat das Ergebnisbild optisch normalerweise keinerlei Ähnlichkeit mit dem Eingangsbild, sondern visualisiert bestimmte Eigenschaften dieses Bildes, bei der Fouriertransformation z. B. seine Spektraleigenschaften.

Nach dieser Definition bedeutet Vorverarbeitung die Veränderung der Helligkeit einzelner Bildpunkte oder die Veränderung der Lage der Punkte zueinander. Man kann die in diesem Kapitel besprochenen Vorverarbeitungsfunktionen, bei denen die Helligkeit einzelner Bildpunkte verändert wird, grundsätzlich in zwei Gruppen einteilen, je nachdem woraus sich die Helligkeit eines Bildpunkts im Ergebnisbild bestimmt:

1. *Punktoperationen* verwenden zur Berechnung der Helligkeit des Ergebnispunkts ausschließlich die des korrespondierenden Punkts im Eingangsbild (abhängig natürlich von der Art der Vorverarbeitungsfunktion und ihren Parametern). Ebenfalls in diese Gruppe fallen die Funktionen der *Bildarithmetik*, die zwei oder mehr Bilder miteinander verknüpfen, denn auch hier werden zur Bestimmung der Helligkeit des Ergebnispunkts nur Punkte an genau dieser Bildkoordinate benutzt. Punktoperationen zerfallen wiederum in *homogene* und *inhomogene Punktoperationen*, je nachdem, ob die Transformationsfunktion in jedem Bildpunkt gleich ist oder vom Ort des Bildpunkts abhängt. Punktoperationen werden in folgenden Abschnitten besprochen:

- Grauwerttransformationen (2.1)
- Bildarithmetik (2.2)

2. *Lokale Operationen* beziehen dagegen eine bestimmte Nachbarschaft des jeweiligen Punkts in die Berechnung ein. Die Helligkeit des Ergebnispunkts kann

z. B. auf die mittlere Helligkeit aller Punkte in einer kleinen Umgebung des korrespondierenden Punkts im Eingangsbild gesetzt werden. Lokale Operationen werden in folgenden Abschnitten besprochen:

- Lineare Filter (2.3)
- Medianfilter (2.4)
- Morphologische Filter (2.5)
- Andere nichtlineare Filter (2.6)

Die dritte Gruppe der in diesem Kapitel vorgestellten Verfahren sind geometrische Transformationen, die die Lage der Bildpunkte zueinander verändern (2.7).

2.1 Grauwerttransformation

Nach (Sonka et al. 2008) versteht man unter einer lokalen Grauwerttransformation eine Operation der Form:

$$o_{ij} = f(i_{ij}). \quad (2.1)$$

Darin ist o_{ij} der Grauwert an den Koordinaten $x = i, y = j$ im Ergebnisbild (*output*) und i_{ij} der Grauwert an derselben Stelle im Eingangsbild (*input*). Gleichung (2.1) besagt also, dass der Grauwert jedes Bildpunkts im Eingangsbild mittels derselben Funktion $f()$ in den Grauwert des entsprechenden Bildpunkts im Ausgangsbild umgerechnet wird. Beachten Sie, dass die Transformationsfunktion $f()$ selbst nicht ortsabhängig ist. Andernfalls spricht man von *Helligkeitskorrekturen*, die beispielsweise durch Verrechnung mit einem Referenzbild erfolgen können (siehe Abschnitt 2.2.4).

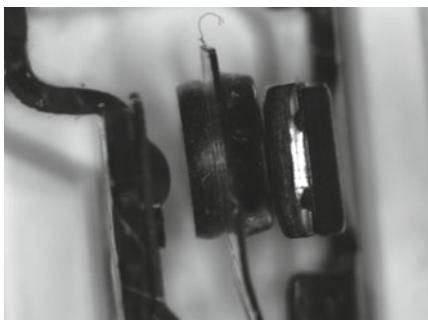
Die Funktion $f()$ ist prinzipiell beliebig. Bei Bildern mit sehr großem Kontrastumfang kann man beispielsweise logarithmische Skalierungen verwenden. $f()$ muss auch durchaus nicht unbedingt analytisch geschlossen darstellbar sein. Ohnehin wird man die Abbildungsfunktion aus Geschwindigkeitsgründen oft in Form einer Look-Up-Tabelle implementieren, womit dann auch völlig beliebige Transformationen möglich werden (siehe Abschnitt 2.1.1).

2.1.1 Look-Up-Tabellen

Wir beginnen diesen Abschnitt über Grauwerttransformationen mit einer Besprechung von Look-Up-Tabellen, da sie für alle nachfolgend beschriebenen Transformationen als Hilfsmittel benutzt werden können. Nicht nur ermöglichen sie Transformationen die sich formal kaum sinnvoll beschreiben ließen, auch analytisch darstellbare Transformationen können unter Benutzung von Look-Up-

Tabellen oft wesentlich schneller durchgeführt werden, als durch direkte Umrechnung jedes einzelnen Bildpunkts.

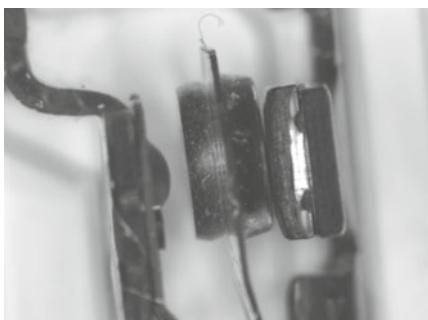
Die Verwendung einer Look-Up-Tabelle zur Grauwerttransformation bedeutet mathematisch, dass die Transformationsfunktion $f()$ in Gleichung (2.1) durch eine Tabelle gegeben ist, die für jeden möglichen Grauwert einen Stützpunkt der Funktion angibt, also jedem Grauwert einen Funktionswert zuordnet. Auch Farbbilder können mit Look-Up-Tabellen transformiert werden, wobei entweder alle Farbkanäle mit derselben Look-Up-Tabelle transformiert werden oder, wenn man die Farbkanäle unterschiedlich behandeln will, jeder Farbkanal mit einer eigenen speziellen Look-Up-Tabelle transformiert wird.



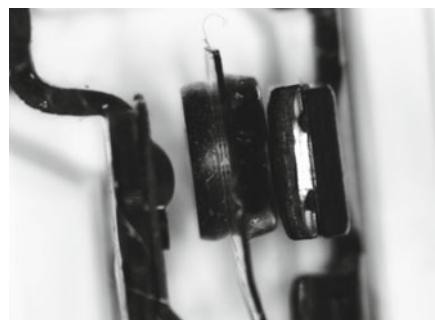
Originalbild



Quadratische Look-Up-Tabelle



Quadratwurzel-Look-Up-Tabelle



Sigmoidale Look-Up-Tabelle

Abbildung 2.1: Wirkung gängiger Look-Up-Tabellen

Je nach Komplexität der Transformation lassen sich durch Look-Up-Tabellen erhebliche Beschleunigungsfaktoren erzielen, bis hin zu Faktoren von mehr als 15 bei logarithmischen oder exponentiellen Skalierungen. Abbildung 2.1 gibt einige Beispiele für die Wirkung gängiger Look-Up-Tabellen. Man sieht, dass die quadratische Look-Up-Tabelle dunkle Bildbereiche weiter verdunkelt, helle noch mehr aufhellt, so dass das Bild mehr Kontrast erhält, aber insgesamt dunkler erscheint. Die Wurzeltabelle dagegen hellt das Bild insgesamt auf, die hellen Bereiche aber weniger stark als die dunklen, so dass der Kontrast nachlässt. Die s-förmige

Gauss-Tabelle homogenisiert sowohl die dunklen als auch die hellen Bildbereiche, verstkt aber den Kontrast in Bereichen mittlerer Bildhelligkeit.

Neben der Beschleunigung, die sich durch die Verwendung von Look-Up-Tabellen erreichen lst, ermglichen sie auch Grauwerttransformationen, die sich nur schwer in einer einfachen Rechenvorschrift erfassen lassen. Abbildung 2.2 zeigt eine speziell definierte Look-Up-Tabelle und ihre Wirkung auf das Originalbild aus Abbildung 2.1. Man erreicht hier praktisch eine Binarisierung des Originalbildes, also eine Umwandlung in nur zwei Graustufen, wobei durch die besondere Form der Look-Up-Tabelle ursprnglich sehr helle Bereiche ebenfalls schwarz werden.

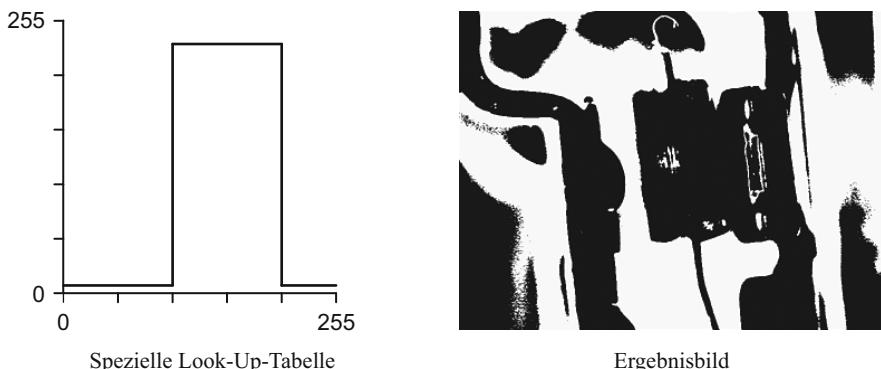


Abbildung 2.2: Wirkung einer speziell angepassten Look-Up-Tabelle

Obwohl sich, wie man an Abbildung 2.2 sieht, mit Look-Up-Tabellen verblffende Effekte erzielen lassen, muss hier eine Warnung ausgesprochen werden. Gerade speziell angepasste Look-Up-Tabellen sind vllig abhngig von stabilen Beleuchtungsbedingungen und einem gleich bleibenden Erscheinungsbild der Prfobjekte. Wenn dies nicht sichergestellt werden kann, dann muss die Grauwerttransformation dynamisch an die aktuellen Bildverhltnisse angepasst werden, was ungleich hheren Aufwand mit sich bringt, sowohl an Rechenzeit als auch an algorithmischer Komplexitt. Look-Up-Tabellen sind also fr Fle geeignet, in denen sich kein absolut befriedigender Beleuchtungszustand herstellen lst, der Erreichbare aber zuverlssig eingehalten werden kann.

Programmhinweis: Sie knnen in **NeuroCheck** ein Bild mit all diesen Look-Up-Tabellen bearbeiten, indem Sie die Funktion *Bild mit Look-Up-Tabelle bearbeiten* in Ihr Prfprogramm aufnehmen.

2.1.2 Lineare Grauwertskalierung

Ein wichtiger Sonderfall der Grauwerttransformation nach Gleichung (2.1) ist die lineare Skalierung:

$$o_{ij} = c_2 \cdot (c_1 + i_{ij}). \quad (2.2)$$

Wiederum ist o_{ij} der Grauwert an den Koordinaten $x = i, y = j$ im Ergebnisbild (*output*) und i_{ij} der Grauwert an derselben Stelle im Eingangsbild (*input*). Für verschiedene Koeffizientenwerte c_1, c_2 ergeben sich folgende Effekte nach (Nischwitz et al. 2007):

- $c_1 > 0$, Bild wird insgesamt heller
- $c_1 < 0$, Bild wird insgesamt dunkler
- $|c_2| > 1$, Bild wird kontrastreicher
- $|c_2| < 1$, Bild wird kontrastärmer

Bei dieser Operation muss man auf korrektes Clipping an den Rändern des Grauwertbereichs achten. Wäre das Ergebnis einer Operation ein negativer Grauwert, so ist null zu verwenden, andererseits sind alle Ergebnisse über 255¹ durch den Wert 255 zu ersetzen.

2.1.3 Kontrastnormierung

Die lineare Skalierung der Grauwerte nach Gleichung (2.2) kann durch entsprechende Wahl der Koeffizienten als einfache, aber in vielen Fällen sehr wirksame Methode zur Bildverbesserung benutzt werden. Dabei verwendet man den minimalen und maximalen Grauwert im Bild, um den Bildkontrast zu normieren, so dass der gesamte Grauwertbereich von 0 bis 255 ausgeschöpft wird.

Wenn g_{\min} der minimale und g_{\max} der maximale Grauwert des Eingangsbildes sind, so wird dieser Effekt durch folgende Abbildung erreicht:

$$o_{ij} = \frac{255}{g_{\max} - g_{\min}} \cdot (i_{ij} - g_{\min}). \quad (2.3)$$

Die Subtraktion von g_{\min} verschiebt den minimalen Grauwert des Bildes auf null, den maximalen auf $g_{\max} - g_{\min}$. Die anschließende Multiplikation sorgt dann dafür,

¹ Wir gehen hier, auch um zusätzliche Abkürzungen zu vermeiden, stets von Bildern mit 256 Grauwerten aus, mit null als niedrigstem und 255 als höchstem Grauwert.

dass der maximale Grauwert des Ergebnisbildes 255 wird. Abbildung 2.3 zeigt den Effekt einer Grauwertnormierung auf ein sehr kontrastarmes Bild.



Abbildung 2.3: Globale Grauwertnormierung eines kontrastschwachen Bildes (normiertes Bild zur Unterscheidung vom Hintergrund umrahmt)

Streng genommen kann man hier nicht mehr von einer Punktoperation im Sinne der obigen Definition sprechen. Tatsächlich hängt jeder Grauwert des Ergebnisbildes von sämtlichen Grauwerten des Eingangsbildes ab, denn die Parameter der Operation, g_{min} und g_{max} werden aus dem gesamten Bild ermittelt. Die Form der Operation ist jedoch genauso wie bei einer willkürlichen Skalierung mit fest vorgegebenen Koeffizienten und daher kann man vom implementatorischen und praktischen Standpunkt aus weiterhin von einer Punktoperation sprechen.

Programmhinweis: In **NeuroCheck** können Sie eine Grauwertspreizung für das gesamte Bild erreichen, indem Sie die Funktion *Bild normieren* benutzen und die Ausdehnung der lokalen Bildbereiche für die Normierung auf die volle Bildgröße setzen.

2.1.4 Histogramm-Ebnung

Außer der in 2.1.3 beschriebenen linearen Grauwertspreizung gibt es noch weitere Methoden zur Kontrastverbesserung, darunter die Histogramm-Ebnung. Histogramme sind Tabellen der Häufigkeit, mit der jeder Grauwert in einem Bild vorkommt. Sie werden ausführlich im Zusammenhang mit der Binärsegmentierung in Abschnitt 4.2.3 beschrieben. Durch entsprechende Grauwerttransformationen kann man dafür sorgen, dass das Histogramm eines Bildes jeder beliebigen vorgegebenen Verteilung entspricht. Im Spezialfall der Histogramm-Ebnung sorgt die Grauwerttransformation dafür, dass jedes Helligkeitsintervall gleichmäßig besetzt ist. Das bedeutet nicht etwa, dass jeder einzelne Grauwert genau gleich häufig vorkommt, das Histogramm also aus lauter gleich hohen Balken besteht, sondern vielmehr, dass in einem Grauwertintervall $[0, g]$ ein der Breite des Intervalls entsprechender Anteil von Pixeln zu finden ist, d. h. $g/256^2$.

Mathematisch funktioniert das folgendermaßen: Jeder Grauwert g des Eingangsbildes wird im Ausgangsbild durch einen Grauwert g' ersetzt, der sich aus

² Allgemeiner müsste man schreiben $g/(g_{max} + 1)$.

der relativen Häufigkeit aller darunter liegenden Grauwerte ergibt. Die Formel dafür lautet:

$$g' = 255 \sum_{i=0}^g \frac{N_i}{N}. \quad (2.4)$$

Darin ist N_i die Anzahl der Pixel mit Grauwert i im Eingangsbild und N die Gesamtzahl der Pixel im Eingangsbild. Der Summenausdruck ist die kumulative Häufigkeit der Grauwerte im Bild. Diese Funktion ist monoton steigend, wodurch sichergestellt ist, dass die Relation zwischen zwei Pixeln unterschiedlichen Grauwerts nicht umgekehrt wird. Ein Pixel, das vor der Transformation heller war als sein Nachbar, wird nach der Transformation nicht dunkler sein.

Eine Anmerkung zur Implementierung: Natürlich führt man nicht die gesamte Berechnung in Gleichung (2.4) für jeden Bildpunkt durch. Man beginnt vielmehr mit der Ermittlung des Histogramms für das gesamte Bild, wie es in Abschnitt 4.2.3 beschrieben ist, und baut dann eine Tabelle auf, die für jeden Grauwert g den nach Gleichung (2.4) berechneten Ergebnisgrauwert g' enthält. Die Transformation der einzelnen Bildpunkte erfolgt dann einfach über diese Look-Up-Tabelle entsprechend Abschnitt 2.1.1.

Welchen Sinn hat diese Operation? Ein Bild möge aus einigen wenigen Regionen mit jeweils ziemlich homogener Helligkeit bestehen. Das Histogramm zeigt dann einige Peaks, deren Größe den Flächen der Regionen entspricht. Eine Histogramm-Ebnung verbreitert diese Peaks bei gleichzeitiger Abflachung, während die Bereiche selten vorkommender Grauwerte zwischen den Peaks komprimiert werden. Die Verbreiterung der Peaks bewirkt, dass die Unterschiede zwischen den Grauwerten in den Regionen größer sind. Die Regionen sind nach der Histogramm-Ebnung folglich weniger homogen, man kann Detailvariationen erkennen, wie Abbildung 2.4 zeigt.

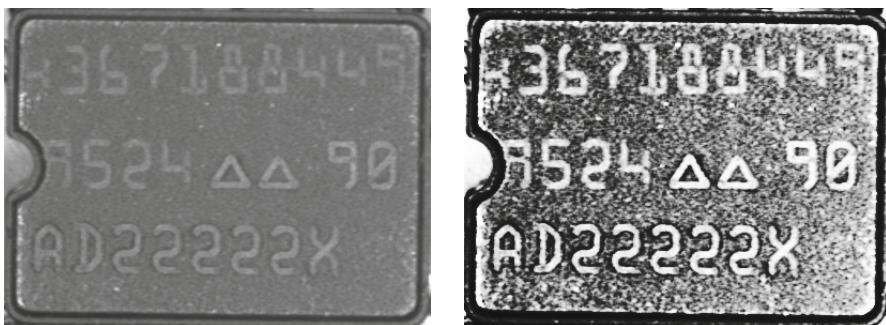


Abbildung 2.4: Verstärkung von Details durch Histogramm-Ebnung

Es sei noch hinzugefügt, dass Verfahren zur Kontrastverbesserung, sei es durch Grauwertspreizung oder Histogramm-Ebnung, für die maschinelle Verarbeitung

eines einzelnen Bildes prinzipiell wenig ändern, da sich die Relation zwischen der Helligkeit zweier Bildpunkte bei dieser Operation nicht umkehren kann: Ein hellerer Punkt bleibt heller, lediglich der Betrag der Differenz ändert sich. Helligkeitsunterschiede zwischen Punkten im Ergebnisbild waren auch im Eingangsbild bereits vorhanden. Dennoch haben diese Verfahren ihren Sinn. Zum einen sind sie sehr nützlich, wenn ein Mensch die Bildinformation begutachten soll, sei es in gedruckter Form oder auf einem Monitor, da Menschen keine so feinen Grauwertabstufungen erkennen können wie ein Bildverarbeitungssystem. Das Unterscheidungsvermögen des Menschen wird also vor allem durch die Histogramm-Ebnung gut unterstützt. Zum anderen kann unter Umständen die Stabilität eines Bildverarbeitungsablaufs dadurch erhöht werden, dass sich die Eingangsbilder in bestimmten Charakteristika – in diesem Fall der Helligkeitsverteilung – gleichen. Ein Beispiel hierfür findet sich etwa in Abschnitt 5.2.5, wo eine lokale Grauwertspreizung dafür sorgt, dass jedes Eingangsbild prinzipiell ähnliche Kontrasteigenschaften aufweist, womit die Segmentierung erheblich erleichtert wird.

2.1.5 Lokale Kontrastnormierung

Zum Abschluss dieses Abschnitts über Grauwerttransformationen, wollen wir noch die lokale Kontrastnormierung vorstellen. Sie ist streng genommen keine Grauwerttransformation im Sinne von Gleichung (2.1) mehr, da die Voraussetzung der Unabhängigkeit der Transformationsfunktion vom Ort des Bildpunkts auf einzelne Bildausschnitte beschränkt wird. Es ist dennoch gerechtfertigt, diesen wichtigen Sonderfall hier zu besprechen, aus folgenden Gründen:

1. Formal enthält die Transformationsfunktion $f()$ weiterhin keine Ortsabhängigkeit; bei einer linearen Skalierung gemäß Gleichung (2.2) wird man nicht die Koeffizienten c_1 und c_2 explizit von den Bildpunktkoordinaten abhängig machen, sondern nur für jeden Bildausschnitt einen eigenen Satz von Koeffizienten verwenden. Die Transformation ist dann also abschnittsweise definiert.
2. Unter praktischen und implementatorischen Gesichtspunkten besteht nur ein geringer Unterschied zwischen der einheitlichen Bearbeitung des gesamten Bildes und der einheitlichen Bearbeitung *vorab definierter* rechteckiger Bildausschnitte, während der Unterschied zu echten lokalen Operationen, bei denen sich die Transformationsfunktion *für jeden einzelnen Bildpunkt* ändert, wesentlich drastischer ausfällt.

Zur lokalen Kontrastnormierung kommen wir, indem die Grauwertgrenzen g_{max} und g_{min} in Gleichung (2.3) nicht mehr für das gesamte Bild bestimmt werden, sondern für jeden Bildausschnitt einzeln. Der Einfachheit halber benutzt man hier normalerweise gleichgroße rechteckige Bildausschnitte, aber natürlich kann man die Operation auch beispielsweise auf den Inhalt eines Arbeitsbereiches oder eines segmentierten Bildobjekts beschränken. Mittels einer solchen lokalen Normierung lassen sich Beleuchtungsgradienten ausgleichen und Schatten beseitigen, wie es z. B. Abbildung 2.5 zeigt. Man sieht, dass eine globale Normierung zwar den

Kontrast allgemein verbessern kann, die Beleuchtungsgradienten jedoch nicht be seitigt. Die lokale Normierung kann hingegen, im Gegensatz zu einfachen Look Up-Tabellen oder Grauwerttransformationen, auch schwankende Beleuchtungs verhältnisse bis zu einem gewissen Grade ausgleichen und verbessert den Kontrast in jedem einzelnen Bildbereich.



Abbildung 2.5: Kontrastarmes Bild mit Hintergrundgradient, nach globaler und lokaler Normierung (von oben nach unten)

Programmhinweis: In NeuroCheck können Sie eine lokale Kontrastnormierung durch entsprechende Einstellung der Feldgrößen im Prüfschritt *Bild normieren* aus der Prüfschritt kategorie Vorverarbeitung erreichen.

Jede beliebige Grauwerttransformation lässt sich in dieser Weise lokal anwenden. Wenn man dann dazu übergeht, das Bild nicht mehr vorab in einzelne Abschnitte einzuteilen, innerhalb derer die Transformationsfunktion jeweils konstant bleibt, sondern die Abschnitte gemeinsam mit dem jeweils bearbeiteten Bildpunkt bewegt, hat man eine echte lokale Operation vor sich, bei der sich die Transformationsfunktion tatsächlich mit jedem einzelnen Bildpunkt ändert. Man bezeichnet dies nach (Jähne 2005) als *inhomogene Punktoperation*. Solche Operationen müs-

sen in der Regel anders behandelt werden als die hier besprochene lokale Normierung. Das kann man sich beispielsweise an der in Abschnitt 2.1.4 beschriebenen Optimierung durch Aufbau einer Look-Up-Tabelle klarmachen. Wenn die Bildausschnitte nicht mehr vorab definiert, sondern mit dem jeweils bearbeiteten Bildpunkt mitbewegt werden, dann muss für jeden Bildpunkt wegen der geänderten Umgebung eine neue Look-Up-Tabelle aufgebaut werden, was sich nur lohnt, wenn die Umgebung deutlich größer ist als die 256 Einträge der Look-Up-Tabelle. Der Charakter des Algorithmus' und seine Implementierung ändern sich dadurch offensichtlich erheblich, was umgekehrt die Einordnung der lokalen Kontrastnormierung unter die Punktoperationen noch einmal rechtfertigt.

2.2 Bildarithmetik

Unter Bildarithmetik versteht man die punktweise Verrechnung zweier (oder mehrerer) Bilder miteinander. In diesem Fall hängt natürlich der Grauwert eines Pixels im Ergebnisbild von den Grauwerten korrespondierender Pixel in *allen* Eingangsbildern ab. Da keine Grauwerte aus einer Umgebung der jeweiligen Pixelkoordinate eine Rolle spielen, ist dies dennoch eine Punktoperation, allerdings eine inhomogene, da die Transformationsfunktion für jeden Bildpunkt in Abhängigkeit von den korrespondierenden Bildpunkten der übrigen Eingangsbilder verschieden sein kann.

Dies wird auch aus der Rechenvorschrift in Gleichung (2.5) klar, die besagt, dass sich der Ergebnisgrauwert an der Stelle (x, y) aus den Grauwerten aller Eingangsbilder an derselben Stelle aufgrund der Rechenvorschrift $f()$ ergibt. Diese Rechenvorschrift hängt aber von den Grauwerten aller anderen Eingangsbilder an derselben Stelle ab, ist also für jeden Punkt möglicherweise verschieden.

$$g_o(x, y) = f(g_0(x, y), g_1(x, y), \dots, g_n(x, y)). \quad (2.5)$$

Im Folgenden werden wir uns nur mit dem häufigen Fall der Verknüpfung zweier Bilder beschäftigen. Prinzipiell lässt sich jede arithmetische und logische Verknüpfung zweier Variabler für die Abbildungsvorschrift $f()$ benutzen. Die in der Bildverarbeitung wichtigsten Operationen sind:

- Addition und Mittelung
- Subtraktion
- Minimum und Maximum
- Division zur Shading-Korrektur

2.2.1 Bildaddition und Bildmittelung

Die Addition von zwei oder mehr Bildern ist eine sehr einfache Operation, jedoch stellt sich für die Darstellung und Weiterverarbeitung das Problem, dass Ergebniswerte den Grauwertbereich von [0, 255] überschreiten können. Um das Ergebnisbild wieder als Grauwertbild darstellen zu können, kann man eine der folgenden Möglichkeiten benutzen:

Clipping: Bei dieser Operation werden Ergebniswerte oberhalb von 255 einfach auf 255 abgeschnitten. Die Methode ist im Multimediacomplex recht verbreitet und daher auch Bestandteil des Befehlssatzes von PC-Prozessoren. Für die Bildverarbeitung hat sie keine große Bedeutung.

Division durch Anzahl der Bilder: Diese Methode entspricht einer Bildmittelung. Sie kann zur Rauschunterdrückung eingesetzt werden. Jede Kamera hat ein gewisses Ausgangsrauschen, das im Gegensatz zur Bildinformation stochastisch ist, d.h. es wird in aufeinander folgenden Bildern derselben Szene verschieden aussehen, während die eigentliche Bildinformation (weitgehend) gleich bleibt. Ein bestimmter Bildpunkt wird also in einem Bild durch das Rauschen etwas aufgehellt sein, in einem anderen Bild etwas verdunkelt. Bei einer Mittelung aufeinander folgender Bilder wird folglich das Rauschen insgesamt reduziert, während die Bildinformation erhalten bleibt. Bei sehr dunklen Bildszenen ist es vorteilhaft, die Bildinformation in der Kamera selbst zu summieren (durch entsprechend lange Belichtungszeit) und auszulesen, bevor ein Pixel Sättigung erreicht hat, bei normal beleuchteten Szenen erreicht man jedoch nach (Russ 2007) die beste Rauschunterdrückung durch Summierung mehrerer normal belichteter Bilder im Rechner. Für eine wirksame Rauschunterdrückung sollte man allerdings wenigstens zehn Bilder verwenden, wodurch die Operation sehr langwierig wird. Daher versucht man sie in der industriellen Bildverarbeitung grundsätzlich durch die Optimierung von Beleuchtung und Sensorik zu vermeiden.

Skalierung auf Minimum und Maximum: Bei dieser Methode werden die Bilder zunächst in einem Ergebnisbild summiert, das alle möglichen Ergebniswerte auch tatsächlich aufnehmen kann. Dann werden der tatsächliche Minimal- und Maximalwert festgestellt. Dieses Intervall wird dann auf [0, 255] skaliert, was einer Kontrastnormierung nach 2.1.3 entspricht. Wegen des Rechen- und Speicheraufwandes vermeidet man in der Regel auch dieses Verfahren.

2.2.2 Bildsubtraktion

Bildsubtraktion ist interessanter und vielseitiger anwendbar als Bildaddition. Da hier auch negative Werte auftreten können, stellen sich jedoch neue Probleme – oder ergeben sich neue Möglichkeiten, je nach Einstellung – für die Ergebnisskalierung. Je nach Effekt, den man erzielen möchte, kann man verschiedene Skalie-

rungsmethoden anwenden. Ihre jeweiligen Wirkungen sind in Abbildung 2.6 gezeigt.

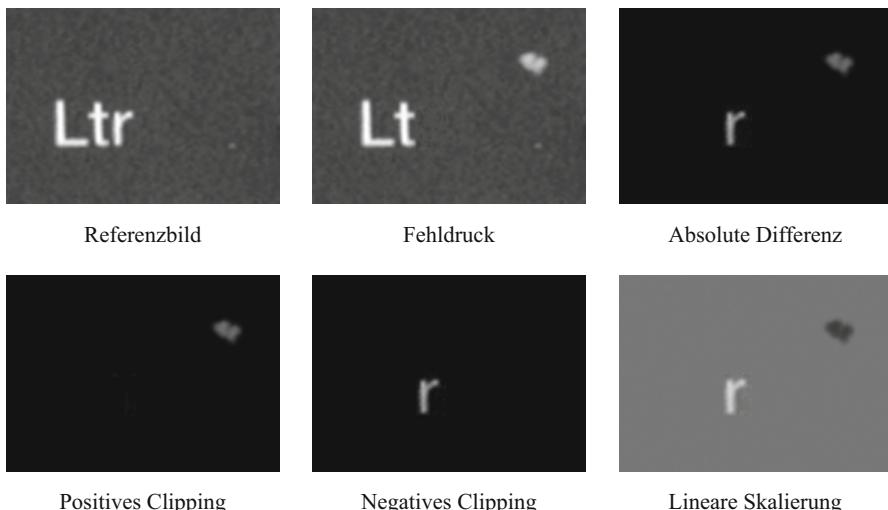


Abbildung 2.6: Ergebnis verschiedener Subtraktionsskalierungen bei zwei verschiedenen Arten von Fehlern im Druck.

Absolutwert: Diese Skalierungsmethode verwendet den Absolutwert des Ergebnisses. Bei nur zwei Eingangsbildern ist dies völlig hinreichend, um die Ergebnisse auf das Intervall [0, 255] zu begrenzen, bei mehr als zwei Bildern ist noch eine Division durch die Anzahl der Bilder minus eins erforderlich (werden von einem Bild zwei andere subtrahiert, so kann das Ergebnis schlimmstenfalls -510 sein; nach Absolutwertbildung und Division durch zwei befindet sich der Wert wieder im Grauwertbereich). Das Ergebnisbild enthält in diesem Fall die Information, an welchen Stellen sich die Eingangsbilder unterscheiden und wie stark. Ein möglicher Einsatzfall wäre die Feststellung der Abwesenheit notwendiger Komponenten, wobei nicht spezifiziert ist, ob diese Komponenten hell oder dunkel sind. Sie müssen sich lediglich vom Hintergrund unterscheiden.

Ein typisches Beispiel für derartige Anwendungen sind Druckqualitätsprüfungen, bei denen sowohl eine Beschädigung des Drucks festgestellt werden soll, d. h. fehlende Teile in den gedruckten Zeichen, als auch Farbflecke. Wir nehmen an, es werde Weiß auf dunkelgrauem Hintergrund gedruckt. Vom aktuellen Kamerabild wird das Referenzbild subtrahiert. Bei einwandfreiem Druck erscheint das Ergebnisbild homogen dunkel (nicht völlig schwarz, so einheitlich sind die Helligkeitswerte realer Objekte niemals). Wenn ein Teil eines Zeichens fehlt, so entsteht zunächst ein großer negativer Wert, da das Kamerabild an dieser Stelle den Hintergrundgrauwert aufweist, das Referenzbild jedoch den der hellen Schrift. Nach der Absolutwertbildung erscheint im Ergebnisbild ein heller Fleck. Genau dasselbe erscheint, wenn im Kamerabild ein zusätzlicher Farbfleck vorhanden ist;

dann ist bereits das Ergebnis der Subtraktion ein hoher Grauwert, und die Absolutwertbildung ändert daran nichts mehr. Beide Arten von Fehlern im Druck erscheinen als helle Flecken im Ergebnisbild der Subtraktion und können danach leicht segmentiert und analysiert werden.

Die Durchführung dieser Subtraktion in einer realen Anwendung zur Druckqualitätskontrolle ist übrigens keineswegs trivial. Kleinste Ungenauigkeiten in der Positionierung führen zu erheblichen Fehlern, da dann sehr deutliche Differenzflächen an den Rändern des Drucks auftreten. Wir kommen in Abschnitt 5.5 auf dieses Beispiel noch einmal zurück.

Positives Clipping: Wenn man sich nur dafür interessiert, wo das erste Bild heller war als das zweite, kann man ein Clipping verwenden, das alle negativen Ergebniswerte auf null setzt. In obigem Beispiel würden dann also fehlende Flächen in den gedruckten Zeichen nicht in Erscheinung treten, da dort das abgezogene Referenzbild heller ist, folglich ein negativer Ergebniswert auftritt, der hier vernachlässigt wird. Diese Abbildungsmethode bietet sich an, wenn nur festgestellt werden soll, wo und in welchem Maße das erste Bild heller ist als das zweite, d. h. ob zusätzliche helle Objekte vorhanden sind oder geforderte dunkle Objekte fehlen. Der Rechenaufwand dieser Methode ist geringer als der für die Absolutwertbestimmung.

Negatives Clipping: Genau umgekehrt kann man auch alle positiven Ergebniswerte auf null setzen und dann den Absolutwert der negativen verwenden. In diesem Fall detektiert man nur die Abwesenheit heller Objekte im ersten Bild oder die Anwesenheit zusätzlicher dunkler.

Lineare Skalierung: Wie bei der Addition ist auch hier eine lineare Skalierung auf den möglichen Grauwertbereich die aufwändigste Methode. Aus denselben Gründen wie dort wird man im Allgemeinen das theoretische Intervall als Ausgangspunkt für die Skalierung wählen und die Abbildung für zwei Eingangsbilder folgendermaßen durchführen:

$$g_0 = \frac{1}{2}(255 + g_1 - g_2). \quad (2.6)$$

Die Differenz der Grauwerte der Eingangsbilder wird um ein volles Grauwertintervall angehoben, d. h. der theoretische Ergebnisbereich von $[-255, 255]$ wird auf $[0, 510]$ verschoben. Dann werden alle Grauwerte halbiert, so dass der Ergebniswert wieder im Bereich $[0, 255]$ liegt. Der Effekt ist, dass alle Punkte, an denen die Grauwerte beider Bilder identisch sind, einen mittleren Grauwert erhalten. Alle Punkte, an denen Bild 1 heller ist als Bild 2, werden heller erscheinen als dieser mittlere Grauwert, alle anderen dunkler. Das Ergebnisbild enthält also nicht nur die Information, wo und wie stark sich die beiden Eingangsbilder unterscheiden, sondern auch noch, in welcher Richtung. Sie ist immer dann anzuwenden, wenn die beiden Fehlertypen unterschiedlich bewertet werden müssen. Bei einer Druckqualitätsprüfung also z. B. dann, wenn eine gewisse Maximalgröße dunkler Krat-

zer im Druck noch toleriert werden kann, nicht aber helle Farbspritzer derselben Größe.

2.2.3 Minimum und Maximum zweier Bilder

Bei der Verknüpfung zweier Bilder mit dem Minimumoperator wird jeder Bildpunkt des Ergebnisbildes auf den Grauwert des Dunkleren der korrespondierenden Punkte in den beiden Eingangsbildern gesetzt. Der Maximumoperator verwendet entsprechend den Helleren der Bildpunkte. Der Minimumoperator kann beispielsweise verwendet werden, um die Gesamtanzahl dunkler Objekte in zwei aufeinander folgenden Bildern zu bestimmen. Er vermeidet dabei, dass Objekte an identischen Positionen doppelt gezählt werden. Der Maximumoperator wird uns in Abschnitt 2.3.4 begegnen, wo er für die Verknüpfung von Kantenbilder benutzt wird. Dabei wird ein Bild mit verschiedenen Kantenfiltern bearbeitet, wodurch Kanten verschiedener Richtung jeweils hell hervortreten, während kantenfreie Bereiche sehr dunkel erscheinen. Die Kanten sind also immer die hellsten Punkte in den einzelnen Ergebnisbildern. Die Verknüpfung dieser Bilder mit dem Maximumoperator sorgt dafür, dass im Ergebnisbild Kanten jeder Richtung hell hervortreten. Dies ist dort in Abbildung 2.22 gezeigt.

2.2.4 Shading-Korrektur

Die so genannte Shading-Korrektur ist ein Spezialfall der Bildarithmetik, in dem die sonst seltene Operation der Bilddivision eingesetzt wird. Es geht hierbei um die Korrektur systematischer Helligkeitsfehler. Diese Fehler können auf verschiedene Arten zustande kommen:

1. Randabschattungen; Bei Objektiven kann man manchmal einen Helligkeitsabfall vom Zentrum des Bildes nach außen hin beobachten, was als Randabschattung oder Vignettierung bezeichnet wird.
2. Ungleichmäßige Beleuchtung, wie z. B. Schatteneffekte, Beleuchtungsgradienten aufgrund von einseitiger Beleuchtung.

Allgemein lassen sich solche Helligkeitsfehler darstellen durch:

$$g(x, y) = f(x, y) b(x, y). \quad (2.7)$$

Hierin sind $g(x, y)$ der Grauwert des aufgenommenen Bildes an der Stelle (x, y) , $b(x, y)$ ist die wahre Helligkeit (*brightness*) an derselben Stelle. $f(x, y)$ ist eine Abbildungsfunktion, die sich im Normalfall nicht analytisch darstellen oder bestimmen lässt. Darin gehen einmal die Beleuchtungsbedingungen ein, dann die Eigenschaften des optischen Systems, außerdem noch die Charakteristika des verwendeten Sensors.

Wie lässt sich nun dieser Fehler korrigieren, wenn die Fehlerfunktion $f(x, y)$ nicht bekannt ist? Da sich hier alles in einem diskreten Pixelraum abspielt, benötigt man eigentlich nur den Zahlenwert von $f(x, y)$ an jeder Stelle. Wenn man nun eine Bildszene hätte, deren wahre Helligkeit an jeder Stelle bekannt wäre, könnte man leicht aus einer Aufnahme dieser Bildszene die Funktionswerte von $f(x, y)$ berechnen und später zur Korrektur benutzen. Genau dies ist der Gedanke der Shading-Korrektur.

Zunächst wird also ein Referenzbild hergestellt, indem eine Fläche homogener Helligkeit unter denselben Bedingungen wie während der späteren Einsatzphase aufgenommen wird (genau genommen müsste die Fläche nicht einmal homogen sein, es erleichtert die Berechnung jedoch erheblich). Es entsteht ein Bild mit den Referenzgrauwerten $g_r(x, y)$, aus denen sich bei bekannter Helligkeit b der aufgenommenen Fläche die Korrekturwerte zu $f(x, y) = g_r(x, y)/b$ ergeben. Die wahren Helligkeiten $b(x, y)$ der Bildpunkte in der zu untersuchenden Bildszene lassen sich dann aus den aufgenommenen Grauwerten $g(x, y)$ mittels Division durch die Referenzgrauwerte und Multiplikation mit der Referenzhelligkeit b berechnen, (Nischwitz et al. 2007):

$$b(x, y) = b \cdot \frac{g(x, y)}{g_r(x, y)}. \quad (2.8)$$

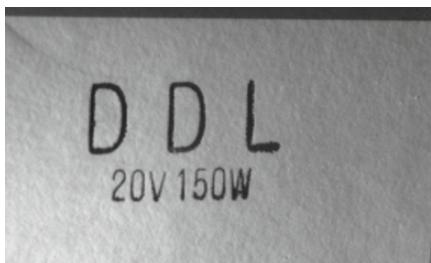
Diese Art der Korrektur ist nach (Russ 2007) richtig für Sensoren mit linearer Beziehung zwischen Helligkeit und gemessenem Grauwert; bei einer logarithmischen Kennlinie, muss man stattdessen eine subtraktive Korrektur durchführen, die im Übrigen auch bei linearen Sensoren häufig ausreichend ist, um ein Bild zu erhalten, das sich vernünftig weiterverarbeiten lässt. Die subtraktive Korrektur hat außerdem den Vorteil, wesentlich weniger rechenaufwändig zu sein. Die Korrekturformel lautet in diesem Fall:

$$b(x, y) = b + (g(x, y) - g_r(x, y)). \quad (2.9)$$

Die wahre Helligkeit in der Bildszene unterscheidet sich also an jedem Punkt von der Referenzhelligkeit b genau um die Differenz zwischen dem beobachteten Grauwert in der Bildszene und dem im Referenzbild. Abbildung 2.7 zeigt das Ergebnis einer Shading-Korrektur bei einem stark einseitigen beleuchteten Etikett. Man sieht deutlich, wie bei einer Binärsegmentierung im Originalbild der letzte Buchstabe aufgrund der schlechteren Beleuchtung mit dem Hintergrund verschmilzt. Im korrigierten Bild gelingt die Segmentierung ohne Probleme (siehe Abschnitt 4.2 für Informationen über Binärsegmentierung). Vor allem bei Beschriftungen, die auf runden Objekten, wie z.B. Flaschen, angebracht sind, muss man häufig eine Shading-Korrektur vornehmen, um die Änderung des Einfallswinkels durch die Krümmung auszugleichen.

Von einem theoretischen Standpunkt aus bleibt natürlich die Schwierigkeit, die wahre Helligkeit des Referenzbildes zu bestimmen. In der Praxis stellt das im All-

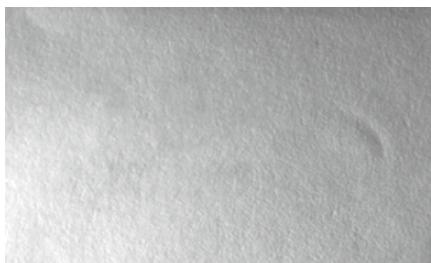
gemeinen kein Problem dar, da es gar nicht darum geht, die *wahre* Helligkeit zu ermitteln, sondern lediglich stabile Bildverhältnisse zu erzeugen, die ein vernünftiges Weiterarbeiten gestatten.



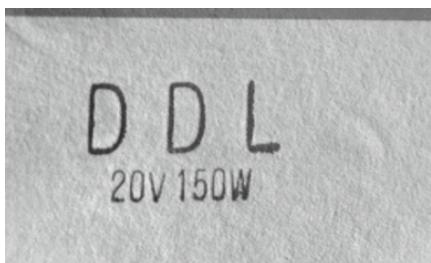
Originalbild



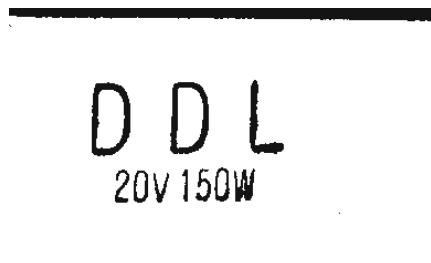
Binärbild



Referenzbild



Korrigiertes Bild



Binärbild

Abbildung 2.7: Bild mit ungleichmäßiger Beleuchtung vor und nach Shading Korrektur

2.3 Lineare Filter

In diesem Abschnitt werden wir das Prinzip linearer Bildfilter und einige gängige Filter mit ihren Verwendungsmöglichkeiten vorstellen. Wir verzichten dabei auf mathematische Analysen des Filterentwurfs und Filterverhaltens. Weitergehende

theoretische Informationen sind beispielsweise in (Jähne 2005), (Nischwitz et al. 2007) oder (Sonka et al. 2008) zu finden.

2.3.1 Lokale Operationen und Bildumgebungen

Lineare Filter gehören zu den gängigsten lokalen Bildoperationen. Unter einer *lokalen Operation* versteht man eine Vorverarbeitungsfunktion, bei der sich der Grauwert eines Pixels im Ergebnisbild aus den Grauwerten einer bestimmten Gruppe von Pixeln im Ursprungsbild ergibt. Diese Gruppe von Pixeln wird als *Umgebung* oder *Nachbarschaft* bezeichnet, bei Filterungen auch als *Operator*, *Filterkern* oder *Filtermaske*.

Das lässt sich folgendermaßen ausdrücken: Der Grauwert g an einer Stelle (x^*, y^*) im Ergebnisbild ist das Ergebnis einer Abbildung $f()$, die auf die Grauwerte g_i aller Pixel im Eingangsbild angewandt wird, die sich in einer (noch nicht näher spezifizierten) Nachbarschaft N^* des Punkts (x^*, y^*) befinden. Üblicherweise ist der Zielpunkt (x^*, y^*) ebenfalls Teil dieser Nachbarschaft, notwendig ist dies jedoch nicht. Formal geschrieben heißt das:

$$g(x^*, y^*) = f(\{g_i(x, y) \mid (x, y) \in N^*\}). \quad (2.10)$$

Die Abbildungsfunktion $f()$ ist zunächst einmal ebenfalls beliebig. Durch entsprechende Spezifikation dieser Funktion entstehen völlig verschiedenartige Filter. Bei der Durchführung einer solchen lokalen Operation wird die definierte Nachbarschaft in der Regel punktweise über das Bild verschoben und so ein Punkt des Ergebnisbildes nach dem anderen berechnet. Dadurch eignen sich diese Verfahren ausgezeichnet für Parallelisierungen, denn da das ursprüngliche Bild nicht verändert wird, kann die Berechnung im Prinzip für alle Bildpunkte gleichzeitig ausgeführt werden – wenn man genügend Prozessoren zur Verfügung hat.

2.3.2 Prinzip linearer Filter

Lineare Filter erhält man, indem für die Abbildungsfunktion $f()$ in Gleichung (2.10) eine gewichtete Summierung der Grauwerte innerhalb der definierten Nachbarschaft verwendet wird:

$$g(x^*, y^*) = \frac{1}{s} \sum_{r=0}^{R-1} \sum_{c=0}^{C-1} f_{cr} g(x^* - dx + c, y^* - dy + r). \quad (2.11)$$

Die Formel sieht auf den ersten Blick vielleicht ein bisschen unübersichtlich aus, an einem konkreten Beispiel erkennt man aber sehr schnell, was sie eigentlich be-

deutet. Zunächst wollen wir die Formel einmal sprachlich etwas aufschlüsseln, dann folgt ein einfaches Beispiel.

dx und dy sind die Abstände zwischen dem Zielpunkt und der linken oberen Ecke der Nachbarschaft, des Filteroperator. Dieser Filteroperator habe R Zeilen (*row*) und C Spalten (*column*). Entsprechend ist r der Zeilenindex und c der Spaltenindex des Filteroperators. Die obige Formel besagt dann nichts anderes, als dass jeder Punkt innerhalb eines Rechtecks der Breite C und der Höhe R , dessen linke obere Ecke am Punkt $(x^* - dx, y^* - dy)$ liegt mit dem zugehörigen Filterkoeffizienten f_{cr} gewichtet und aufsummiert wird. Das Ergebnis wird dann mit einem Faktor s skaliert. Meist ist s die Summe aller Filterkoeffizienten, da dann der mittlere Grauwert des Bildes erhalten bleibt, aber das ist nicht unbedingt zwingend.

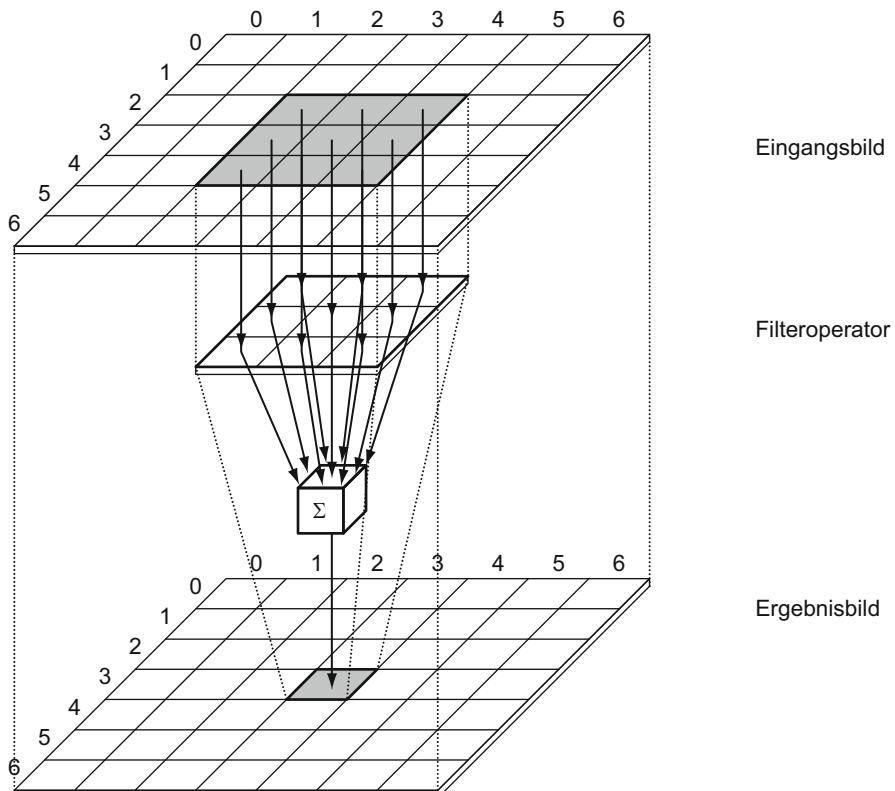


Abbildung 2.8: Prinzip eines linearen Filters

In Abbildung 2.8 sehen Sie dieses Prinzip in einem einfachen Mittelwertfilter umgesetzt. Der Filter hat die gängige Größe von 3×3 Pixeln. Alle Filterkoeffizienten sind eins, der Zielpunkt liegt in der Mitte. Wir haben also folgende Werte:

$$R = 3, \text{ Anzahl der Zeilen des Operators}$$

$C = 3$, Anzahl der Spalten des Operators

$dx = 1$, Abstand zwischen Zielpunkt und linkem Operatorrand

$dy = 1$, Abstand zwischen Zielpunkt und oberem Operatorrand

$s = 9$, Summe der Filterkoeffizienten

Für den Zielpunkt $(3, 3)$ lautet Gleichung (2.11) dann für diesen Spezialfall, da alle Filterkoeffizienten eins sind:

$$\begin{aligned} g(3, 3) &= \frac{1}{9} \sum_{r=0}^2 \sum_{c=0}^2 g_i(3-1+c, 3-1+r) \\ &= \frac{1}{9} (g_i(2, 2) + g_i(2, 3) + g_i(2, 4) + g_i(3, 2) + g_i(3, 3) + g_i(3, 4) \\ &\quad + g_i(4, 2) + g_i(4, 3) + g_i(4, 4)). \end{aligned}$$

und man sieht, dass das Ergebnis einfach der Mittelwert der Grauwerte aller neun Punkte innerhalb des Filteroperators ist. Alle linearen Filter folgen diesem Rechenprinzip, aber durch unterschiedliche Filterkoeffizienten lassen sich sehr vielfältige Effekte erzielen.

Randeffekt: An allen Bildrändern entsteht ein Problem bei der Filterung. Bei der Berechnung des Ergebnispunkts $(0, 0)$ mit einem quadratischen Filter der Größe 3×3 fehlen fünf der erforderlichen neun Punkte, da sie negative Koordinaten hätten. Entsprechendes gilt für alle Punkte in der Nähe des Bildrandes und andere Operatorgrößen. Es gibt verschiedene Möglichkeiten, mit diesem Problem umzugehen:

Verkleinern des Bildes: Bei dieser Methode verzichtet man völlig auf die Randpunkte, da keine exakt gültigen Werte bekannt sind. Dabei verringert sich dann allerdings die Bildgröße durch die Filterung, was sehr unpraktisch ist, da sich bei einem Wechsel der Filtergröße die Bildgröße für alle nachfolgenden Verarbeitungsschritte ändert.

Randpunkte festsetzen: In diesem Fall werden alle Randpunkte auf einen festen Grauwert gesetzt, meist null (schwarz). Die Bildgröße bleibt dabei erhalten. Der Hauptnachteil dieses Verfahrens liegt darin, dass unter Umständen Pseudokanten entstehen, die eine Segmentierung empfindlich stören können.

Verfügbare Punkte benutzen: Man kann Werte für die Randpunkte aus allen im Filteroperator verfügbaren Bildpunkten berechnen. Bei einfachen rechteckigen Mittelwertfiltern funktioniert das normalerweise recht gut, bei komplexeren Filterformen und -algorithmen kann es jedoch wieder zur Bildung von Pseudokanten kommen.

Gültige Randpunkte wiederholen: Dieses Verfahren ersetzt alle nicht gültigen Randpunkte durch den Grauwert des jeweils nächsten Gültigen. Bei dem obigen Beispiel eines 3×3 -Filters ist der erste Punkt, der sich vollständig berechnen lässt, der Punkt (1, 1). Dessen Grauwert wird dann auch für die angrenzenden Randpunkte (0, 0), (1, 0) und (0, 1) verwendet. Entsprechend wird der Grauwert von Punkt (2, 1) für Punkt (2, 0) benutzt usw. Abbildung 2.9 zeigt dieses Verfahren. Die Filterumgebung ist grau angedeutet, die Pfeile bezeichnen die Kopie des Grauwertes von einem gültigen Punkt in einen Randpunkt. Auf diese Weise stellt man die Stetigkeit der Grauwertinformation am Rand sicher und erhält in den meisten Fällen günstige Bildeigenschaften für die Weiterverarbeitung.

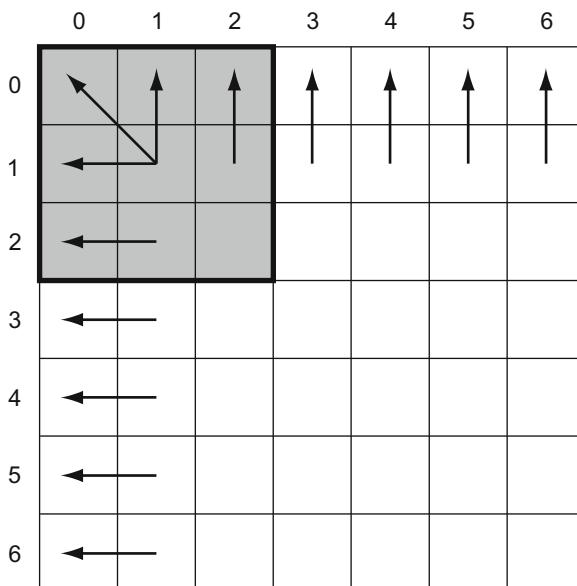


Abbildung 2.9: Behandlung von Randpunkten bei der Filterung

2.3.3 Glättungsfilter

Als Glättungsfilter bezeichnet man im Allgemein Tiefpassfilter, die zur Unterdrückung von Rauschen oder feinen Details benutzt werden können. Ihre Funktion beruht auf Mittelwertbildungen. Da Bildrauschen sich in einer zufälligen Veränderung des Grauwerts einzelner Punkte äußert, lässt es sich durch Mitteln der Helligkeit in kleinen Bildbereichen unterdrücken. Wenn sich beispielsweise auf einer hellen Fläche ein dunkles Störpixel befindet, so wird durch die Mittelung die Helligkeit der Fläche ein wenig abnehmen. Der Grauwert des Störpixels aber wird durch die umliegenden hellen Punkte deutlich angehoben werden.

Rechteckfilter: Der einfachste Glättungsfilter ist der Rechteckfilter. Er wird oft auch einfach als Mittelwertfilter bezeichnet, da er die aus der Statistik bekannte Form der Mittelwertbildung durchführt: Alle Grauwerte innerhalb des Operators werden summiert und durch die Anzahl der Operatorpunkte dividiert. Der Gewichtsfaktor f_{cr} aus Gleichung (2.11) ist also eins und kann daher auch entfallen.

Wir wollen dieses einfache Beispiel dazu benutzen, die Arbeitsweise eines linearen Filters noch einmal im Detail zu zeigen, da man dieses Prinzip auch bei vielen anderen Filtertypen wieder findet. Wir verwenden dazu wie in Gleichung (2.12) und Abbildung 2.8 einen Rechteckfilter der Größe 3×3 , d. h. die Filtermaske sieht folgendermaßen aus:

$$\frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \quad (2.12)$$

Die Transformationsgleichung für einen Bildpunkt ergibt sich aus Gleichung (2.11) dann zu

$$g(x^*, y^*) = \frac{1}{9} \sum_{r=0}^2 \sum_{c=0}^2 f_{cr} g(x^* - 1 + c, y^* - 1 + r). \quad (2.13)$$

Abbildung 2.10 zeigt den Effekt dieses Filters auf ein künstlich verrauschtes Bild eines schwarzen Quadrates auf weißem Grund. Man sieht, dass das Rauschen weniger ausgeprägt ist, die einzelnen Störpixel sind verbreitert, unterscheiden sich aber weniger stark vom weißen Hintergrund als im Originalbild. Gleichzeitig erkennt man aber, dass sich auch die scharfen Kanten des Quadrates verbreitert haben. Sie bestehen jetzt aus mehreren Linien, deren Helligkeit von außen nach innen abnimmt.

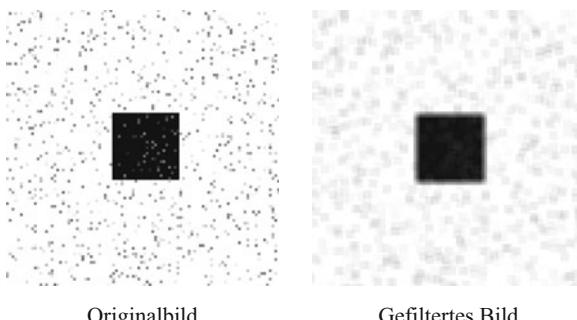


Abbildung 2.10: Wirkung eines 3×3 -Rechteckfilters

Diese Kantenunschärfe kann man besonders deutlich an einem Bild mit periodischen Strukturen erkennen, wie in Abbildung 2.11. Die Abbildung zeigt noch einige andere Eigenschaften dieses einfachen Filters:

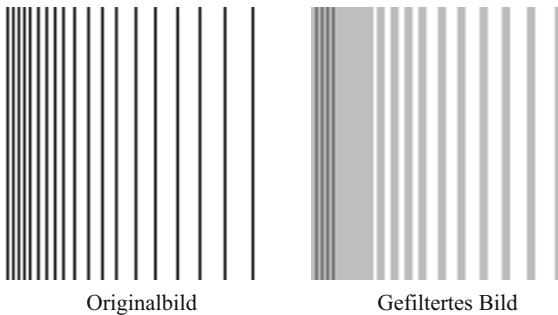


Abbildung 2.11: Wirkung eines 3×3 -Rechteckfilters auf periodische Strukturen

- Die weit auseinander liegenden Linien rechts im Bild bleiben als Linien erhalten, sie werden jedoch verbreitert und aufgehellt. Der Ort der Linien lässt sich nach der Filterung also nicht mehr präzise lokalisieren.
- Linien, deren Abstand gerade der Breite des Filterkerns entspricht, verschmelzen vollständig. Dies entspricht der Grenzwellenlänge des Filters.
- Linien, deren Abstand geringer ist als die Breite des Filterkerns, werden nicht vollständig unterdrückt. Ihre Amplitude sinkt vielmehr nur um einen Faktor drei, d. h. von 255 auf 85. Die folgende Berechnung, die nach beiden Seiten entsprechend der Ausdehnung des Bildes fortzusetzen ist, zeigt dies explizit. Man sieht also, dass die Sperrwirkung dieses Tiefpassfilters nicht besonders gut ist. In (Jähne 2005) werden diese Eigenschaften anhand einer Berechnung des Frequenzverhaltens mithilfe einer Fouriertransformation gezeigt.

$$\begin{array}{ccccccc}
 & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\
 \dots & 255 & 0 & 255 & \dots & \dots & 170 & 85 & 170 & \dots \\
 \dots & 255 & 0 & 255 & \dots & * \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} = & \dots & 170 & 85 & 170 & \dots \\
 \dots & 255 & 0 & 225 & \dots & \dots & 170 & 85 & 170 & \dots \\
 & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots
 \end{array}$$

Abbildung 2.12 zeigt die Nachteile dieses sehr einfachen Filtertyps anhand eines Barcodes. Die Zerstörung der Kanten durch das Rauschen lässt keine sichere Lesung des Barcodes mehr zu. Das Rauschen erscheint zwar nach der Filterung gedämpft, durch die Kantenunschärfe sind die Zwischenräume jedoch stellenweise so stark verringert, dass weiterhin keine sichere Lesung möglich ist. Dieser Effekt wird natürlich um so stärker, je größer der Filterkern wird.

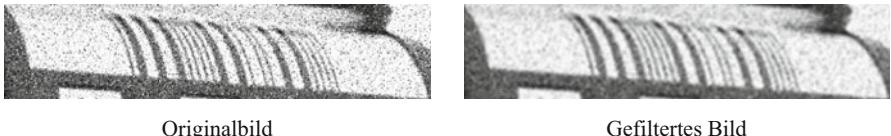


Abbildung 2.12: Wirkung eines 3×3 -Rechteckfilters auf einen verrauschten Barcode

Reduzierter Rechteckfilter: Bildfilterungen sind rechenaufwändig und zeitraubend. Man kann jedoch ohne nennenswerte Einbußen an Filterwirkung auf etwa die Hälfte der Operatorpunkte verzichten, indem man nur jeden zweiten Punkt bearbeitet. Um die Form von Gleichung 2.11 beizubehalten, schreibt man einen solchen Filterkern mit Nullen an der Stelle der entfallenen Operatorpunkte. Bei der Ausführung der Filterung überspringt man diese Punkte natürlich einfach. Ein reduzierter 3×3 -Filterkern hat folgendes Aussehen:

$$\frac{1}{5} \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}. \quad (2.14)$$

Die Wirkung auf lineare, periodische Strukturen verändert sich relativ stark, wie man an Abbildung 2.13 sieht. Die dicht benachbarten Linien werden noch weniger stark gefiltert als beim vollständigen 3×3 -Filter, die vollständige Unterdrückung der Linien im Abstand von drei Bildpunkten entfällt ebenfalls. Durch die kreuzförmige Struktur des Filters zeigen die weiter auseinander liegenden Linien eine doppelte Kantenstruktur.

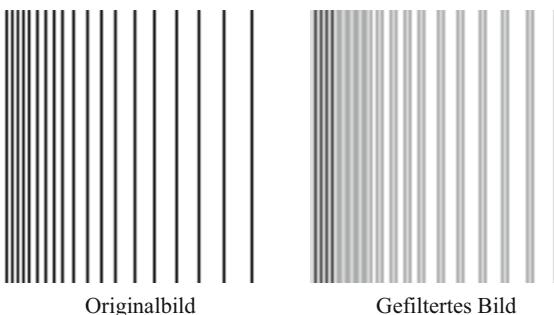


Abbildung 2.13: Wirkung eines reduzierten 3×3 -Rechteckfilters auf periodische Strukturen

Trotz dieser Verringerung der Filterwirkung auf periodische Strukturen lassen sich solche reduzierten Filter gut auf „normale“ Bildszenen anwenden. Das hängt damit zusammen, dass reale Bildszenen eine hohe Autokorrelation aufweisen, d. h. benachbarte Bildpunkte unterscheiden sich normalerweise nicht sehr stark. Die Autokorrelation der Linienbilder ist hingegen sehr gering, da bei den im Abstand

von einem Bildpunkt angeordneten Linien eine Verschiebung um einen Bildpunkt zu einer maximalen Veränderung führt. Bei den eher flächigen Objekten realer Bildszenen hat man bei nahezu halbiert Rechenzeit annähernd dieselbe Filterwirkung, abgesehen von der Aufprägung einer leichten Textur durch die kreuzförmige Struktur des Filters. Abbildung 2.14 zeigt dies anhand des oben bereits verwendeten schwarzen Quadrats.

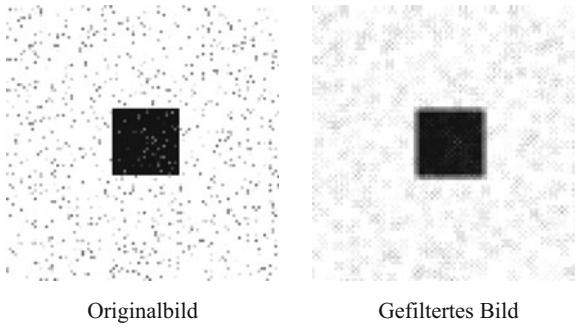


Abbildung 2.14: Wirkung eines reduzierten 3×3 -Rechteckfilters auf flächige Gebilde

Mittelwertfilter mit angepassten Operatoren: Von der eben gezeigten Reduzierung des Filterkerns durch Weglassen jedes zweiten Operatorpunkts ist es nur noch ein kleiner Schritt hin zu Filterkernen, die speziell an bestimmte Bildstrukturen und Anwendungen angepasst sind. Abbildung 2.15 zeigt die Wirkung eines Filterkerns, der dazu entworfen wurde, Kanten zu glätten, die in einem ganz bestimmten Winkel durch das Bild verlaufen. Bei dem Prüfteil handelt es sich um eine Gitarrensaite, deren Metallumwicklung fehlerhaft ist. Zur Messung der Breite der Abstände zwischen den einzelnen Wicklungen muss der mittlere Abstand zwischen den Kanten der hellen Objekte gemessen werden. Dies ist wesentlich leichter, wenn die Kanten möglichst glatt sind. Daher wurde ein Filterkern entworfen, der im gleichen Winkel verläuft wie die betreffenden Kanten. Der Filterkern hat eine Gesamtdimension von 6×11 Punkten. Der Zielpunkt ist mit einem zusätzlichen weißen Rahmen versehen.

Programmhinweis: In NeuroCheck können Sie für jeden linearen und viele nichtlineare Filtertypen angepasste Operatoren entwerfen, indem Sie im Prüfschritt *Bild filtern* den Filtertyp „Benutzerdefiniert“ wählen. Sie können dann einen der Standardfilter als Vorgabe für den Operator verwenden und diesen beliebig modifizieren.

Gaußfilter: Anhand einer Fouriertransformation der Übertragungsfunktion kann man zeigen, dass ein optimaler Glättungsfilter die Form einer Gaußschen Glockenkurve aufweisen muss. Im orts- und wertdiskreten Raum eines digitalen Bildes nähert man diese Funktion mit ganzzahligen Filterkoeffizienten an. Gleichung (2.15) zeigt Gauß-Filtermasken der Größe 3×3 und 5×5 . Größere Masken sind in (Russ 2007) zu finden, allerdings sollte man bedenken, dass auch bei diesem

optimierten Filter ein größerer Operator zu größerer Unschärfe führt und dass die Rechenzeit quadratisch mit der Seitenlänge des Operators ansteigt.

$$G_3 = \frac{1}{28} \begin{pmatrix} 1 & 4 & 1 \\ 4 & 12 & 4 \\ 1 & 4 & 1 \end{pmatrix},$$

$$G_5 = \frac{1}{121} \begin{pmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 7 & 11 & 7 & 2 \\ 3 & 11 & 17 & 11 & 3 \\ 2 & 7 & 11 & 7 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{pmatrix}. \quad (2.15)$$

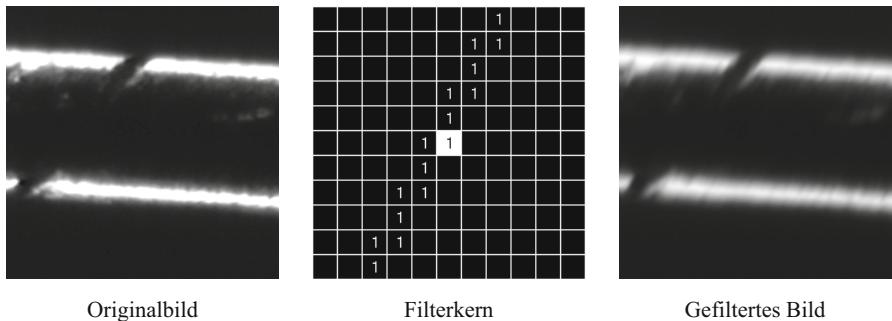


Abbildung 2.15: Wirkung eines speziell angepassten Mittelwertfilters

Abbildung 2.16 zeigt einen theoretischen Gauß-Operator und die Approximation durch die beiden Filtermasken aus Gleichung (2.15). Durch die Diskretisierung entsteht natürlich ein um so größerer Fehler, je kleiner die Filtermaske ist.

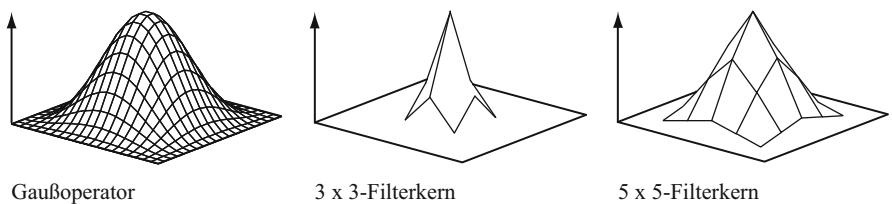


Abbildung 2.16: Gauß-Filteroperator und diskrete Näherungen

Abbildung 2.17 zeigt die Wirkung eines Gauß-Filters der Größe 3×3 auf die Beispielbilder aus den Abbildungen 2.10 und 2.11. Natürlich tritt auch hier eine Verschiebung der Kanten auf, es kommt jedoch nicht zu völligen Verschmelzungen

und durch entsprechende Interpolationsmodelle bleiben die Kantenpositionen feststellbar.

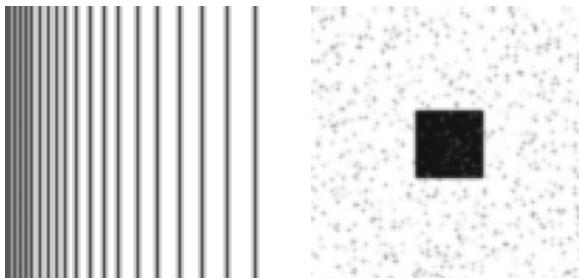


Abbildung 2.17: Wirkung eines 3×3 -Gaußfilters auf periodische Strukturen und verrauschte Objekte

Abbildung 2.18 zeigt die Filterung des verrauschten Barcodes aus Abbildung 2.12 mit einem 5×5 -Gaußfilter. Es bleibt genügend Kantenschärfe erhalten, um den Barcode zu identifizieren.

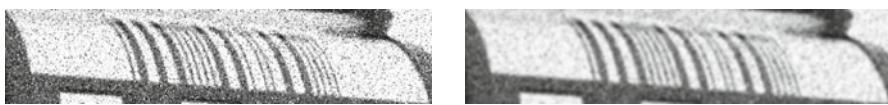


Abbildung 2.18: Filterung eines verrauschten Barcodes mit einem 5×5 -Gaußfilter

Derartige Filterungen sind sehr zeitaufwändig. (Jähne 2005) führt eine Reihe von Möglichkeiten auf, Eigenschaften der Filter zur Beschleunigung der Berechnung auszunutzen. Auf Filter angepasster Form lassen sich diese Methoden allerdings gewöhnlich nicht anwenden, da Eigenschaften wie Symmetrie und Zerlegbarkeit in separate horizontale und vertikale Operationen bei Veränderungen der Operatormform verloren gehen können.

Programmhinweis: In **NeuroCheck** wird die Berechnung des gefilterten Bildes im Prüfschritt *Bild filtern* bei jedem linearen und nichtlineare Filtertyp automatisch auf sämtliche im Rechner verfügbaren Prozessorkerne verteilt.

2.3.4 Kantenfilter

Glättungsfilter unterdrücken Variationen im Bild. Kanten sind nichts anderes als solche Variationen, schnelle Veränderungen des Grauwertes auf kleinem Raum. Der Gedanke liegt nahe, Hochpassfilter zur Hervorhebung von Kanten zu verwenden. Tatsächlich ist dies möglich. Es handelt sich dabei um Differenzfilter. Numerisch bestimmt man im einfachsten Fall die Änderung einer Funktion durch Berechnung der Differenz benachbarter Funktionswerte. Nach diesem Prinzip

werden auch Kantenfilter konstruiert, die zur Herstellung von Differenzen negativen Filterkoeffizienten enthalten müssen. Damit ergibt sich wieder das gleiche Problem wie bei der Bildsubtraktion in Abschnitt 2.2.2: Die Ergebniswerte können nicht nur wie bei Glättungsfilters größer sein als der höchste Grauwert, sondern auch negativ werden. Die einfache Skalierungsmethode einer Division durch die Summe der Filterkoeffizienten genügt daher nicht unbedingt zur Abbildung des Ergebnisses in das normale Grauwertintervall.

Prinzip eines Kantenfilters: Wir zeigen das Prinzip an einem einfachen Beispiel mit folgender Filtermaske:

$$\frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}. \quad (2.16)$$

Die Wirkung dieses Filters ist eine numerische Annäherung des Grauwertgradienten in X-Richtung. Auf das Problem möglicher negativer Werte gehen wir gleich noch ein. Im Augenblick verwenden wir zunächst einmal die absoluten Beträge der Ergebniswerte. Für die Skalierung auf den Grauwertbereich von [0, 255] wird nach Gleichung (2.16) der Normierungsfaktor 1/4 verwendet³. Abbildung 2.19 zeigt das Ergebnis der Filterung eines einfachen Objekts mit diesem Operator.

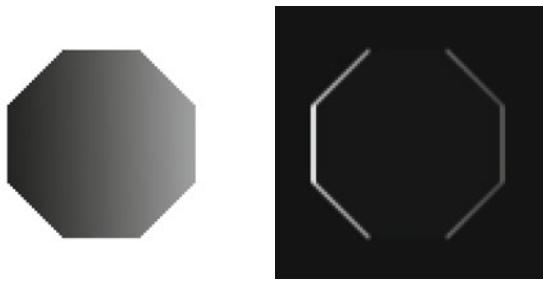


Abbildung 2.19: Wirkung eines einfachen Kantenfilters

Man erkennt folgende Eigenschaften:

- Im völlig homogenen Hintergrund heben sich die negativen und positiven Beiträge bei der Filterung auf, das Ergebnis ist null. Der Hintergrund wird also

³ Wir weichen hier von der sonstigen Literatur ab, die einen Skalierungsfaktor von 1/8 verwendet. Wir werden aber gleich sehen, dass 1/4 ausreichend ist, die Begrenzung auf den Grauwertbereich sicherzustellen

homogen schwarz. Die folgende Gleichung zeigt das Ergebnis der Berechnung eines einzelnen Hintergrundpunkts⁴.

$$\begin{vmatrix} 255 & 255 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 255 & 255 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 255 & 255 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \end{vmatrix} = 0.$$

- An exakt waagerechten Kanten tritt in der Differenzierungsrichtung des Filters ebenfalls keine Veränderung auf. Die punktweisen Multiplikationen heben sich daher gerade auf und waagerechte Kanten verschwinden, wie die folgende Gleichung zeigt:

$$\begin{vmatrix} 255 & 255 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 128 & 128 & 128 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 0 & 0 & 0 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \end{vmatrix} = 511 - 511 = 0.$$

- An exakt senkrechten Kanten tritt der maximale Effekt auf, da jede Zeile des Filteroperators einen Beitrag zum Endergebnis leistet. Die folgende Gleichung zeigt das anhand einer Kante, die vom Grauwert null (schwarz) über einen Zwischenschritt auf den Grauwert 255 (weiß) ansteigt.

$$\begin{vmatrix} 0 & 128 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 0 & 128 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \\ 0 & 128 & 255 & \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \end{vmatrix} = \frac{1}{4} |-1020| = 255.$$

Dieses Beispiel zeigt auch gleich noch, warum die Skalierung mit 1/4 ausreichend ist. Es gibt keine Möglichkeit, mit dieser Filtermaske einen höheren Absolutbetrag als 1020 in einem Grauwertbild zu erhalten. Jede Erhöhung der Grauwerte *vor* der Kante führt zu einer Verringerung des Betrages, da zusätzliche positive Produkte addiert werden, jede Absenkung der Grauwerte *hinter* der Kante senkt ebenfalls den Betrag, da die negativen Produkte kleiner werden.

- An diagonalen Kanten wirken die Anteile der ersten und letzten Zeile des Filteroperators gegeneinander und schwächen so die Wirkung ab, wie die folgende Gleichung zeigt.

⁴ Man beachte, dass dies keine Matrixmultiplikation ist, bei der die Skalarprodukte von Zeilen und Spalten berechnet werden. Die Multiplikation erfolgt hier punktweise.

$$\begin{vmatrix} 0 & 0 & 128 \\ 0 & 128 & 255 \times \frac{1}{4} \\ 128 & 255 & 255 \end{vmatrix} = \frac{1}{4} \left| -765 \right| = 191,25.$$

- Das Objekt ist links dunkler als rechts, d. h. die Kantensteilheit ist links höher. Im Ergebnisbild der Filterung sind daher die vorderen Kanten heller als die hinteren.

Skalierungsmethoden: Nun kommen wir zu der Frage, wie man mit den negativen Ergebniswerten solcher Differenzfilter umgehen kann. Ähnlich wie bei der Bildsubtraktion kann man auch hier mehrere Skalierungsmodi unterscheiden, deren Wirkung Abbildung 2.20 anhand einer Filterung des Beispielbildes aus Abbildung 2.19 zeigt.

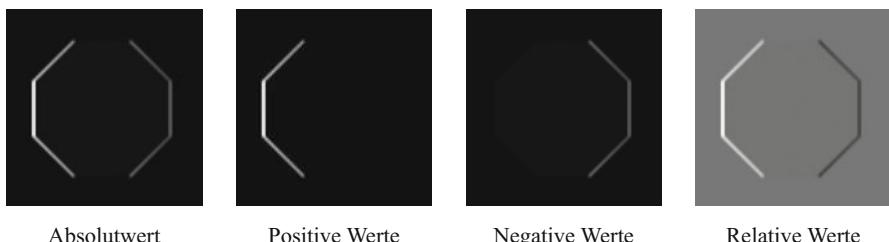


Abbildung 2.20: Wirkung verschiedener Skalierungsmodi bei der Kantenfilterung

Die verschiedenen Skalierungsmodi haben folgende Wirkung:

Absolutwerte: Der endgültige Grauwert wird durch lineare Skalierung des Betrages des ursprünglichen Ergebniswertes auf das Grauwertintervall berechnet. Informationen über die Kantenrichtung gehen dabei verloren, da nicht mehr zu erkennen ist, welches Vorzeichen das ursprüngliche Filterergebnis hatte.

Positive Werte: Nur die positiven Ergebniswerte werden benutzt und linear auf das Grauwertintervall skaliert. Damit findet eine Auswahl der Kantenrichtung statt. Welche Kanten erhalten bleiben, hängt von der Definition des Filteroperators ab, im Beispiel sind es die von Hell auf Dunkel fallenden Kanten.

Negative Werte: Nur die negativen Ergebniswerte werden benutzt, alle positiven Ergebnisse werden auf null gesetzt. Von den verbleibenden negativen Werten werden die Beträge benutzt und auf das Grauwertintervall skaliert. Gegenüber der Verwendung der positiven Werte werden genau die entgegen gesetzten Kanten ermittelt.

Relative Werte: Hier werden alle Ergebniswerte zunächst in ein zu Null symmetrisches Intervall, z.B. $[-128, 127]$, skaliert, dann in das Intervall $[0, 255]$ additiv verschoben. Alle Bereiche ohne Bildvariationen werden dadurch auf einen mittleren Wert gesetzt.

ren Grauwert abgebildet und Kanten verschiedener Richtung lassen sich danach unterscheiden, ob sie hell oder dunkel im Ergebnisbild erscheinen.

Je nach Skalierungsmodus muss die Gleichung 2.11 entsprechend etwas abgewandelt werden, um den endgültigen Grauwert des Ergebnisbildes zu erhalten. Am Prinzip ändert sich jedoch nichts.

Es gibt in der Literatur eine unüberschaubare Vielfalt von Kantenfiltern, von denen wir hier nur kurz einige erwähnen wollen.

Laplace-Operator: Im Gegensatz zu dem eben betrachteten einfachen Operator aus Gleichung (2.16) erfassen die Laplace-Operatoren alle Kantenrichtungen gleichzeitig. Gleichung (2.17) zeigt zwei Laplace-Operatoren verschiedener Größe. Man findet diese Operatoren übrigens auch mit umgekehrtem Vorzeichen, was keinen Unterschied macht, da sie ja nicht richtungsabhängig sind.

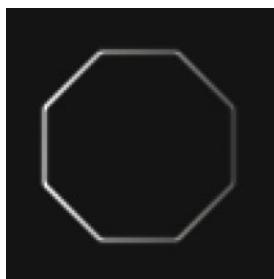
$$L_3 = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}, \quad (2.17)$$

$$L_5 = \begin{pmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

Abbildung 2.21 zeigt den Effekt einer Filterung mit Laplace-Operatoren bei Verwendung des Absolutwertes. Da diese Filter richtungsunabhängig sind, sind andere Skalierungsmethoden hier nicht sinnvoll.



3 x 3-Filter



5 x 5-Filter

Abbildung 2.21: Extraktion von Kanten mit Laplace-Filtern (Kontrast auf [0; 255] verstärkt)

Sobel-Operator: Die Sobel-Methode der Kantenextraktion verwendet eine Reihe aufeinander abgestimmter Filteroperatoren, die Kanten in verschiedenen Richtun-

gen extrahieren. Eine typische Zusammenstellung ist z. B. in (Sonka et al. 2008) zu finden (wobei wir hier, wie schon in Gleichung (2.16), mit 1/4 statt 1/8 skalieren):

$$\begin{aligned} S_1 &= \frac{1}{4} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}, & S_3 &= \frac{1}{4} \begin{pmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{pmatrix}, \\ S_2 &= \frac{1}{4} \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}, & S_4 &= \frac{1}{4} \begin{pmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{pmatrix}. \end{aligned} \quad (2.18)$$

Abbildung 2.22 zeigt das Ergebnis von Filterungen mit diesen Operatoren bei Verwendung der Absolutwertskalierung. Wenn alle Kantenrichtungen festgestellt werden sollen, kann man die einzelnen Ergebnisbilder nachträglich mit einem Maximumoperator entsprechend Abschnitt 2.2.3 verknüpfen. Da alle Kanten hell hervortreten, sind sie im Maximumbild sämtlich enthalten.

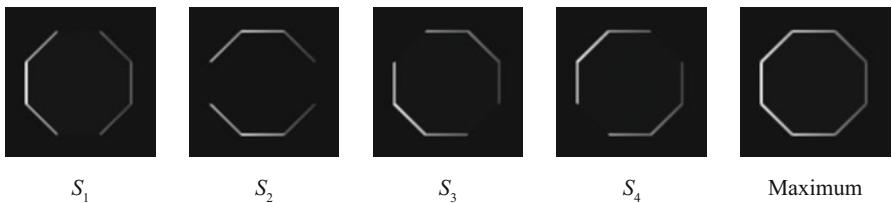


Abbildung 2.22: Kantenextraktion mit verschiedenen Sobeloperatoren und Verknüpfung mit Maximumoperator

Selbstverständlich sind Sobel-Operatoren auch mit größeren Abmessungen als 3×3 möglich.

Sobel-Betragsfilter: Wir erwähnen diesen Filter hier wegen seiner engen Beziehung zu den Sobel-Operatoren, es ist aber kein linearer Filter, sondern eine nicht-lineare Kombination zweier Sobelfilter. Man erreicht in etwa dasselbe Resultat wie bei einer Verknüpfung von vier Sobelfilterungen mit dem Maximumoperator, indem man den Betrag der Gradienten in zwei zueinander orthogonalen Richtungen berechnet. Dazu filtert man das Bild mit zwei zueinander orthogonalen Sobel-Operatoren, also beispielsweise S_1 und S_2 aus Gleichung (2.18), und bildet den Betrag der Ableitungen in beiden Richtungen. Die Filterung mit S_1 ermittelt eine Näherung für den Grauwertgradienten in x-Richtung, d. h. $(\partial g / \partial x)$, die Filterung mit S_2 entsprechend in y-Richtung. Das Ergebnis des Sobel-Betragsfilters ist dann:

$$\text{Magnitude} = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}. \quad (2.19)$$

2.4 Medianfilter

Beim einfachen Mittelwert- oder Rechteckfilter hatten wir gesehen, dass die Mittelwertbildung zu einer starken Kantenunschärfe führt. An der Kante entstehen Zwischenwerte der tatsächlich im Bild vorhandenen Grauwerte. Außerdem können Störpixel, wie wir in Abbildung 2.10 gesehen haben, zu deutlichen Veränderungen des Grauwertes in ihrer Umgebung führen. Zur Reduzierung dieser Empfindlichkeit gegenüber Ausreißern verwendet man in der Statistik den Median. Der Median ist gleichbedeutend mit dem Zentralwert einer Zahlenmenge, hier der durch den Filteroperator gegebenen Menge von Grauwerten. Wenn man diese Grauwerte ihrer Größe nach ordnet:

$$g_1 \leq g_2 \leq \dots \leq g_n,$$

dann ist der Median nach (Bronstein et al. 2005) definiert als

$$\bar{g} = \begin{cases} g_{\frac{n+1}{2}} & , \text{für ungerades } n \\ \frac{1}{2}(g_{\frac{n}{2}} + g_{\frac{n}{2}+1}) & , \text{für gerades } n . \end{cases} \quad (2.20)$$

Der Medianfilter ist ein sehr häufig angewandter Spezialfall sogenannter Rangordnungsfilter. In der Statistik spricht man hier von Quantilen. Ein Quantil x_q ist dadurch definiert, dass q Werte einer diskreten Folge kleiner oder gleich x_q sein müssen. Der Median ist nach dieser Definition ein $x_{1/2}$ -Quantil. Prinzipiell kann man beliebige Quantile verwenden und dadurch entsprechend andere Effekte erzielen, der Medianfilter ist jedoch bei weitem der häufigste Fall.

Aufgrund seiner Unempfindlichkeit gegenüber Ausreißern eignet sich der Medianfilter besonders zur Eliminierung punktförmiger Störungen bei gleichzeitiger Erhaltung der Kantenbeschärfe. Aufgrund der erforderlichen Sortierung ist er aber bei gleicher Operatorgröße zeitaufwändiger als ein Mittelwertfilter.

Abbildung 2.23 zeigt die Wirkung eines 3×3 -Medianfilters auf ein verrauschtes Bild. Man sieht, dass nahezu alle punktförmigen Störungen beseitigt wurden, die Kantenbeschärfe aber von kleinen verbliebenen Störungen abgesehen erhalten geblieben ist.

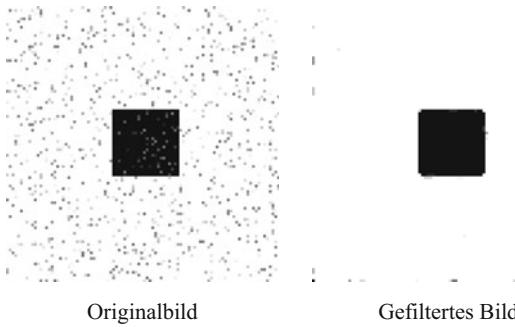


Abbildung 2.23: Wirkung eines 3×3 -Medianfilters auf flächige Gebilde

Ein vollständig ausgefüllter 3×3 -Medianfilter ordnet jeweils neun Werte aus einer quadratischen Umgebung jedes Bildpunkts in eine Folge und verwendet dann den fünften Wert als Grauwert des Punkts im Ergebnisbild. Wie ein Mittelwertfilter kann auch ein Medianfilter beliebige Form annehmen. Mit einem waagerechten 3×1 -Medianfilter lassen sich beispielsweise senkrechte Linien der Breite eins vollständig beseitigen, wenn sie ausreichend weit voneinander entfernt sind. Waagerechte Linien werden hingegen nicht beeinflusst, wie Abbildung 2.24 zeigt.

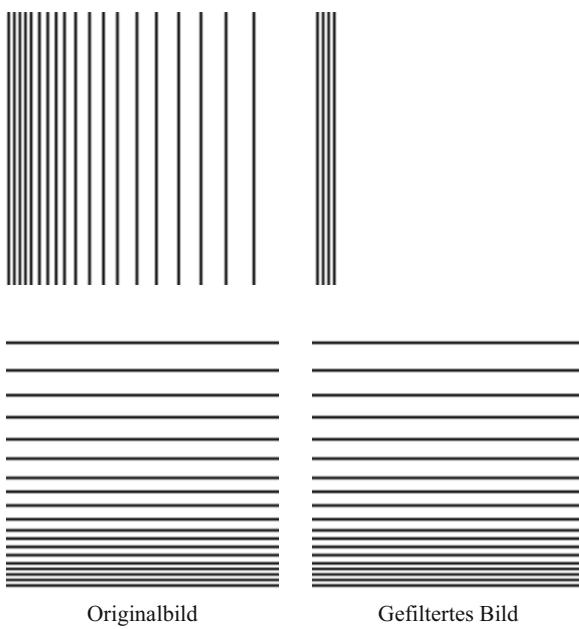


Abbildung 2.24: Wirkung eines 3×1 -Medianfilters auf Linienstrukturen

2.5 Morphologische Filter

Die bisher betrachteten Vorverarbeitungsmethoden verwendeten hauptsächlich Methoden aus der Matrizenrechnung und der Differentialrechnung. Eine weitere Klasse nichtlinearer Filter beruht dagegen auf der mathematischen Morphologie. Morphologie heißt wörtlich „Formenlehre“. Die mathematische Morphologie beruht auf mengentheoretischen Überlegungen, die z. B. in (Gonzalez und Woods 2008) und (Sonka et al. 2008) dargestellt sind.

Für die praktische Anwendung in der Bildvorverarbeitung werden morphologische Filter meist in der Form von Minimum- und Maximum-Operationen implementiert. Nach (Nischwitz et al. 2007) weisen die morphologischen Basisoperationen *Erosion* und *Dilatation* dem Ergebnisbildpunkt jeweils den minimalen, bzw. maximalen Grauwert innerhalb des Filteroperators zu.

Erosion: Wenn wir wie bei der Definition der lokalen Operationen in Gleichung (2.10) die durch den Filteroperator gegebene Umgebung mit N^* bezeichnen ergibt sich der Ergebnisgrauwert g_e bei der Erosion aus den Grauwerten g_i des Eingangsbildes zu:

$$g_e(x^*, y^*) = \min \{ g_i(x, y) | (x, y) \in N^* \}. \quad (2.21)$$

Der Name Erosion röhrt von dem Effekt her, dass diese Operation helle Strukturen verdünnt. Sobald sich innerhalb des Filteroperators auch nur ein dunkler Punkt befindet, wird der Zentralpunkt ebenso dunkel. Ein flächiges helles Objekt wird folglich rundherum abgetragen, also erodiert. Umgekehrt kann man auch sagen, dass eine Erosion dunkle Strukturen verdickt. Abbildung 2.25 zeigt die Wirkung eines 3×3 -Erosionsfilters. Jedes Störpixel wird auf die volle Größe des Operators verdickt. Auch das zentrale Quadrat wird entsprechend größer, was sich gut daran erkennen lässt, dass ein weißer Rand stehen bleibt, wenn man das gefilterte Bild vom Originalbild subtrahiert (die Subtraktion wurde hier mit Relativwertskalierung durchgeführt, d. h. alle Punkte, die in beiden Bildern identisch sind, erhalten einen mittleren Grauwert; siehe hierzu Abschnitt 2.2.2).

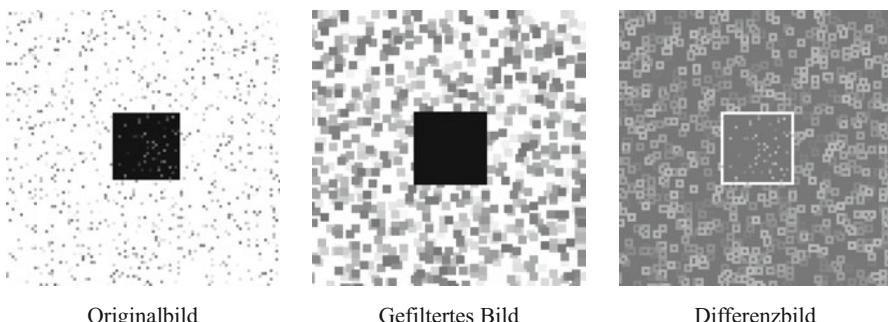


Abbildung 2.25: Wirkung eines 3×3 -Erosionsfilters

Dilatation: Die Dilatation ist die reziproke Operation zur Erosion, d. h. es gilt für den Ergebnisgrauwert g_d der Dilatation:

$$g_d(x^*, y^*) = \max\{g_i(x, y) \mid (x, y) \in N^*\}. \quad (2.22)$$

Man beachte, dass die Dilatation zwar von der Definition her das Gegenteil einer Erosion darstellt, es sich aber nicht um umkehrbare Operationen handelt. Die Durchführung einer Dilatation nach einer Erosion kann den Originalzustand nicht wiederherstellen. Dasselbe gilt für eine Erosion, die einer Dilatation folgt. Beide Kombinationen sind eigene Operationen, auf die wir gleich noch eingehen. Den Effekt einer Dilatation mit einem 3×3 -Operator zeigt Abbildung 2.26. Die punktförmigen Störungen sind völlig verschwunden, da sich in ihrer Umgebung immer ein heller Hintergrundbildpunkt befindet. Das schwarze Quadrat ist entsprechend kleiner geworden, was sich im Differenzbild als dunkler Rand niederschlägt.

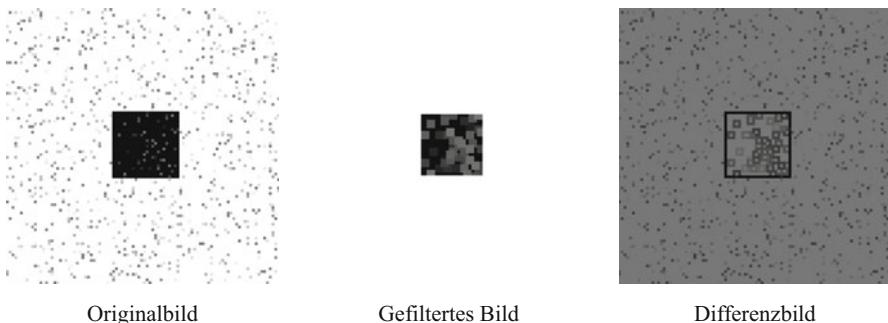


Abbildung 2.26: Wirkung eines 3×3 -Dilatationsfilters

Man sieht also, dass die Dilatation eine Verbreiterung heller und eine Verdünnung dunkler Strukturen zur Folge hat. Diese Operation kann daher gut dafür benutzt werden, das Zerfallen heller Zeichen zu verhindern, wie wir in Abschnitt 5.3 sehen werden.

Opening: Unter dem *Opening* versteht man die Abfolge einer Erosion und einer Dilatation. Die Erosion schließt Lücken in dunklen Strukturen, während die Dilatation die Verdickung dunkler Objekte wieder rückgängig macht. Da bei der Erosion eintretende Vergrößerung der Fläche dunkler Objekte wird also durch die Dilatation aufgehoben. Wenn die Lücken in den dunklen Strukturen im Verhältnis zum Filteroperator so klein waren, dass sie vollständig geschlossen wurden und kein heller Punkt übrig blieb, können sie durch die Dilatation nicht wieder geöffnet werden. Der Name Opening röhrt daher, dass dünne helle Strukturen mit dieser Operation zerlegt werden können, wobei Objektmerkmale wie die Fläche oder die Grunddimensionen erhalten bleiben. Abbildung 2.27 zeigt die Wirkung des Openings. Die hellen Störpunkte innerhalb des Quadrates sind beseitigt, aber die

Abmessungen des Quadrates haben sich nicht geändert. Daher erscheint im Differenzbild keine quadratische Randlinie.



Abbildung 2.27: Wirkung eines 3×3 -Openingfilters

Closing: Die Operation *Closing* verläuft genau umgekehrt zum Opening: Zuerst wird eine Dilatation durchgeführt, dann eine Erosion. Die Wirkung kehrt sich entsprechend ebenfalls um: Lücken in hellen Objekten werden ohne Veränderung der Objektfläche geschlossen (daher der Name). Abbildung 2.28 zeigt diese Wirkung anhand eines mit Rauschen überzogenen weißen Quadrats. Die Dilatation überschreibt die dunklen Störungen im Objekt mit weißen Bildpunkten, die nachfolgende Erosion macht die Veränderung der Dimensionen des Quadrats rückgängig, so dass das Differenzbild keinen Rand zeigt.



Abbildung 2.28: Wirkung eines 3×3 -Closingfilters

2.6 Andere nichtlineare Filter

Es gibt noch eine Vielzahl weiterer Techniken zur Bildverbesserung. Die wirksamsten verwenden Vorwissen über die Art der Störung des Bildes, wie z. B. die

Wiener-Filterung nach (Rosenfeld und Kak 1982), (Sonka et al. 2008), die davon ausgeht, dass die statistischen Parameter des Rauschens bekannt sind. Unter dieser Voraussetzung kann man damit bemerkenswerte Bildrestorationen durchführen, üblicherweise verfügt man jedoch nicht über diese Informationen. Andere Verfahren beruhen darauf, dass benachbarte Linien oder Spalten einer Bildszene meist eine hohe Korrelation aufweisen. Dadurch konnte man beispielsweise durch Übertragungsprobleme verloren gegangene Bildzeilen von Weltraumsonden wiederherstellen. Allerdings muss man sich dabei darüber im Klaren sein, dass es sich hier nur um eine optische Aufbesserung handelt. Die verlorene Information kann natürlich nicht wiedererzeugt werden. Zudem sind diese Methoden für industrielle Anwendungen im Allgemeinen zu rechenaufwändig.

Erwähnen wollen wir hier aber noch den Min-Max- oder Kontrastfilter. Er weist jedem Punkt im Ergebnisbild die Differenz des kleinsten und größten Wertes innerhalb des Filteroperators im Originalbild zu. Der Effekt dieses Filters besteht darin, dass Variationen im Bild verstärkt und relativ einheitliche Bereiche weiter homogenisiert werden. Abbildung 2.29 zeigt diesen Effekt an unserem bekannten Beispielquadrat. Man sieht, dass zwar auch das Rauschen verstärkt und verdickt wird, aber bei weitem nicht so deutlich wie die Kanten des Quadrates. Gleichzeitig stellt man einen „Outlining-Effekt“ fest. Da abgesehen vom Rauschen das Innere des Quadrates selbst wieder homogen ist, bleibt ein nicht ausgefülltes Quadrat stehen.

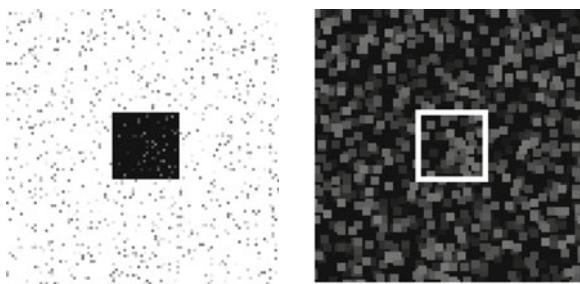


Abbildung 2.29: Wirkung eines 3×3 -Kontrastfilters auf flächige Gebilde

Ein besonderer Vorteil ist dabei in bestimmten Anwendungen, dass die Richtung der Variation gleichgültig ist. Das bedeutet, dass ein heller Fleck auf dunklem Grund und ein dunkler Fleck auf hellem Grund nach der Kontrastfilterung identisch aussehen. Diesen Effekt machen wir uns in Abschnitt 5.3 zu Nutze, um die unterschiedlichen Reflexionseigenschaften von Prägeziffern auszugleichen, deren einzelne Punkte nicht gleich tief in das Metall geschlagen wurden. Als Vorgriff sehen Sie in Abbildung 2.30 zwei Beispiele. Im einen Fall erscheinen die Ziffern im Originalbild hell, im anderen dunkel. Nach der Differenzfilterung sind die Kontrasteigenschaften nahezu identisch.

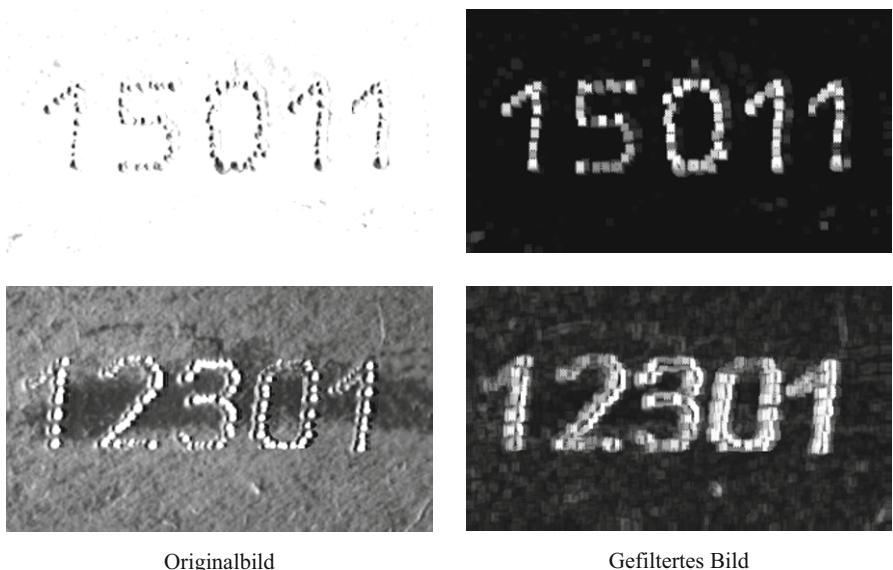


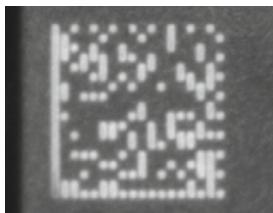
Abbildung 2.30: Geprägte Ziffern vor und nach Kontrastfilterung

2.7 Koordinatentransformationen

Bei den bisher vorgestellten Vorverarbeitungsmethoden wurden stets die Grauwerte der Bildpunkte zwischen Eingangs- und Ausgangsbild verändert, während die Anordnung der Bildpunkte zueinander unverändert blieb. Bei der Koordinatentransformation, die wir in diesem Abschnitt vorstellen wollen, ist es genau umgekehrt: während die Grauwerte der Bildpunkte unverändert bleiben, werden deren Koordinaten geändert. Die Bildpunkte befinden sich also im Ausgangsbild an einer anderen Stelle als im Eingangsbild. Die Koordinatentransformation kommt zum Einsatz, wenn ein Prüfobjekt unter einem ungünstigen Blickwinkel aufgenommen wird und vor der weiteren Analyse das Bild entzerrt werden muss, also die perspektivische Verzerrung herausgerechnet werden muss. Die Abbildung 2.31 zeigt einen, aus einem ungünstigen Blickwinkel aufgenommenen, DataMatrix-Code vor und nach der Entzerrung. Auch für diesen Teil der Vorverarbeitung gilt natürlich die Regel, dass man zunächst die Bildaufnahmesituation optimieren sollte, bevor man mit Vorverarbeitung nachträglich versucht die Fehler algorithmisch zu korrigieren.



Originalbild



Entzerrtes Bild

Abbildung 2.31: Koordinatentransformation bei einem DataMatrix-Code.

Mathematisch lässt sich eine Koordinatentransformation beschreiben mit

$$\begin{aligned}x_o &= f_x(x_i, y_i), \\y_o &= f_y(x_i, y_i).\end{aligned}\quad (2.23)$$

Dabei sind (x_i, y_i) die Koordinaten des Bildpunkts im Eingangsbild und (x_o, y_o) die Koordinaten im Ausgangsbild. Für die Grauwerte im Eingangsbild g_i und im Ausgangsbild g_o gilt die Beziehung:

$$g_o(x_o, y_o) = g_i(x_i, y_i) \quad (2.24)$$

Zur Berechnung des Grauwerts eines Ausgangsbildpunkts muss man die Koordinaten des zugehörigen Eingangsbildpunkts berechnen und dessen Grauwert übernehmen. Meistens wird aber der zu einem Ausgangsbildpunkt gehörige Eingangsbildpunkt nicht exakt auf einem Pixel liegen, sondern irgendwo zwischen vier Pixeln. Die Koordinaten des gesuchten Bildpunkts haben also meistens nicht ganzzahlige Werte. Für die Berechnung des Grauwerts an diesem Zwischenpunkt gibt es eine Reihe gängiger Interpolationsmethoden, wie z. B. Pixelwiederholung und bilineare Interpolation.

Pixelwiederholung: Bei der Pixelwiederholung wird der Grauwert des Bildpunkts verwendet, der den berechneten Koordinaten am nächsten liegt. Dabei kann es vorkommen, dass ein Eingangsbildpunkt mehrfach ins Ausgangsbild übernommen wird, daher der Name des Verfahrens. Die Pixelwiederholung ist das einfachste und damit schnellste Verfahren, führt allerdings zu einer schlechten Ergebnisbildqualität. Insbesondere bei schrägen Kanten im Bild sieht man die für dieses Verfahren charakteristischen Stufen.

Bilineare Interpolation: Bei der bilinearen Interpolation wird der Ausgangswert aus den Grauwerten der vier Eingangsbildpunkte interpoliert, die um den theoretischen Eingangsbildpunkt liegen. Dabei findet sowohl in x- als auch in y-Richtung eine lineare Interpolation statt, daher der Name bilinear. Wenn der rechnerische Punkt (x, y) zwischen den vier Bildpunkten $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ und (x_4, y_4) mit den Grauwerten g_1, g_2, g_3 und g_4 liegt, ergibt sich der gesuchte Grauwert als der

gewichtete Mittelwert der Grauwerte der vier umgebenden Punkte gemäß folgender Gleichung:

$$g = \sum_{i=1}^4 (1 - |x_i - x|) (1 - |y_i - y|) g_i \quad (2.25)$$

Da der Abstand des Pixelgitters eins ist und der rechnerische Punkt zwischen den Punkten auf dem Pixelgitter liegt, sind die Beträge immer zwischen null und eins. Das Gewicht, mit dem der Grauwert eines Bildpunkts in die Summe eingeht, ist also umso größer je näher der rechnerische Punkt an dem Ausgangsbildpunkt liegt.

Um die Koordinatentransformation ausführen zu können fehlen noch die Funktionen f_x und f_y aus Gleichung 2.23. Laut (Jähne 2005) kann man perspektivische Verzerrungen beschreiben mit

$$\begin{aligned} f_x(x, y) &= \frac{ax + by + c}{gx + hy + 1}, \\ f_y(x, y) &= \frac{dx + ey + f}{gx + hy + 1}. \end{aligned} \quad (2.26)$$

Um das Bild zu entzerren, muss man die Koeffizienten der Gleichung 2.26 bestimmen und die Umkehrfunktion berechnen. Die Koeffizienten werden bestimmt, in dem man von mindestens vier Punkten im Eingangsbild die Koordinaten bestimmt und die zugehörigen Ausgangsbildkoordinaten festgelegt. Aus diesen Koordinatenpaaren berechnet man die Koeffizienten, entweder bei vier Punkten exakt oder bei mehr Koordinatenpaaren durch Methoden der Ausgleichsrechnung. Ein Ausführliche Darstellung der Berechnung der Koeffizienten findet sich in (Jähne 2005).

Es gibt eine Reihe von Erweiterungen und Alternativen zu dem beschriebenen Verfahren. Eine häufige Erweiterung des Ausgleichs der perspektivischen Verzerrung ist die Korrektur von Objektivverzeichnungen. In der Literatur werden diese Methoden häufig zusammen mit einer Umrechnung von Bild- in Weltkoordinaten als Kamerakalibrierung behandelt, siehe (Steger et al. 2008). Neben der Entzerrung mit festen Funktionen wie oben dargestellt kann man auch zweidimensionale Spline-Funktionen verwenden, die zwar mehr Stützpunkte benötigen, also Koordinatenpaare im Eingangs- und Ausgangsbild, aber dafür auch vielfältigere Verzerrungen korrigieren können.

2.8 Integraltransformationen

Zum Abschluss dieses Kapitels über Bildvorverarbeitung wollen wir noch darauf hinweisen, dass es eine Reihe globaler Bildtransformationen auf der Basis von Integraltransformationen gibt, auf deren Darstellung wir hier aus verschiedenen Gründen verzichten. Dazu gehören vor allem die Fouriertransformation und die Familie der Wavelettransformationen. Beide Methoden können zur Charakterisierung von Texturen, d. h. Oberflächenstrukturen, eingesetzt werden und zur verschiebungs- und rotationsinvarianten Beschreibung von Objekten.

Wir besprechen diese Verfahren hier aus folgenden Gründen nicht:

- Sie sind sehr rechenaufwändig, da sie eine Vielzahl von Operationen mit dem gesamten Bild durchführen müssen.
- Die für das Verständnis der Fouriertransformation und die Interpretation ihrer Ergebnisse erforderliche Mathematik geht weit über den Umfang dieses Einführungsbuches hinaus.
- Textur-Analysen dieser Art sind keine typischen Aufgaben der industriellen Bildverarbeitung.
- Objektmerkmale auf der Basis von Fourier- und Wavelettransformation werden meist im Bereich der Handschrifterkennung eingesetzt. In industriellen Anwendungen kann man Verschiebungen und Verdrehungen meist mit weniger aufwändigen Methoden kompensieren.

Der interessierte Leser findet diese Themen in vielen Büchern ausführlich behandelt, beispielsweise in (Jähne 2005), (Tönnies 2005), (Nischwitz 2007), (Gonzalez und Woods 2008), (Russ 2008) und anderen.

2.9 Zusammenfassung

Bildvorverarbeitung: Bei der Bildvorverarbeitung wird aus einem Eingangsbild ein neues Bild erzeugt, das dem Eingangsbild im Allgemeinen ähnlich ist, sich aber in bestimmten Eigenschaften von ihm unterscheidet, z. B. einen besseren Kontrast oder geringeres Rauschen aufweist.

Punktoperationen: Diese Vorverarbeitungsverfahren verwenden nur den Bildpunkt an der jeweiligen Stelle des Eingangsbildes (oder der Eingangsbilder). Bei inhomogenen Punktoperationen hängt die Rechenvorschrift vom Ort des Bildpunkts ab, bei homogenen nicht. Die wichtigsten Punktoperationen sind die Grauwerttransformationen und die Funktionen der Bildarithmetik.

Grauwerttransformation: Dies ist eine homogene Punktoperation, die den Grauwert i_{ij} jedes einzelnen Bildpunkts mittels einer Abbildungsfunktion $f(i_{ij})$ in

einen neuen Grauwert o_{ij} umrechnet. Die Funktion selbst ist nicht vom Ort des Bildpunkts abhängig.

Look-Up-Tabellen: Grauwerttransformationen werden häufig in Form von Look-Up-Tabellen implementiert, in denen für jeden Eingangsgrauwert direkt der Ergebniswert nachgesehen werden kann. Dadurch lassen sich viele Transformationen beschleunigen, z. B. logarithmische Funktionen. Auch analytisch nicht darstellbare Transformationen lassen sich so beschreiben.

Lineare Grauwertskalierung: Bei dieser Transformation wird der Grauwert nach einer Gleichung der Form $o_{ij} = c_2 \cdot (c_1 + i_{ij})$ transformiert.

Kontrastnormierung: Durch entsprechende Wahl der Koeffizienten einer linearen Grauwertskalierung kann man erreichen, dass das Bild den Grauwertbereich von [0, 255] vollständig ausnutzt. Wenn das Bild in Subbilder eingeteilt wird, spricht man von einer *lokalen Kontrastnormierung*. Dieses Verfahren ist zwar streng genommen keine homogene Punktoperation, lässt sich aber in ähnlicher Weise implementieren. Damit können Beleuchtungsgradienten ausgeglichen werden.

Histogramm-Ebnung: Diese Grauwerttransformation sorgt dafür, dass jedes Helligkeitsintervall gleichmäßig besetzt ist. Sie dient vor allem der Verbesserung der Erkennbarkeit von Objekten durch den Menschen.

Bildarithmetik: Hierunter versteht man die Verrechnung korrespondierender Bildpunkte zweier Bilder. Die wichtigsten Verfahren sind:

- Addition und Mittelung, z. B. zur Rauschunterdrückung bei dunklen Bildzeilen
- Subtraktion, z. B. für Druckqualitätskontrollen und Shading-Korrektur
- Minimum und Maximum, z. B. zur Zusammensetzung der Ergebnisbilder von Kantenfiltern
- Division, z. B. zur Shading-Korrektur

Bei den meisten bildarithmetischen Verfahren können Werte außerhalb des Grauwertintervalls [0, 255] entstehen, die auf verschiedene Weise (Clipping, lineare Skalierung von positiven, negativen oder absoluten Werten) wieder in dieses Intervall abgebildet werden können.

Lokale Operationen: Diese Verfahren benutzen eine Umgebung N^* des aktuellen Punkts zur Berechnung des Ergebniswertes. Man spricht hier häufig auch von Bildfiltern und bezeichnet die Umgebung N^* als *Operator*, *Filterkern* oder *Filtermaske*.

Lineare Filter: Dieser Typ der lokalen Operationen errechnet den Ergebnisgrauwert durch eine gewichtete Summierung aller Grauwerte in der Umgebung N^* des aktuellen Punkts. Hierzu gehören z. B. die Glättungsfilter und die Kantenfilter. Allgemein wird die Filterwirkung mit zunehmender Größe des Operators stärker,

allerdings steigt auch der Rechenaufwand linear mit der Anzahl der Punkte in der Umgebung.

Glättungsfilter: Diese Filter dienen zur Rauschunterdrückung. Der einfachste Fall ist der *Rechteckfilter*, oft auch als *Mittelwertfilter* bezeichnet. Der *Gaußfilter* bietet bessere Rauschunterdrückung bei geringerer Kantenunschärfe.

Kantenfilter: Diese Filter arbeiten nach dem Prinzip der numerischen Differenzierung. Sie heben Unterschiede im Bild hervor, meist in einer bestimmten Richtung, z. B. waagerechte oder senkrechte Kanten. Bekannte Kantenfilter sind die *Sobelfilter* und die *Laplacefilter*, es gibt jedoch noch viele weitere.

Medianfilter: Dieser Filter verwendet im Gegensatz zum einfachen Glättungsfilter nicht den Mittelwert, sondern den weniger störempfindlichen Median der Grauwerte innerhalb der Umgebung N^* . Es handelt sich daher um einen nichtlinearen Filter. Er eignet sich gut zur Unterdrückung von Pixelrauschen bei Erhaltung der Kantenschärfe.

Morphologische Filter: Diese Filter beeinflussen die Form von Bildstrukturen. Sie verwenden hauptsächlich Minimum- und Maximumoperationen, sind also nichtlineare Filter. Die bekanntesten sind *Erosion*, *Dilatation*, *Opening* und *Closing*.

Kontrastfilter: Dieser nichtlineare Filter ersetzt den Grauwert jedes Punkts durch den maximalen Kontrast in der Umgebung N^* . Damit lassen sich helle und dunkle Variationen im Bild gleichermaßen hervorheben.

Koordinatentransformationen: Koordinatentransformationen verändern die Anordnung der Bildpunkte im Bild. Damit kann man z. B. durch ungünstige Kameraspektiven verzerrte Bilder korrigieren.

Globale Operationen: Diese Vorverarbeitungsalgorithmen errechnen das Ergebnisbild aus dem gesamten Eingangsbild. Bekannte Beispiele sind Integraltransformationen wie die Fourier- oder die Wavelettransformation. Sie ermöglichen komplexe Verarbeitungsvorgänge, sind aber sehr rechenaufwändig.

3. Lageerkennung

Es mag etwas ungewöhnlich erscheinen, die ausführliche Besprechung der verschiedenen Anwendungsbereiche industrieller Bildverarbeitung gerade mit der Lageerkennung zu beginnen, statt mit der Objekterkennung, da man ja ein Objekt zunächst finden muss, um seine Lage feststellen zu können. Objekterkennung ist aber ein recht weiter Begriff und erfordert häufig eine Vielzahl von Hilfsfunktionen, um ein Objekt schließlich einer Objektklasse zuordnen zu können. Lageerkennung hingegen ist strukturell – nicht unbedingt algorithmisch – eine vergleichsweise einfache Angelegenheit, sobald das fragliche Objekt einmal gefunden ist. Die einzige notwendige Voraussetzung ist also die Segmentierung eines Referenzobjekts. Der Hauptgrund dafür, dieses Gebiet an den Anfang zu stellen, ist jedoch, dass es in der industriellen Bildverarbeitung gewissermaßen eine unerlässliche „Hilfswissenschaft“ darstellt.

Die Beispielanwendung im Abschnitt 1.6 war nur deshalb so leicht zu lösen, weil genau bekannt war, wo sich das zu prüfende Objekt befand. Insbesondere die Vermessung und noch mehr die gezeigte Methode zur Anwesenheitskontrolle sind darauf angewiesen, ihre Untersuchungsobjekte mit sehr geringen Toleranzen immer an derselben Stelle vorzufinden, während man die Typnummer noch mit einer – zeitraubenden – Suche im gesamten Bild hätte finden können. Man kann nun zwar in der industriellen Fertigung zumeist davon ausgehen, dass Prüfteile innerhalb gewisser Grenzen positioniert sind, aber in der Regel sind diese Grenzen nicht eng genug für die Anforderungen der Bildverarbeitung. In den allermeisten Fällen wird man daher nicht umhin kommen, zunächst die Lage von Referenzobjekten festzustellen, die leicht zu finden sind – und sich mit Sicherheit im Blickfeld der Kamera befinden, und erst danach mit der eigentlichen Prüfung beginnen.

3.1 Position eines Einzelobjekts

Kehren wir noch einmal zu dem Beispiel in Abschnitt 1.6 zurück. Selbst wenn die Zündkerze auf einem Werkstückträger befestigt ist, der sie vor der Kamera vorbeiführt, wird man in der Praxis immer mit kleinen Positionsschwankungen rechnen müssen. Diese fallen natürlich besonders unangenehm auf, wenn für die Objektsuche nur so kleine Toleranzbereiche verfügbar sind, wie hier in den Teilaufgaben Anwesenheitskontrolle und Vermessung.

Abbildung 3.1 zeigt die Wirkung einer solchen Positionsabweichung auf die Suchlinien für die Vermessung der Gewindetiefe aus Abschnitt 1.6, hier im Auflicht dargestellt. Die beiden Suchlinien treffen das Gewinde nicht mehr in der Mit-

te, die Bereichsgrenzen liegen entsprechend falsch, so dass die gefundene Kontur nichts mit der eigentlich zu vermessenden Gewindekontur zu tun hätte.



Abbildung 3.1: Suchlinien für Gewindekontur auf horizontal verschobener Zündkerze

Programmhinweis: Sie können eine solche Verschiebung in den im Download enthaltenen Beispielprogrammen leicht simulieren, indem Sie im Prüfschritt *Bild in Speicher übertragen* den definierten Bildausschnitt verschieben.

Es ist offensichtlich, wie das Problem zu beheben ist: Man muss zunächst feststellen, an welcher Position sich das Gewinde eigentlich im Bild befindet und dann die Suchlinien entsprechend verschieben. Dabei sind verschiedene Strategien denkbar, die in den folgenden Abschnitten besprochen werden.

3.1.1 Positionierung anhand des Gesamtobjekts

Der einfachste Weg ist, die gesamte Zündkerze als Objekt zu segmentieren und ihre Position festzustellen. Bei der ersten Durchführung dieser Prozedur wird die festgestellte Position als Referenz gespeichert, und die Suchlinien werden gemäß dieser Referenzposition definiert. Bei allen folgenden Durchläufen wird dann die Abweichung der derzeitigen Position zu dieser gespeicherten Referenz festgestellt und die Suchlinien entsprechend dieser Abweichung verschoben. Dabei genügt es in diesem speziellen Fall natürlich, die Abweichung in horizontaler Richtung festzustellen.

An der Lösung der eigentlichen Prüfaufgabe ändert sich im Vergleich zu Abschnitt 1.6 überhaupt nichts, lediglich der zweite Schritt, das Setzen der Arbeitsbereiche, ändert sich, da deren Positionen nur relativ zur Position der gesamten Zündkerze festliegen. Der gesamte Ablauf wird im folgenden ausführlich erklärt.

1. *Bildaufnahme*
2. *Arbeitsbereich für Suche der Referenzposition setzen:* Die Größe und Position des Suchbereichs für die Referenzposition muss sich nach der Art des Referenzobjekts und der zu erwartenden Positionsungenauigkeit richten.
Die Position variiert in diesem Beispiel nur wenig in horizontaler Richtung, es genügt also beispielsweise, einen gewissen Bereich links im Bild abzusuchen, um den Übergang zwischen Hintergrund und Objekt zu finden, wie es Abbildung 3.2 zeigt.

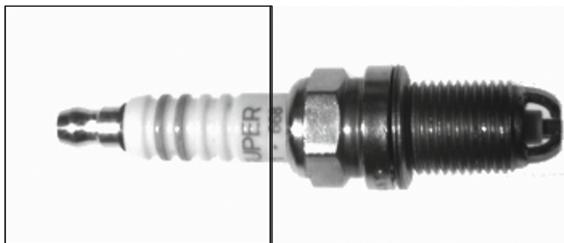


Abbildung 3.2: Suchbereich zur Erzeugung des Referenzobjekts

3. *Referenzobjekt erzeugen:* Nachdem der Übergang zwischen Hintergrund und Referenzobjekt gefunden wurde, lässt sich mit der in Kapitel 4 beschriebenen Methode der Konturverfolgung die gesamte Zündkerze als Objekt erzeugen, wie es Abbildung 3.3 zeigt.



Abbildung 3.3: Gesamte Zündkerze als Referenzobjekt

4. *Objekteigenschaften berechnen:* Die einzige wesentliche Eigenschaft dieses Referenzobjekts ist seine Position in vertikaler Richtung. In diesem Schritt werden daher die Koordinaten des Objektschwerpunkts berechnet.
5. *Arbeitsbereich für Prüfung setzen:* Mit der Feststellung der Referenzposition sind alle erforderlichen Informationen vorhanden, um die endgültigen Suchbereiche anzulegen, hier beispielsweise 173 Bildpunkte rechts des Objektschwerpunkts. Abbildung 3.4 zeigt die relativ zu der Objektposition von Abbildung 3.1 entsprechend positionierten Suchlinien.

Alles weitere läuft dann genauso ab wie bei dem Beispiel in Abschnitt 1.6, weshalb wir hier nur die Abfolge der Schritte angeben.

6. *Objekte erzeugen:* Auf den Suchlinien, die das Gewinde jetzt wieder mittig schneiden, werden die Begrenzungen des Gewindes gesucht und die Konturlinien innerhalb der Begrenzungen erzeugt.
7. *Objekteigenschaften berechnen:* Der minimale und maximale Abstand der Konturlinien in senkrechter Richtung wird ermittelt.
8. *Entscheidung:* Wenn beide Abstände innerhalb der vorgeschriebenen Toleranzen liegen, gilt das Gewinde als einwandfrei.



Abbildung 3.4: Relativ zur Zündkerze korrekt positionierte Suchlinien

Dieser Ablauf entspricht immer noch dem Vorgehensmodell aus Abschnitt 1.5. Hinzu kommt allerdings ein fast vollständiger zweiter Ablauf mit dem die Referenzposition bestimmt wird, anhand der die Arbeitsbereiche für die eigentliche Prüfung positioniert werden.

Programmhinweis: Wenn man ein Prüfprogramm für den Einsatz in der Fertigung konfiguriert – oder allgemein: Wann immer man eine Prüfung auf der Basis echter Kamerabilder realisiert, sollte man es sich zur Regel machen, (mindestens) ein Kamerabild als Referenz abzuspeichern. Dieses Bild sollte man auch zur Konfiguration einer Positionskorrektur verwenden, damit man später jederzeit die Positionsreferenz zur Verfügung hat. **NeuroCheck** bietet speziell zu diesem Zweck die Möglichkeit, so genannte Einrichtbilder mit Markierungen zu versehen und mit dem aktuellen Kamerabild zu vergleichen, um Fehlpositionierungen leicht feststellen zu können.

3.1.2 Positionierung anhand einer Kante

Die oben angewandte Methode, den Schwerpunkt des Gesamtobjekts als Referenzposition zu benutzen, besticht durch ihre Einfachheit und ist in der Regel auch wegen ihrer Stabilität vorzuziehen. Da bei der Berechnung des Schwerpunkts eine Mittelung über alle Punkte des Objekts stattfindet, ist die Position gegenüber kleinen Veränderungen der Beleuchtung oder gewissen Unschärfen weitgehend unempfindlich. Gegen diese Methode spricht, dass es zu Verfälschungen der Berechnung kommen kann, wenn sich die Form des Objekts durch die Aufnahmebedingungen ändert.

Formänderung aufgrund von Segmentierungsproblemen: Die Schrifterkennung erfordert Auflichtbeleuchtung, damit die zu lesenden Zeichen sichtbar sind. In diesem Fall besteht aufgrund des hellen Keramikkörpers und des glänzenden Gewindes die Gefahr, dass die Segmentierung teilweise fehlschlägt. Dadurch verändert sich die Form des Referenzobjekts und aufgrund dessen die Schwerpunkt-lage. Abbildung 3.5 zeigt den Extremfall, dass der gesamte hintere Objektbereich nicht mehr segmentiert wird, wodurch sich der Schwerpunkt von der korrekten Position null an die Position eins verschiebt.

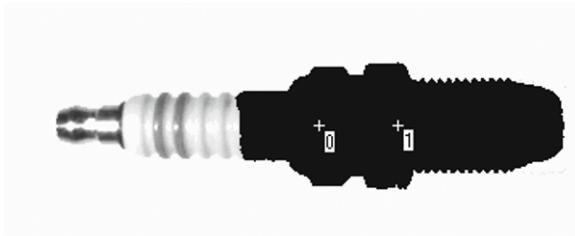


Abbildung 3.5: Pseudoverschiebung des Schwerpunkts aufgrund von Segmentierungsproblemen

Derart krasse Fehler wie in Abbildung 3.5 sind natürlich selten und leicht zu erkennen, aber subtilere Abweichungen können selbst im Durchlicht auftreten. Ein zu starker Reflex am Keramikkörper kann leicht, wie in Abbildung 3.6 gezeigt, zu einem teilweisen Verschmelzen mit dem Hintergrund führen, was bei entsprechend empfindlichen Prüfungen ebenfalls Fehler hervorruft.



Abbildung 3.6: Teilweise Hintergrund-Verschmelzung durch Überstrahlung an der Kontur

Formänderung aufgrund zu großer Verschiebung: Wenn das Objekt so stark verschoben ist, dass es sich teilweise nicht mehr im Bild befindet, wird die – in diesem Fall tatsächlich vorhandene – Verschiebung aufgrund der Änderung der Objektform völlig falsch berechnet, wie es Abbildung 3.7 zeigt.



Abbildung 3.7: Pseudoverschiebung des Schwerpunkts aufgrund einer Objektverschiebung über den Bildrand hinaus

In einem solchen Fall ist es empfehlenswert, nicht anhand des Schwerpunkts zu positionieren, sondern anhand der Position einer Objektkante. Man benötigt dann

eine vertikale Kante als x-Referenz und eine horizontale Kante als y-Referenz. Eine solche Kante muss hinreichend breit sein, um trotz Positionstoleranzen in der jeweils anderen Richtung sicher gefunden zu werden. Das ist bei der Zündkerze, soweit es die x-Referenz betrifft, nicht ganz einfach, da sie kaum entsprechende vertikale Kanten aufweist. Daher würde man hier zunächst eine horizontale Kante als y-Referenz suchen, beispielsweise den oberen Rand im Bereich des Sechskants. Wenn man die detektierte Kante breit genug wählt, ist sicher, dass immer die oberste Begrenzung der Zündkerze enthalten ist, man erhält also eine genaue y-Referenz. Abbildung 3.8 zeigt den entsprechenden Suchbereich oben und das detektierte Konturstück unten.



Abbildung 3.8: Suchlinie und detektierte obere Kontur (schwarz hervorgehoben)

Anhand dieser vertikalen Position kann man dann neue Suchlinien positionieren, mit denen sich die kurzen vertikalen Kanten des Sechskants als x-Referenz finden lassen. Eine solche hierarchische Vorgehensweise kommt bei positionskritischen Prüfungen häufig vor. Die beiden gefundenen Positionsdaten in x- und y-Richtung liefern in den beiden oben genannten Fällen im Gegensatz zur Schwerpunktsmethode einwandfreie Ergebnisse – und benötigen darüber hinaus auch weniger Rechenzeit.

Fehlerquellen bei der Kantenpositionierung: Die Positionierung auf eine Kante ist jedoch kein Allheilmittel. Die möglichen Probleme werden im Folgenden kurz beschrieben.

Empfindlichkeit: Bei der Positionierung auf ein Kantenstück werden wesentlich weniger Punkte verwendet als bei der Schwerpunktsberechnung. Daher ist, außer bei so drastischen Formänderungen wie in Abbildung 3.5, die Schwerpunktsberechnung weniger empfindlich. Wenn z. B. durch eine leichte Überbelichtung das ge-

samte Objekt zu schrumpfen scheint, weil die Sensorelemente entlang der Objektkante zu viel Licht erhalten, wird sich der Schwerpunkt durch die Mittelung über das gesamte Objekt kaum verschieben, während die Kanten entsprechend der scheinbaren Schrumpfung des Objekts wandern.

Kantenqualität: Die zur Positionsbestimmung verwendeten Kanten sollten gerade, glatt, möglichst lang und stets im Bildfeld sein, damit man sie auch zuverlässig finden kann. Die Positionierung auf das sehr kleine Kantenstück in Bild 3.8 war nur deshalb möglich, weil das Objekt in vertikaler Richtung fixiert war, man sich also darauf verlassen konnte, dass die Suchlinie den Sechskant an dem betreffenden Punkt schneiden würde.

Wenn entlang der Kante Störungen auftreten, kann unter Umständen eine Filterung helfen, die eine Glättung der Kante senkrecht zur Suchrichtung vornimmt. Dann lassen sich auch so ungleichmäßige Konturen wie ein Gewinde zur Positionierung verwenden. Je nachdem, ob die Suchlinie das Gewinde unserer Beispielzündkerze unten in einem Gewindegang oder oben trifft, ergibt sich ein Unterschied in der Y-Koordinate von mehreren Bildpunkten. Wenn man jedoch das Gewinde über eine gewisse Breite mit einem Glättungsfilter bearbeitet, kann man diese Unterschiede herausmitteln, wie es Abbildung 3.9 zeigt. Auch wenn die Suchlinie jetzt innerhalb des geglätteten Bereiches wandert, wird sie die Kante immer an derselben Y-Koordinate finden.



Abbildung 3.9: Glättung einer Gewindekontur zur Erzeugung einer glatten Positionierungskante

Programmhinweis: **NeuroCheck** enthält eine spezielle Filterfunktion, *Arbeitsbereiche glätten*, die speziell für die Glättung von Positionierungskanten eingeführt wurde.

Verdrehung: Das größte Problem bei der Positionierung auf Objektkanten stellen jedoch Verdrehungen des Objekts dar. Selbst wenn das Objekt sich nur um seinen Schwerpunkt verdreht, verschieben sich seine Kanten ganz erheblich, so dass beträchtliche Pseudoverschiebungen auftreten. Wenn darüber hinaus auf ganz bestimmte Konturstücke kompliziert geformter Objekte positioniert werden muss,

können die betreffenden Punkte unter Umständen gar nicht mehr gefunden werden. Eine Verdrehung der Zündkerze um fünf Grad führt z. B. zu einer Schwerpunktsverschiebung von 0,07 Bildpunkten (aufgrund der Ortsdiskretisierung bei der Bildaufnahme verrutschen natürlich einzelne Konturpunkte), die geglättete Gewindekante aus Abbildung 3.9 verändert ihre Position jedoch um volle 20 Bildpunkte und wird damit für die Positionierung völlig unbrauchbar.

Wie das Problem der Kantenqualität lässt sich auch die Verdrehung mit etwas erhöhtem Aufwand durchaus bewältigen, indem auch die Orientierung des Objekts festgestellt und zur Positions korrektur herangezogen wird. Rotationskorrekturen werden im folgenden Abschnitt besprochen.

3.2 Ausrichtung eines einzelnen Objekts

Die Orientierung eines Objekts zu messen, ist in der Regel wesentlich schwieriger als seine Position festzustellen. Besondere Probleme bereiten natürlich symmetrische Objekte, bei denen eine eindeutige Bestimmung unter Umständen gar nicht möglich ist, aber auch Objekte, deren Ausrichtung dem Menschen völlig klar erscheint, können gelegentlich überraschende Probleme aufwerfen, v. a. wenn man Kompromisse zwischen der Genauigkeit der Rotationsbestimmung und dem Rechenaufwand finden muss. Der Schwerpunkt einer Kontur ist noch verhältnismäßig schnell und mit hoher Genauigkeit durch Mittelung der Koordinaten aller Konturpunkte zu bestimmen, die Hauptträgheitsachse eines Objekts erfordert dagegen einen höheren Aufwand, wie wir gleich noch sehen werden.

3.2.1 Orientierungsberechnung anhand der Hauptachse

Im Folgenden werden wir einige theoretische Grundlagen für die Berechnung der Hauptträgheitsachse, sowohl exakt als auch näherungsweise, darlegen. Moderne Bildverarbeitungssysteme bieten zwar in der Regel diese Funktionalität bereits fertig eingebaut an, aber es lauern dort einige Fallen, die sich leichter verstehen und umgehen lassen, wenn man ein Gefühl für die zugrundeliegenden Verfahren entwickelt.

Theoretische Berechnung der Hauptträgheitsachse: Jedes Objekt hat eine Hauptachse, d. h. eine Linie, für die es sein geringstes Trägheitsmoment besitzt, um die es folglich stabil rotieren kann. Den Winkel dieser Achse bezogen auf die Horizontale kann man nach (Russ 2007) folgendermaßen berechnen:

$$\theta = \tan^{-1} \left\{ \frac{M_{xx} - M_{yy} + \sqrt{(M_{xx} - M_{yy})^2 + 4M_{xy}^2}}{2M_{xy}} \right\}. \quad (3.1)$$

worin die M_{ij} die Elemente des Trägheitstensors darstellen, die mit A als Gesamtfläche des Objekts durch Summation über alle Punkte i des Objekts folgendermaßen berechnet werden können:

$$M_{xx} = \sum_i x_i^2 - \frac{\left(\sum_i x_i\right)^2}{A}, \quad (3.2)$$

$$M_{yy} = \sum_i y_i^2 - \frac{\left(\sum_i y_i\right)^2}{A}, \quad (3.3)$$

$$M_{xy} = \sum_i x_i y_i - \frac{\sum_i x_i \sum_i y_i}{A}. \quad (3.4)$$

Für jeden Punkt des Objekts sind also drei Multiplikationen und fünf Additionen auszuführen; bei dem Beispielobjekt Zündkerze, das eine Fläche von gut 60 000 Pixeln aufweist, sind dies immerhin über 180 000 Multiplikationen und 300 000 Additionen. Prinzipiell ist es natürlich möglich, die Anzahl der erforderlichen Operationen deutlich zu verringern, indem man statt der Gesamtfläche nur die Konturpunkte benutzt. Dabei tritt allerdings ein Fehler auf, der je nach Lage des Objekts nicht gleich stark ausgeprägt ist, da komplexe Teile der Kontur mit vielen Ein- und Ausbuchtungen durch die Vielzahl von Punkten überbewertet werden. Außerdem führt die Beschränkung auf die Kontur zu einer größeren Empfindlichkeit der Achsenberechnung gegenüber Formänderungen des Objekts, wie etwa in Abbildung 3.5.

Näherungsverfahren zur Hauptachsenberechnung: Wegen des hohen Rechenaufwandes für die mathematisch exakte Bestimmung der Hauptachse verwendet man in der Regel Näherungsverfahren. Ein mögliches Näherungsverfahren ist in (Parker 1994) angegeben. Es geht davon aus, dass die Hauptachse immer durch den Schwerpunkt und auch durch einen Konturpunkt verläuft. Damit ist die Anzahl der Kandidatenlinien bereits erheblich reduziert, so dass entsprechend weniger Berechnungen erforderlich sind. Andere Verfahren verwenden umschreibende Polygone zur Bestimmung der Achse. Damit lässt sich eine weitere Beschleunigung erreichen, allerdings muss eine Reihe von Sonderfällen beachtet werden, z. B. der Fall, dass das Polygon zu einem Rechteck degeneriert, wodurch Unstetigkeiten auftreten können.

Probleme bei Hauptachsenverfahren: Die Richtungsbestimmung anhand der Hauptachse wirft in jedem Fall gewisse Probleme auf, ob man nun den Rechenaufwand für die mathematisch exakte Formulierung treibt oder ein Näherungsverfahren anwendet.

Orientierung: Mit dem Hauptachsenverfahren lässt sich die Orientierung des Objekts berechnen, während die *Richtung* des Objekts, d. h. wo vorne und hinten ist,

nicht berechnet werden kann. Um die Richtung eines Objekts zu bestimmen, kann man auf eine Hilfskonstruktion zurückgreifen, die die Lage des Objektschwerpunkts gegenüber der Achsenmitte berücksichtigt. Bei *achsensymmetrischen* Objekten können aber trotzdem schon kleine Konturänderungen ein Kippen der berechneten Richtung um 180° verursachen.

Punktsymmetrie: Bei Objekten, die in ihrer Gestalt einem Kreis oder Quadrat ähneln, sind die beiden Hauptträgheitsmomente so wenig verschieden, dass anstelle des noch relativ leicht beherrschbaren Kippens der Hauptachse um 180° eine Vertauschung der Haupt- und Nebenachse eintritt. Dadurch kippt die festgestellte Achse um 90° . Je kleiner das Objekt ist, desto stärker wirken sich dabei Änderungen seiner Kontur aus. Abbildung 3.10 illustriert dies anhand eines einfachen Objekts, dessen umschreibendes Rechteck genau quadratisch ist. Bei dem links gezeigten Originalobjekt führt die (mathematisch exakte) Berechnung der Hauptachse auf den Winkel null Grad entsprechend der eingezeichneten Achse. Das Hinzufügen eines einzigen Bildpunkts an der linken oberen Ecke dreht die Achse auf 84° bei Heranziehung der gesamten Fläche zur Achsenberechnung, $80,9^\circ$ bei Beschränkung auf die Kontur. Bei Verwendung der oben beschriebenen Näherungsrechnung tritt diese Unsicherheit grundsätzlich bei Objekten mit annähernd quadratischem Höhen/Breitenverhältnis auf, da in diesem Fall das umschreibende Polygon zu einem Quadrat degeneriert.

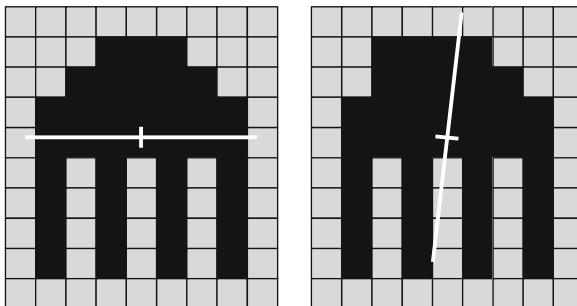


Abbildung 3.10: Effekt einer kleinen Konturänderung auf die Achse bei annähernd quadratischen Objekten

Anwendung der Rotationskorrektur: Nun haben wir so viel Negatives über die Hauptachsenberechnung gesagt, dass Sie sich vielleicht schon die Frage stellen, wozu man das Verfahren überhaupt verwenden kann? Die Antwort ist, dass es für viele Objekte brauchbare Ergebnisse liefert – nach den obigen Ausführungen sogar überraschend gute, wenn man einige Grundregeln beachtet:

- Das Referenzobjekt sollte lang gestreckt sein, damit die Verwechslung von Haupt- und Nebenachse nicht auftreten kann.
- Das Referenzobjekt sollte relativ zu seiner Nebenachse klar asymmetrisch sein, wenn neben der Richtung auch die Orientierung des Objekts, d. h. die Unterscheidung zwischen vorn und hinten, bestimmt werden soll. Sehr gute Ergebnisse werden z. B. bei Personenköpfen erreicht.

nisse liefern z. B. lang gestreckte Dreiecksformen, bei denen sich selbst mit dem Näherungsverfahren Genauigkeiten besser als $0,5^\circ$ erreichen lassen.

- Wenn es nur um die Richtung geht, eignen sich gerade Kanten ganz ausgezeichnet, da man in diesem Fall einfach eine Gerade zwischen Anfangs- und Endpunkt der Kante ziehen und den Winkel direkt bestimmen kann. Wenn man beispielsweise davon ausgeht, dass bei der Zündkerze nur leichte Verkippungen im Bereich von $\pm 10^\circ$ auftreten, dann kann man die geglättete Gewindekante aus Abbildung 3.9 zur Bestimmung der Rotation benutzen. Auch damit lässt sich eine Genauigkeit besser als $0,5^\circ$ erreichen.

Die Zündkerze als Ganzes eignet sich trotz ihrer lang gestreckten Form nicht so gut als Referenzobjekt für das Näherungsverfahren, da sie annähernd ellipsoid ist. Hier muss man mit Fehlern im Bereich von $1,5^\circ$ rechnen. Auch das würde aber durchaus noch für eine Repositionierung der Suchlinien für die Vermessungsaufgabe ausreichen, wie Abbildung 3.11 zeigt. Da die Gewindekontur nach der Positionierung noch separat gesucht und als Objekt erzeugt wird, erzeugt die kleine Winkelabweichung keinen nennenswerten Fehler. Natürlich muss man daran denken, auch die Messrichtung für den Abstand um den gemessenen Rotationswinkel nachzuführen.



Abbildung 3.11: Mit Hauptachsennäherung positionierte Suchlinien

3.2.2 Polarabstandsprojektion

Wir haben im vergangenen Abschnitt gesehen, dass es mehr oder weniger schnelle und stabile Verfahren gibt, die Hauptträgheitsachse eines Objekts zu bestimmen, um diese Richtung zum Ausgleich von Verdrehungen zu benutzen. Was kann man nun bei den Objekten tun, die sich notorisch schlecht für Korrekturen anhand der Hauptachsenrichtung eignen, selbst wenn die Hauptachse mathematisch exakt berechnet wird? Eine Möglichkeit dazu bietet die *Polarabstandsprojektion*.

Definition der Polarabstandsprojektion: Die Polarabstandsprojektion¹ ist in (Gonzalez und Woods 2008) definiert als eindimensionale Repräsentation des

¹ Englisch: distance-versus-angle signature

zweidimensionalen Gebildes *Kontur*, basierend auf dem Abstand der Konturpunkte vom Schwerpunkt des Objekts. Die Polarabstandsprojektion ist demnach ein Vektor, in dem jedes Element den Abstand eines Konturpunkts vom Schwerpunkt in einer bestimmten Richtung enthält. Die Anzahl der Elemente ist nicht festgelegt, es ist z. B. naheliegend, die Abstandsmessung in Schritten von einem Grad vorzunehmen, womit man eine Projektion aus 360 Elementen erhält.

Die Polarabstandsprojektion eines exakten Kreises ist ein Vektor aus lauter gleichen Elementen, jedes so groß wie der Radius des Kreises. Abbildung 3.12 zeigt Polarabstände für einige einfache Figuren. Die Berechnung der Polarabstandsprojektion selbst ist prinzipiell einfach, bei der tatsächlichen Umsetzung treten allerdings, wie immer bei der Verarbeitung realer Objekte, Schwierigkeiten auf, die zu lösen nicht immer trivial ist. So kann ein Strahl vom Schwerpunkt zur Kontur die Kontur durchaus mehrfach schneiden, wenn es sich nicht um sehr einfache geometrische Formen handelt. In diesem Fall muss man dann entscheiden, welchen Wert man benutzen will. Gewöhnlich wird man den Abstand zu dem Punkt wählen, der am weitesten entfernt ist. Ein anderes Problem entsteht bei relativ kleinen Objekten, bei denen dann nicht für jeden Projektionswinkel auch tatsächlich ein Konturpunkt existiert, so dass Interpolationen erforderlich werden.

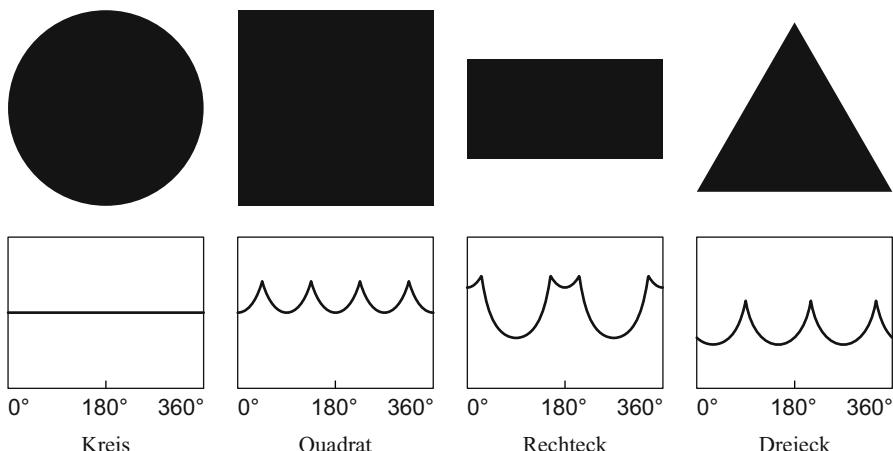


Abbildung 3.12: Polarabstandsprojektionen für einfache geometrische Figuren

Trotz der konzeptionellen Einfachheit ist der Rechenaufwand für die Bestimmung der Polarabstandsprojektion nicht unerheblich. Für jeden Strahl muss aus dem gegebenen Winkel eine Geradenrepräsentation erzeugt werden², es müssen alle Schnittpunkte dieser Geraden mit der Kontur festgestellt werden, und für jeden der 360 letzten Schnittpunkte, die bei einer Auflösung von einem Grad herauskommen, muss der Abstand zum Schwerpunkt bestimmt werden.

² Wenn man nicht die Koordinaten jedes Punkts einzeln mittels Winkelfunktionen berechnen will, was sehr langsam wäre.

Verwendung der Polarabstandsprojektion: Wenn nun für das Referenzobjekt und das aktuelle Objekt jeweils die Polarabstandsprojektion vorliegt, muss man beide vergleichen, um die aktuelle Orientierung des Objekts herauszufinden. Die grundsätzliche Vorgehensweise besteht darin, die Referenzprojektion über die Projektion des aktuellen Objekts zu legen, ein Ähnlichkeitsmaß zu berechnen, dann die Referenzprojektion eine Zelle weiter zu schieben, wieder das Ähnlichkeitsmaß zu berechnen usw., bis die Ausrichtung gefunden ist, bei der sich die Projektionen am ähnlichsten sind. Dabei können verschiedenen Ähnlichkeitsmaße Verwendung finden, z. B. der euklidische Abstand oder eine echte Korrelation, wobei letztere die, je nach Anwendung erwünschte oder weniger erwünschte Eigenschaft hat, die realen Größenverhältnisse der Objekte unberücksichtigt zu lassen.

Anschaulich kann man sich das auch so vorstellen, dass die Kontur des Referenzobjekts in Schritten, die der Auflösung der Projektion entsprechen, über der des aktuellen Objekts gedreht wird, bis die Konturen gewissermaßen übereinander einrasten. Die beiden Projektionen müssen dabei natürlich als zyklisch geschlossene Sequenzen betrachtet werden, nicht als einfache lineare Vektoren. Mit diesem Verfahren lässt sich die Orientierung auch für problematische Objekte mit großer Sicherheit bestimmen. Auch Konturdeformationen wie in Abbildung 3.5 verringern die Genauigkeit praktisch nicht, denn solange die Kontur nicht völlig entstellt wird, wird die höchste Ähnlichkeit immer noch an der gleichen Stelle ermittelt werden, nur der absolute Wert des Ähnlichkeitsmaßes wird absinken.

Ein Wort der Warnung ist jedoch angebracht: Der Hauptvorteil der Polarabstandsmethode ist die höhere Stabilität bei Objekten mit sehr ähnlichem ersten und zweitem Trägheitsmoment und bei deformierten Konturen. Die Genauigkeit des Verfahrens ist zwangsläufig auf die Auflösung der Projektion begrenzt, d. h. bei einer Auflösung von einem Grad kann auch die Orientierung nicht genauer bestimmt werden. Diese Genauigkeit lässt sich auch nicht beliebig erhöhen, denn bei einer Steigerung der Auflösung auf $0,1^\circ$ vergrößert sich nicht nur der Rechenaufwand ganz erheblich, es steigen auch die Anforderungen an die Referenzobjekte, die dann, um diese Auflösung auch wirklich auszunutzen, eine entsprechend lange Kontur aufweisen müssen. Insbesondere bei Vermessungsaufgaben mit hohen Genauigkeitsanforderungen kann keine nachträgliche Korrektur eine sorgfältig geplante und aufgebaute Bildverarbeitungsstation ersetzen, bei der die Objekte präzise und stabil positioniert werden.

3.3 Roboterpositionierung

Die bisherigen Beispiele mögen vielleicht etwas akademisch erscheinen, besonders im Lichte unseres Anspruchs, in diesem Buch zu zeigen, wie Bildverarbeitung industriell tatsächlich eingesetzt werden kann. Daher lösen wir nun unser Versprechen ein, reale Anwendungsbeispiele zu zeigen, die so in der Fertigung realisiert wurden und zeigen dabei gleichzeitig noch die Verbindung von Rotations-

und Translationskorrektur und die Verwendung ganzer Objektkonfigurationen als Positionsreferenz.

3.3.1 Aufgabenstellung

Abbildung 3.13 zeigt einen Drehknopf für ein elektronisches Gerät und seine Einzelteile. Diese Drehknöpfe werden automatisch montiert. Der Arbeitsschritt, um den es hier geht, ist das Einsetzen des Gummirings in den Korpus, das wie folgt abläuft:

1. Die Gummiringe laufen über ein Band in einen Vereinzeler.
2. Die vereinzelten Ringe rutschen auf eine Drehscheibe.
3. Ein Greifer hebt den Ring von der Drehscheibe auf und setzt ihn in den Korpus ein.



Abbildung 3.13: Drehknopf für elektronisches Gerät mit Einzelteilen

Dieser Vorgang wäre völlig problemlos, wenn es sich um einen allseits gleichermaßen runden Gummiring handelte, aber wie Sie in Abbildung 3.14 sehen können, hat der Ring eine nach innen zeigende Nase und eine Öffnung, kann also nur in einer Stellung in den Drehknopf eingesetzt werden. Der Ring muss dem Greifer daher in der richtigen Orientierung präsentiert werden. Die Feststellung der Lage des Rings auf der Drehscheibe nach dem Vereinzeler war in diesem Fall die Aufgabe eines Bildverarbeitungssystems.

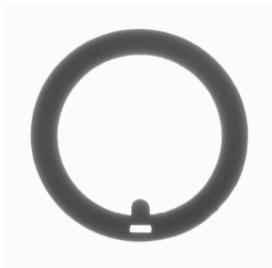


Abbildung 3.14: Gummiring für Drehknopf im Durchlicht

Abbildung 3.15 zeigt den Aufbau schematisch: Die transparente Drehscheibe wird von unten beleuchtet, so dass der Ring und seine Öffnungen wie in Abbildung 3.14 klar erkennbar sind. Die Drehscheibe wird von der Kamera aufgenommen, das Bildverarbeitungssystem ermittelt Lage und Orientierung des Rings, meldet diese Werte an die Steuerung der Drehscheibe und des Greifers. Die Drehscheibe richtet den Ring auf die festgelegte Orientierung aus, der Greifer wird an die entsprechende Position bewegt, nimmt den Ring auf und montiert ihn an der vorgesehenen Stelle³.

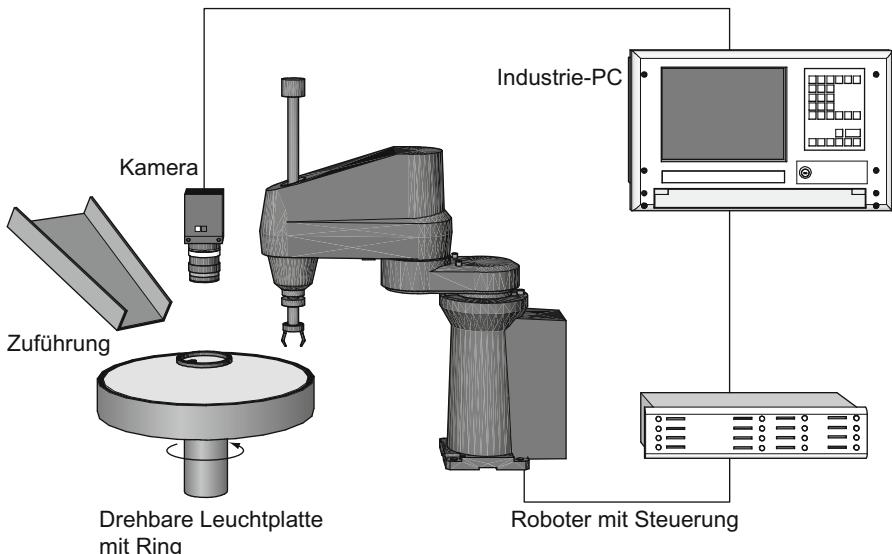


Abbildung 3.15: Schematischer Aufbau des Positionierungssystems für die Ringmontage

3.3.2 Bildverarbeitungskomponenten

Die vorliegende Aufgabe stellt keine besonders hohen Anforderungen an die optische Qualität. Der Gummiring hat einen Außendurchmesser von etwa 17 mm. Der Bereich der Drehscheibe, in dem er sich befinden kann, ist etwa 40 mm groß, so dass bei einer Kamerabildgröße von 1032×776 ein Bildpunkt rund 0,05 mm entspricht. Selbst wenn die Messung nur auf drei oder vier Pixel genau erfolgt, hätte der Greifer daher keine Schwierigkeiten, das Teil aufzunehmen.

³ Die Steuerung muss natürlich den Winkel, um den die Drehscheibe zuvor entsprechend der Orientierung des Rings gedreht wurde, in die Positions berechnung einbeziehen, aber das ist erheblich weniger Aufwand als noch einmal eine Bildaufnahme und Positionsbestimmung durchzuführen.

Die Drehscheibe besteht aus halbtransparentem Kunststoff, der von unten beleuchtet wird, so dass ein homogen leuchtender Hintergrund entsteht. Vor dieser diffusen Durchlichtbeleuchtung zeichnet sich der Ring klar ab.

Wegen der geringen optischen Anforderungen ist die Wahl des Objektivs für diese Aufgabe nicht besonders kritisch. Sie wird hauptsächlich von den Einbaubedingungen bestimmt, d. h. dem Abstand, der zwischen Drehscheibe und Kamera eingehalten werden muss, um die Bewegungsfreiheit des Greifers zu erhalten. Die Schärfentiefe stellt bei der geringen Dicke des Rings, guten Lichtverhältnissen und der Art der Aufgabe kein Problem dar.

3.3.3 Positionsermittlung an einem Objekt

Die Ermittlung der Position des Rings auf der Drehscheibe ist bei einem so klar definierten Objekt ausgesprochen einfach. Den Ring als Objekt zu finden, ist durch den hervorragenden Kontrast der Durchlichtaufnahme überhaupt kein Problem, womit der Schwerpunkt dann sehr einfach zu berechnen ist. Damit ist aber erst die halbe Aufgabe gelöst, denn die Orientierung ist noch unbekannt. An der kreisförmigen Außenkontur kann man die Orientierung selbstverständlich nicht bestimmen. Nach den vorangegangenen Ausführungen bietet es sich an, die Innenkontur zu benutzen, die in Form der ausgeprägten „Nase“ einen klaren Anhaltspunkt für die Ausrichtung des Teils.

Mithilfe der Polarabstandsprojektion ist es auch tatsächlich ein Leichtes, die Orientierung des Rings so zu bestimmen. Die Hauptachsenmethode funktioniert hier aus den bekannten Gründen nicht besonders stabil, da die Kontur doch nahezu kreisförmig ist. Abbildung 3.16 zeigt das Ergebnis der Positions- und Rotationsbestimmung anhand einer entsprechend nach geführten Suchlinie. Das Bild ganz links stellt die Referenzposition und -Orientierung dar. Sie sehen, dass in allen Bildern die definierte Suchlinie wieder genau durch die Nase und die Öffnung im Ring verläuft.

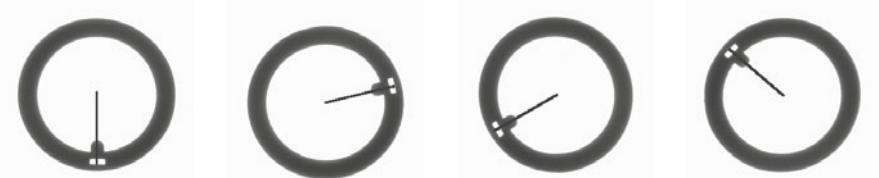


Abbildung 3.16: Ergebnis einer Nachpositionierung anhand der Polarabstandsprojektion der Innenkontur

Programmhinweis: Die Innenkontur und die Öffnung im Ring sind nach unserem Vorgehensmodell leicht zu finden, indem einfach innerhalb der Außenkontur eine zweite Suche nach hellen Objekten durchgeführt wird. Wenn man die Objektsuche

auf das größte Objekt beschränkt, erhält man nur die Innenkontur. Sie finden dieses Beispiel im Download im Verzeichnis Beispiele\Positionierung.

3.3.4 Orientierung einer Objektgruppe

Es gibt noch einen anderen Weg die Orientierung mit großer Sicherheit und Genauigkeit zu ermitteln. Innerhalb der Außenkontur des Rings gibt es vom Standpunkt eines Bildverarbeitungssystems aus zwei helle Objekte, obwohl man das als Mensch vielleicht nicht ganz so empfinden würde: den eigentlichen „Innenraum“ des Rings mit seiner charakteristischen Nase, die wir oben für die Ermittlung der Ausrichtung benutztten, und die benachbarte kleine Öffnung. Die Linie vom Mittelpunkt des gesamten Rings zum Schwerpunkt dieser Öffnung liefert eine ebenso gute Referenz für die Ausrichtung des Rings. Abbildung 3.17 zeigt die beiden Objekte, zur Unterscheidung von Ring und Hintergrund schwarz eingefärbt.

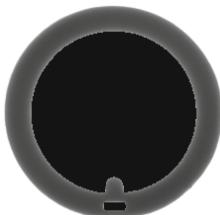


Abbildung 3.17: Objektgruppe zur Rotationsbestimmung

Nachdem die beiden Objekte gefunden sind, ist die Richtung der Verbindungsline ganz leicht zu ermitteln, indem man einfach die beiden Schwerpunkte in eine Geradengleichung einsetzt und den Winkel als Arcustangens der Steigung berechnet. Dieses Ergebnis ist allerdings nicht eindeutig, da der Arcustangens eine Periode von 180° aufweist. Die Orientierung fehlt also noch. Sie lässt sich aber problemlos bestimmen, indem wir definieren, dass die Verbindungsline immer am größeren der beiden Objekte, also dem Innenraum des Rings, beginnen soll. Aus der Objektsuche ist die Größe der beiden Objekte bekannt, so dass sich ganz leicht die definierte Richtung ermitteln lässt. Die Qualität der Positionierung ist ebenso gut wie bei der Polarabstandsmethode an der Innenkontur, die wir oben verwendeten, aber die Berechnung weit weniger aufwändig, so dass bei großen Objekten eine merkliche Zeitsparnis möglich ist.

Programmhinweis: Wenn im Prüfschritt *Arbeitsbereiche Position bestimmen* in **NeuroCheck** bei der Ermittlung der Referenzlage die Option „Arbeitsbereich-Gruppe“ zusammen mit der Einstellung „ $\pm 180^\circ$ “ benutzt wird, verfährt das Programm genau wie oben beschrieben.

Allgemein kann man festhalten, dass die Bestimmung von Position und Orientierung anhand der Anordnung mehrerer Objekte in der Regel sehr sicher, stabil und schnell erfolgen kann. In manchen Fällen ist es sogar die einzige Möglichkeit, z.B. bei rotationssymmetrischen Teilen, deren Ausrichtung nur anhand von Oberflächenmerkmalen festgestellt werden kann.

3.3.5 Bemerkungen zur Positionskorrektur

Wie wir oben gesehen haben, gibt es eine Vielzahl von Möglichkeiten, Positionen und Orientierungen zu bestimmen, die sich auf verschiedene Weise kombinieren lassen. Dabei gibt es einige Dinge zu beachten, die zu subtilen Fehlern führen können, die sich nicht immer auf Anhieb entdecken lassen.

Drehzentrum: Es ist klar, dass es nicht genügt, Arbeitsbereiche, deren Position und Orientierung korrigiert werden soll, zu verschieben und auf der Stelle zu drehen. Sie müssen auch um den Drehpunkt geschwenkt werden. Andernfalls passiert das, was in Abbildung 3.18 zu sehen ist. Dort wurde die Suchlinie zwar entsprechend der gemessenen Translation verschoben, aber nur am Platz gedreht. Man sieht, dass sie der neuen Ausrichtung des Rings entspricht, sich aber an der völlig falschen Stelle befindet.

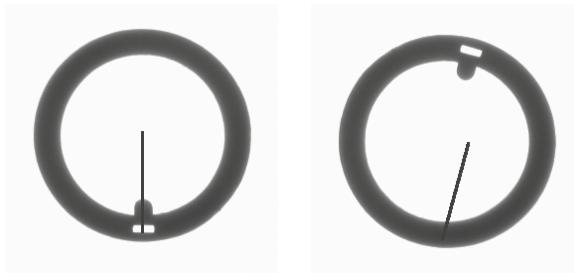


Abbildung 3.18: Fehlende Rotation um das Drehzentrum

Dieser Fehler ist natürlich ziemlich offensichtlich und wird daher kaum vorkommen. Ein subtilerer Fehler kann auftreten, wenn Translation und Rotation an verschiedenen Objekten gemessen werden. Dieser Fall ist sogar ziemlich häufig, denn wie wir weiter oben in Abbildung 3.9 sahen, eignen sich Objektkanten sehr gut zur Rotationsmessung. Man kann aber nicht gleichzeitig auch die Translation an der Kante messen, da die Kante sich ja nicht um sich selbst, sondern um den Schwerpunkt des Gesamtobjekts dreht. Der Schwerpunkt der Kante hat daher eine Pseudoverschiebung. In einem solchen Fall wird man also die Translation am Gesamtobjekt messen, die Rotation an der Kante. Die Drehung muss aber immer um das Referenzobjekt der Translation erfolgen, wie Abbildung 3.19 zeigt. Die Messung der Rotation erfolgt hier, wie in Abbildung 3.9, an der geglätteten Gewindekante. Im oberen Bild wurde die Drehung korrekt um den Schwerpunkt des Ge-

samtobjekts ausgeführt, im unteren Bild fälschlicherweise um den Schwerpunkt der Referenzkante.



Abbildung 3.19: Drehung um falsches Drehzentrum

Reihenfolge von Drehung und Verschiebung: Diesen Aspekt braucht man gewöhnlich nur zu beachten, wenn man selbst Bildverarbeitungsanwendungen programmiert, aber auch bei der Arbeit mit sofort einsetzbarer Bildverarbeitungssoftware können Situationen vorkommen, in denen man die Reihenfolge von Drehungs- und Verschiebungskorrektur beachten muss. Das kann beispielsweise der Fall sein, wenn aufeinander folgende Korrekturen erforderlich sind, um verschiedene Referenzobjekte überhaupt erst finden zu können. In diesem Fall muss die Translationskorrektur *vor* der Rotationskorrektur erfolgen. Abbildung 3.20 zeigt den Effekt, der eintritt, wenn im zweiten Bild von Abbildung 3.16 zuerst die Orientierung der Linie korrigiert wird, dann die Verschiebung erfolgt.

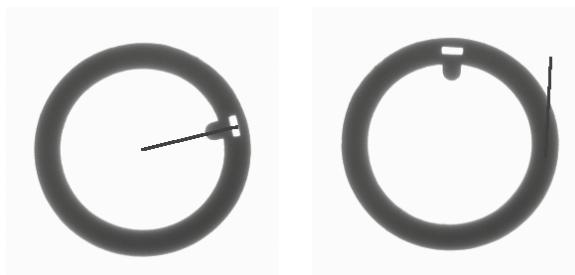


Abbildung 3.20: Trennung von Translations- und Rotationskorrektur, links in richtiger, rechts in falscher Reihenfolge

Pseudoverschiebungen: Wir erwähnten schon, dass Formveränderungen zu scheinbaren Verschiebungen des Schwerpunkts führen können. Aber auch ohne eine optisch bedingte Formveränderung kann ein solcher Effekt auftreten. Aufgrund der Nase liegt der Schwerpunkt des Ringinnenraums, der in 3.3.3 zur Rotationsbestimmung benutzt wurde, nicht genau im Zentrum des Rings. Bei einer Drehung des Rings verschiebt sich folglich der Schwerpunkt des Innenraums gegenüber dem Schwerpunkt der Außenkontur. Bei dieser Aufgabe mit ihren relativ niedrigen Genauigkeitsanforderungen entsteht hierdurch kein Problem, bei filigranen Aufgabenstellungen muss man jedoch derartige Effekte berücksichtigen.

In manchen Anwendungen ist es vorteilhaft, die Verschiebung nicht anhand des Objektschwerpunkts zu messen, sondern anhand der vertikalen oder horizontalen Ausdehnung, also z. B. die vertikale Verschiebung anhand des obersten Punkts. Dies ist aber nur möglich, wenn sichergestellt ist, dass sich das Objekt nicht dreht, da durch eine Drehung um den Schwerpunkt eine Verschiebung der äußersten Punkte des Objekts hervorgerufen wird. Prinzipiell lässt sich diese Pseudoverschiebung natürlich bei bekannter Drehung zurückrechnen, bei komplizierten Objektkonturen kann das aber sehr aufwändig sein.

3.4 Zusammenfassung

Bedeutung von Positionsbestimmungen: Sie sind Bestandteil der meisten Bildverarbeitungsanwendungen, da sich Position und Orientierung von Prüfteilen vor der Kamera nur selten mit absoluter Genauigkeit reproduzieren lassen. In einem solchen Fall ist die Positionsbestimmung lediglich ein Zwischenschritt, ein Hilfsmittel für die Durchführung der eigentlichen Sichtprüfaufgabe.

Roboterpositionierung: In manchen Fällen ist die Positionsbestimmung das eigentliche Ziel der Bildverarbeitungsanwendung. Ein typisches Beispiel hierfür ist die Roboterpositionierung, bei der die Lage des Prüfobjekts an einen Roboter gemeldet wird, der das Objekt dann z. B. greifen soll.

Translationskorrektur: Bei der Translationskorrektur werden Verschiebungen des Prüfteils gegenüber einer Referenzposition festgestellt und korrigiert. Die Positionsermittlung erfolgt meist anhand des Schwerpunkts, es können aber auch Objektkanten d. h. die äußersten Punkte in einer Koordinatenrichtung, benutzt werden. Formveränderungen, z. B. durch Beleuchtungseffekte, und Drehungen des Objekts können zu Pseudoverschiebungen führen.

Rotationskorrektur: Bei der Rotationskorrektur werden Drehungen des Prüfteils gegenüber einer Referenzlage festgestellt und korrigiert. Gängige Methoden zur Rotationsbestimmung sind die Hauptachsenberechnung und die Polarabstandsprojektion.

Hauptachsenberechnung: Eine formal exakte Berechnung der Hauptachse ist sehr rechenaufwändig, Näherungsverfahren eignen sich nicht für alle Objekte. Besonders Objekte mit einem Höhen-/Breitenverhältnis nahe eins sind hier problematisch, da die Achse unvermittelt um 90° kippen kann. Sehr gut eignen sich lang gestreckte Objekte, wie z. B. Kanten, ggf. nach vorheriger Glättung. Hauptachsenberechnungen werden außerdem durch Formänderungen, z. B. aufgrund von Beleuchtungsschwankungen, stark beeinflusst.

Polarabstandsprojektion: Bei dieser Methode zur Rotationsbestimmung werden die Abstände zwischen Objektschwerpunkt und Objektkontur in bestimmten Winkelinkrementen (typischerweise ein Grad) gemessen und in einen Projektionsvektor eingetragen. Dieser wird dann mit dem Projektionsvektor des Objekts in der Referenzlage korreliert. Diese Methode ist im Allgemeinen nicht so rechenaufwändig wie eine mathematisch exakte Hauptachsenberechnung, aber wesentlich stabiler als Näherungsverfahren hierfür. Die Genauigkeit hängt vom verwendeten Winkelinkrement ab und lässt sich bei kleinen Objekten mit wenigen Konturpunkten nicht beliebig steigern.

Hierarchische Korrekturen: Positionskorrekturen lassen sich hierarchisch hintereinander staffeln. Das ist oft in komplexen Bildszenen der Fall, in denen das eigentliche Referenzobjekt selbst erst nach einer Positionskorrektur gefunden werden kann.

Objektgruppen als Referenz: Objektgruppen sind eine sehr stabile Referenz für Positions- und Rotationsbestimmungen, die auch bei rotationssymmetrischen Objekten mit identifizierbaren Oberflächenmerkmalen eingesetzt werden kann. Das Verfahren ist außerdem sehr schnell, da nur die Schwerpunkte der segmentierten Objekte erforderlich sind (und ggf. die Größe der Objekte, wenn eine 180° -Orientierung bestimmt werden muss).

Rotationsbestimmung an Kanten: Eine zur Rotationsbestimmung verwendete Kante sollte nicht zur Berechnung der Translation verwendet werden, da eine Rotation um den Schwerpunkt des Gesamtobjekts eine Pseudoverschiebung der Kante hervorruft.

Translationsbestimmung an Kanten: Wenn die Translation anhand von Kanten anstelle von Schwerpunkten bestimmt wird, muss man darauf achten, dass keine Pseudoverschiebung durch Drehungen des Objekts auftreten kann.

4. Querschnitt: Segmentierung

Der Begriff des *Objekts* ist zentral für das Vorgehensmodell in Abschnitt 1.5, ja in der Tat entscheidend für das Wesen der industriellen Bildverarbeitung, geht es hier doch letztlich immer darum, eine Aussage über in der realen Welt existierende und in Bildern erfasste Objekte zu machen. Im Einführungsbeispiel in Abschnitt 1.6 und im gesamten Kapitel 3 über Lageerkennung wurde entsprechend immer wieder Gebrauch von Verfahren gemacht, die eine *Segmentierung* vornehmen, d. h. Objekte aus der Bildszene isolieren. Dabei wurde zunächst einmal angenommen, dass diese Verfahren verfügbar sind und die gewünschten Objekte aus der Bildszene isolieren können. Im Laufe der Zeit wurde eine Vielzahl derartiger Methoden entwickelt. Die wichtigsten und am meisten verbreiteten werden in diesem Kapitel vorgestellt.

4.1 Arbeitsbereiche

Wir beginnen dieses Kapitel mit einem Abschnitt über *Arbeitsbereiche*. Bereits in Abschnitt 1.5 wurde darauf hingewiesen, dass wir in unserem Vorgehensmodell für Bildverarbeitungsanwendungen keinen grundsätzlichen Unterschied zwischen Objekten machen, die aus der Bildszene segmentiert werden, und solchen, die der Anwender manuell definiert. Diese Äquivalenz ist ein ungewöhnlicher Denkan-satz, der sich nur in wenigen Bildverarbeitungssystemen wieder findet und darum zu Beginn dieses Kapitels über Bildobjekte noch einmal erläutert und begründet werden soll. Zudem lassen sich einige Aspekte an manuell definierten Objekten leichter erklären, da gewisse Komplikationen entfallen, die bei segmentierten Objekten auftreten.

4.1.1 Arbeitsbereiche und Objekte

Anwendungen ohne Segmentierung: Es ist durchaus möglich, in einer Bildverarbeitungsanwendung gänzlich ohne Segmentierung auszukommen. Bei einer Oberflächenprüfung kann es z. B. genügen, in einem vorher definierten Bereich des Bildes – oder auch dem gesamten Bild – bestimmte charakteristische Zahlenwerte zu berechnen, wie etwa die mittlere Helligkeit, die Streuung der Grauwerte, das Vorkommen von Grauwertübergängen etc.

Bei einer Oberfläche, die homogen weiß erscheinen soll, lässt sich das Vorhandensein von Kratzern oder Verschmutzungen einfach dadurch feststellen, dass ein

starker Kontrast oder Grauwertsprünge auftreten. Andererseits kann die Problemstellung auch verlangen, dass genau diese Merkmale für die Oberfläche eines Objekts berechnet werden sollen, das zunächst einmal aus der Bildszene isoliert werden muss. Wenn aber dieselbe Merkmalsberechnung auf zwei verschiedenen Gebilden erfolgen soll, erscheint es zumindest programmtechnisch sinnvoll, eine Verallgemeinerung einzuführen, die eine Gleichbehandlung beider Strukturen, des manuell definierten Arbeitsbereiches und des automatisch gefundenen Objekts, erlaubt. Damit ist eine erste Motivation für die Äquivalenz beider Strukturen gegeben.

Hierarchische Suche: In den bisherigen Beispielen in Abschnitt 1.6 und Kapitel 3 sind wir jeweils so vorgegangen, dass Rechtecke und Suchlinien definiert wurden, die dann zur Detektierung einzelner Ziffern oder Objektkonturen dienten. Das ist die typische Verwendung von Arbeitsbereichen (Suchgebieten, Prüffenstern, Areas of Interest, Regions of Interest, ROIs) in der Bildverarbeitung: Eingrenzung des Bereichs, in dem Bildobjekte segmentiert werden sollen.

In der Technik sind hierarchische Strukturen sehr häufig: auf einem Motor befindet sich eine Plakette, auf der Plakette befinden sich Schriftzeichen die gelesen werden sollen; Bauteile sind in Gruppen auf einer Platine angeordnet; auf einer Gehäusefläche befinden sich Bohrungen, in diesen wiederum Gewindegänge; die Liste ließe sich beliebig fortsetzen. Der Gedanke liegt nahe, diese hierarchische Struktur bei der Suche nach Bildobjekten nachzubilden: Statt das gesamte Motorgehäuse nach den Schriftzeichen abzusuchen, sucht man zunächst nach der Plakette, die gewöhnlich viel leichter zu finden ist, dann erst nach den einzelnen Schriftzeichen. Man führt also auf dem bereits durch Segmentierung gefundenen Bildobjekt „Plakette“ genau dieselbe Operation durch wie sonst auf einem manuell definierten Suchbereich. Damit ist die Äquivalenz von Bildobjekten und Arbeitsbereichen auch von dieser Seite her motiviert.

Unterschiede zwischen Arbeitsbereichen und Objekten: Es ist klar, dass sich manuell definierte Arbeitsbereiche und durch Segmentierung gefundene Objekte in gewisser Hinsicht verschieden verhalten müssen. So ist z. B. eine Lageerkennung an manuellen Arbeitsbereichen zwar möglich aber sinnlos, da sie sich immer an derselben Stelle und in derselben Orientierung befinden werden. Umgekehrt ist eine Lagekorrektur segmentierter Objekte wenig nutzbringend, da sich die Objekte ja zwangsläufig an ihrer wahren Bildposition befinden müssen.

4.2 Binärsegmentierung

In Kapitel 3 war mehrfach die Rede davon, in einem Suchbereich oder auf einer Suchlinie einen Hell-Dunkel-Übergang zu detektieren. Verfahren, die auf der Unterscheidung heller und dunkler Bereiche beruhen, werden als Binärsegmentierung bezeichnet, da sich die beiden Zustände Hell und Dunkel in einem Binärbild darstellen lassen, d. h. einem Bild mit nur zwei Zuständen, die üblicherweise als Weiß

und *Schwarz* bezeichnet werden. Zwei Aspekte sind entscheidend für die Binärsegmentierung:

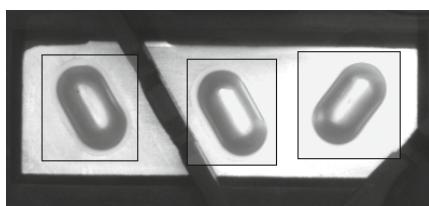
- Die Definition für Hell und Dunkel; wir gehen hier stets von einem Grauwertbild mit 256 Helligkeitsstufen aus, es muss also festgelegt werden, welche Helligkeitsbereiche des Grauwertbildes auf die binären Zustände Weiß und Schwarz abgebildet werden. Die häufigste, aber keineswegs einzige Methode ist die Verwendung eines Schwellwertes, die im Folgenden ausführlich besprochen wird.
- Die Methode zur Erzeugung der Objektrepräsentation aus dem Binärbild. Grundsätzlich kann man hier zwischen konturbasierten und regionenbasierten Methoden unterscheiden, d. h. solchen, die zunächst den Umriss eines Objekts suchen, dann daraus die Punktmenge erzeugen, die das Objekt repräsentiert, und solchen, die von einem Punkt der gesuchten Farbe ausgehen und das Objekt von innen nach außen erzeugen.

4.2.1 Schwellwerte

In diesem Abschnitt befassen wir uns mit dem ersten Aspekt der Binärsegmentierung, der Bestimmung, welche Bereiche des Grauwertbildes im Binärbild hell, welche dunkel erscheinen.

Globale Schwellwerte: Die einfachste Methode, ein Grauwertbild in ein Binärbild umzuwandeln, besteht darin, alle Punkte mit einer Helligkeit unterhalb eines bestimmten Wertes als schwarz, alle anderen als weiß zu definieren. Den betreffenden Wert bezeichnet man als *Binärschwelle*.

Abbildung 4.1 zeigt den Effekt eines globalen Schwellwertes auf ein Grauwertbild. Man sieht deutlich, dass die Unterscheidung der hellen Metallkontakte vom dunklen Kunststoff nicht in jedem der eingezeichneten Arbeitsbereiche gleich gut gelingt. Die Helligkeit der Kontakte ist aufgrund des Beleuchtungswinkels zu unterschiedlich für die Verwendung einer globalen Schwelle. Daher franst das linke Objekt aus.



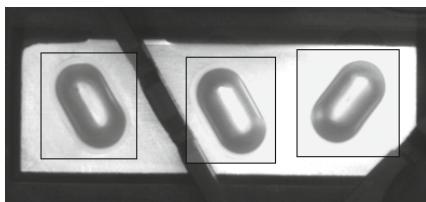
Originalbild mit Arbeitsbereichen



Wirkung einer globalen Schwelle

Abbildung 4.1: Binarisierung eines Grauwertbildes mit globaler Schwelle

Ortsabhängige Schwellwerte: Probleme bei der Segmentierung mit einem globalen Schwellwert treten immer dann auf, wenn relevante Bildbereiche unterschiedliche Helligkeitseigenschaften aufweisen. Weder Bilder mit lokal unterschiedlicher Beleuchtung lassen sich so segmentieren, noch eignet sich ein globaler Schwellwert, wenn in einem Bereich des Bildes das gesuchte Objekt auf andere Weise mit dem Hintergrund kontrastiert als in einem anderen. Abbildung 4.2 zeigt, wie sich das Ausfransen des linken Objekts in Abbildung 4.1 durch lokale Schwellen vermeiden lässt. Für jeden Arbeitsbereich wird hier ein eigener Schwellwert berechnet.



Originalbild mit Arbeitsbereichen



Wirkung lokaler Schwellen

Abbildung 4.2: Binarisierung eines Grauwertbildes mit lokalen Schwellen

In diesem Fall kann man die Segmentierung leicht vornehmen, indem man in den verschiedenen Bildbereichen unterschiedliche Schwellwerte zulässt. Der konsequent an Bildobjekten bzw. Arbeitsbereichen orientierte Ansatz aus Abschnitt 1.5 unterstützt ein solches differenziertes Vorgehen ganz natürlich, indem man jedem Objekt den jeweiligen Schwellwert als Eigenschaft mitgibt und die Segmentierungsfunktion in jedem Arbeitsbereich diese Eigenschaft benutzen lässt.

Bei den bisher besprochenen Methoden zur Schwellwertfestlegung ist es übrigens für den Segmentierungsprozess nicht unbedingt erforderlich, ein Binärbild auch tatsächlich zu erzeugen. Es genügt vielmehr, in jedem Bildpunkt die Entscheidung zu treffen, ob er ober- oder unterhalb des Schwellwertes liegt, um ihn einem Objekt zuzuordnen.

Diese Vorgehensweise ist allerdings praktisch kaum mehr möglich, wenn man die Festlegung lokaler Schwellwerte bis zum Extrem treibt und jedem einzelnen Bildpunkt einen eigenen Schwellwert zuordnet. Ein derart aufwändiges Vorgehen kann bei Bildern problematischer Qualität durchaus sinnvoll sein, jedoch wird man es dann oftmals vorziehen, das Bild einer Vorverarbeitung zu unterziehen, die danach eine einfachere Segmentierung mit wenigen lokalen Schwellwerten erlaubt (Vorverarbeitungsmethoden werden in Kapitel 2 besprochen).

4.2.2 Schwellwertbestimmung aus Histogrammanalysen

Bisher haben wir angenommen, dass der Schwellwert (oder die Schwellwerte, wenn lokale Schwellen verwendet werden) einfach vorgegeben wird, und in der Tat gibt es Anwendungen, bei denen man so vorgehen und einen bestimmten

Grauwert als Binärschwelle einstellen kann. Gerade unter Fertigungsbedingungen wird man jedoch oft keine ausreichend stabilen Verhältnisse vorfinden. Die Beleuchtung kann Schwankungen unterliegen, aber in der Regel ist es das wechselnde Erscheinungsbild von Objekten, das die größeren Probleme aufwirft. Prüfteile können verschmutzt sein, Kontrasteigenschaften können sich von Teil zu Teil ändern, wie es Abbildung 4.3 zeigt. In jedem der beiden Bilder ist eine Segmentierung mit einer globalen Schwelle möglich, jedoch nicht mit derselben Schwelle in beiden Bildern. Typischerweise wird man also auf die Eigenschaften des gerade vorliegenden Bildes reagieren und den Schwellwert aus dem Bild – oder bei lokalen Schwellwerten einzeln aus jedem Arbeitsbereich – ableiten müssen.



Abbildung 4.3: Segmentierung mit fester Schwelle bei unterschiedlichen Kontrasteigenschaften

Programmhinweis: **NeuroCheck** unterstützt die Verwendung unterschiedlicher Binärschwellen durch die Option, die Schwelle innerhalb jeder Gruppe von Arbeitsbereichen separat mittels Bimodalitätsanalyse zu optimieren (siehe hierzu Abschnitt 4.2.3).

Eine typische Methode zur Bestimmung von Schwellwerten aus Bildeigenschaften besteht darin, das Grauerthistogramm des Bildes – oder einzelner Bildbereiche – zu analysieren. Der folgende Abschnitt stellt daher Grauerthistogramme und Analysemethoden vor.

4.2.3 Grauerthistogramme

Die Binärschwelle als Segmentierungsinstrument basiert auf den Helligkeitseigenschaften der Bildbereiche. Das typische Instrument zur Darstellung der Helligkeitseigenschaften von Bildern ist das *Grauerthistogramm*.

Definition: In der beschreibenden Statistik versteht man unter einem Histogramm die Darstellung einer Häufigkeitstabelle in einem Balkendiagramm. Wir gebrauchen es hier meist gleichermaßen für die Tabelle selbst und die spezielle Darstellungsform. In der Häufigkeitstabelle wird erfasst, wie oft bestimmte Klassen von Werten vorkommen. Genauso verfährt man auch bei einem Grauerthistogramm. Für jeden Grauwert wird im Histogramm eingetragen, wie oft er im gesamten Bild oder einem Teilbereich vorkommt.

In der Regel werden diese Häufigkeiten für jeden Grauwert einzeln gezählt. Man kann natürlich auch, wie das in der Statistik vielfach üblich ist, mehrere Grauwerte zu Grauwertklassen zusammenfassen, z. B. in Stufen von 16 Grauwerten. Man erhielte dann ein Histogramm mit nur 16 Werten anstelle von 256. Der erste Balken erfasst dann die Häufigkeit aller Grauwerte zwischen 0 und 15 einschließlich, der zweite zwischen 16 und 31 etc. Wir werden uns hier aber stets auf Histogramme mit der vollen Auflösung von 256 Graustufen beziehen. Ein solches Histogramm ist sehr einfach zu berechnen, indem man einfach ein Feld von 256 Zählern anlegt, dann eine Schleife über alle Bildpunkte durchführt und für jeden Bildpunkt die seinem Grauwert entsprechende Zelle des Feldes inkrementiert. Wenn nur einzelne Bildbereiche verarbeitet werden, womöglich sogar von unregelmäßiger Form, also segmentierte Arbeitsbereiche, ist das Ganze natürlich entsprechend komplizierter, da man darauf achten muss, nur zur Region gehörige Pixel mitzuzählen. Ein wesentlicher Aspekt bleibt jedoch erhalten: In jedem Schritt wird immer nur der Grauwert eines einzigen Bildpunkts verwendet. Daher ist das Histogramm eine statistische Beschreibung erster Ordnung für die Verteilung der Grauwerte im Bildbereich. Beispiele für Statistiken zweiter Ordnung finden sich v. a. im Bereich der Texturanalyse, wo es auf räumliche Beziehungen zwischen den Grauwerten ankommt.

Bimodalitätsprüfung: (Sahoo et al. 1988) gibt eine umfangreiche Übersicht verschiedener Methoden zur Bestimmung von Binärschwellen aus Grauerthistogrammen. Wir beschränken uns hier auf die Bimodalitätsanalyse, die in (Sonka et al. 2008) ausführlich besprochen wird. Diese Methode geht von der Voraussetzung aus, dass ein typisches Histogramm zwei ausgeprägte Maxima aufweist, eines für den Hintergrund, eines für die Objekte. Abbildung 4.4 zeigt das Histogramm des Bildes eines Relaiskontakte. Man sieht, dass die Voraussetzung der Bimodaliät gut erfüllt ist, was auch auf viele industrielle Bildszenen zutrifft.

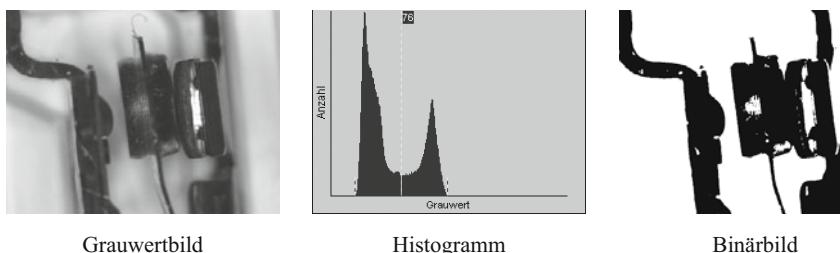


Abbildung 4.4: Schwellenbestimmung mittels Histogrammanalyse

Die übliche Vorgehensweise besteht darin, die Binärschwelle in das Minimum zwischen den beiden höchsten Maxima zu legen. Häufig wird man besondere Maßnahmen ergreifen müssen, um zu vermeiden, dass die Schwelle zwischen zwei lokale Maxima zu liegen kommt, die zu ein- und demselben globalen Maximum gehören. Mögliche Methoden hierfür sind die Forderung eines Mindestabstandes zwischen den gewählten Maxima oder eine vorhergehende Histogrammglättung.

Ungeachtet des naheliegenden, einfachen Ansatzes sind eine große Anzahl von Verfeinerungen der Bimodalitätsanalyse entwickelt und untersucht worden, von denen hier nur einige wenige erwähnt werden können. Weitere Methoden werden in (Sonka et al. 2008) und (Gonzalez und Woods 2008) besprochen. Dort finden sich auch umfangreiche Literaturhinweise hierzu.

Zu guten Segmentierungsergebnissen führt erfahrungsgemäß eine Verschiebung der Schwelle entsprechend der Fläche der beiden Hauptmaxima; die Schwelle wird also in den Schwerpunkt der Maxima gelegt. Dieses Verfahren kann allerdings zu einem Problem führen, wenn eines der Maxima erhebliches Übergewicht hat, da die Schwelle dann in das Maximum hineinrutscht. Abhilfe dagegen lässt sich leicht durch eine nichtlineare Skalierung des Histogramms schaffen, z. B. mit einer Wurzel- oder – stärker – einer Logarithmusfunktion. Abbildung 4.5 zeigt den Effekt dieser Skalierungsmethoden auf die Gestalt des Histogramms und das Segmentierungsergebnis.

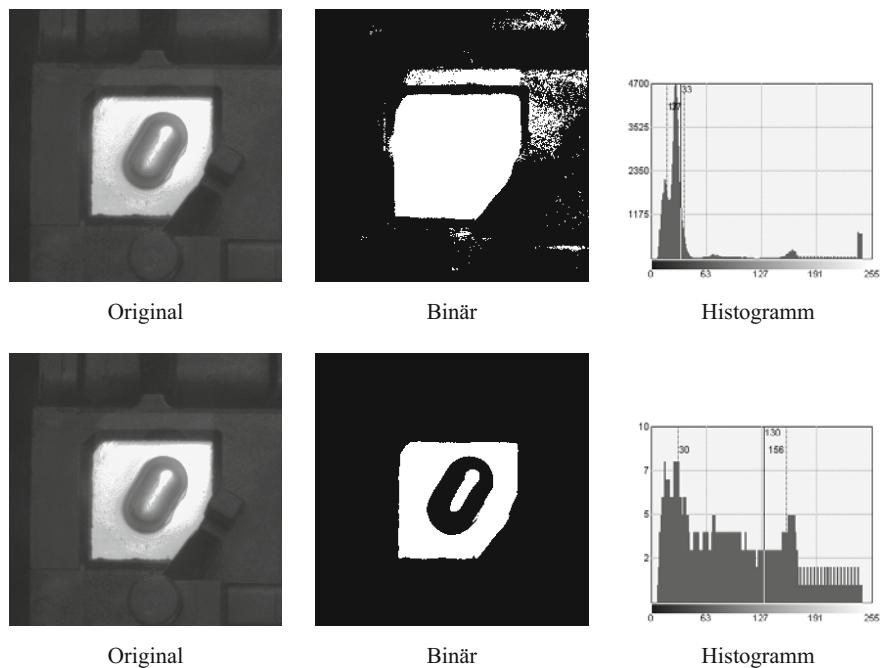


Abbildung 4.5: Nichtlineare Transformation eines ungleichgewichtigen Histogramms

In der praktischen Anwendung tritt häufig das Problem von Artefakten auf. Als Artefakte bezeichnet man Bildbereiche, die bei der Segmentierung als Objekte erscheinen, in der RealweltSzene jedoch keine echten Objekte darstellen. Dies können z. B. Reflexe sein, die als helle Objekte hervortreten, oder umgekehrt Schattenwürfe. In der Regel lassen sich solche Artefakte durch Vorwissen über den Aufbau der Szene oder aufgrund ihrer Größe von der Segmentierung ausschließen, sie stellen aber ein Problem für die Histogrammanalyse dar. Derartige Störungen sind gewöhnlich auf sehr enge Grauwertbereiche im extrem hellen oder dunklen Bereich begrenzt, bilden also ausgeprägte Maxima, die einen starken Einfluss auf die Schwellenberechnung mittels Modalitätsanalyse ausüben, aber für die Analyse der Szene in Wahrheit ohne Bedeutung sind. In solchen Fällen ist es von Vorteil, die Extrembereiche des Histogramms auszublenden.

Programmhinweis: Zur Transformation ungleichgewichtiger Histogramme bietet **NeuroCheck** im Prüfschritt *Binärschwelle erzeugen* die Option „Histogrammausgleich“, die ein Histogramm entweder mit der Wurzel- oder der Logarithmusfunktion skaliert. Außerdem können Arbeitsbereiche, die keinerlei Objekte enthalten, durch den Parameter „Minimaler Kontrast“ aus der Berechnung des Histogramms ausgeschlossen werden. Dadurch wird verhindert, dass reine Hintergrundbereiche die Gewichte im Histogramm verschieben. Zur Unterdrückung des Einflusses von Artefakten schließlich dient die Option „Störunterdrückung“.

Es gibt natürlich noch eine Reihe weiterer Ansätze. Man kann z. B. alle Bildpunkte aus Gebieten mit hohen Gradienten ausschließen. Da hohe Grauwertgradienten an Objekträndern vorkommen, besteht dann das Histogramm nur aus Punkten, die tatsächlich im Innern von Objekten oder im „Innern“ des Hintergrundes liegen, nicht aber auf den Kanten. Dadurch wird das Minimum ausgeprägter und leichter aufzufinden. Auch die umgekehrte Vorgehensweise ist denkbar.

4.2.4 Verallgemeinerung der Binarisierung

Bereichsbinarisierung: Bisher war in diesem Abschnitt stets von Bildern die Rede, in denen entweder der Hintergrund hell und das interessante Objekt dunkel war, oder umgekehrt. Es kann natürlich auch der Fall eintreten, dass das Objekt von mittlerer Helligkeit ist, der Hintergrund aber teils dunkler, teils heller als das Objekt. Ein Beispiel dafür ist die Abgrenzung einer Dichtlippe oder Kleberaupe, die einen hellen Innenbereich und einen dunklen Außenbereich trennt.

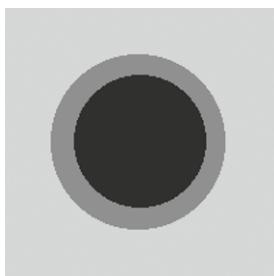
Auch die umgekehrte Situation ist möglich. Mit den bisher vorgestellten Schwellwertmethoden könnte man z. B. nicht ein Schachbrett muster von einem homogen grauen Hintergrund trennen. Im technischen Bereich tritt ein solcher Effekt z. B. bei Schlagzahlen auf. Je nach Einfallsinkel des Lichts erscheint ein Teil der Kanten einer Schlagzahl hell, ein Teil dunkel, während der weitgehend ebene Hintergrund homogen bleibt.

Es gibt eine Reihe von „Tricks“, mit denen man sich hier behelfen kann. Look-Up-Tabellen, wie sie in Abschnitt 2.1.1 besprochen werden, sind ein Beispiel für

einen Vorverarbeitungskunstgriff, der eine Segmentierung mit einer Schwelle auch bei derartigem Bildmaterial ermöglicht. Look-Up-Tabellen, so nützlich sie sein können, haben für diese Problemstellung einige Nachteile:

- Obwohl Look-Up-Tabellen eine der schnellsten Vorverarbeitungsmethoden darstellen, erfordern sie doch die Bearbeitung jedes einzelnen Bildpunkts, zumindest innerhalb des fraglichen Arbeitsbereiches.
- In ihrer Standardform sind Look-Up-Tabellen nicht kontextsensitiv, sie verarbeiten das gesamte Bild auf die gleiche Weise. Es ist natürlich möglich, eine Look-Up-Tabelle auf die Umgebung des zu bearbeitenden Punkts abzustimmen, damit steigt aber der Rechen- und Verwaltungsaufwand ganz erheblich.
- Statische Look-Up-Tabellen sind nur schwer auf die Anforderungen dieser Fälle abzustimmen; es ist sehr viel visuelle Kontrolle und Iteration notwendig, um eine Look-Up-Tabelle zu erhalten, die unter den typischen Schwankungen industrieller Bilder zufrieden stellende Ergebnisse liefert.

Eine weitaus elegantere und einfachere Methode besteht in der Einführung der Begriffe *logisch Weiß* und *logisch Schwarz*. Bei der normalen Binarisierung, die wir bisher behandelt haben, werden alle Bildpunkte oberhalb einer bestimmten Schwelle als weiß definiert, alle anderen als schwarz. Natürlich kann niemand uns daran hindern, einfach noch eine Schwelle einzuführen und alle Punkte mit einer Helligkeit oberhalb dieser Schwelle wieder als schwarz zu definieren. Das läuft natürlich dem anschaulichen Empfinden völlig zuwider, ermöglicht aber ohne besonderen Zusatzaufwand gegenüber der bisher besprochenen „normalen“ Binarisierung das Herausschneiden von Objekten oder Hintergrundbereichen mittlerer Helligkeit. Abbildung 4.6 verdeutlicht das Prinzip der Bereichsbinarisierung. In diesem Fall ist der Bereich zwischen den Grauwerten 100 und 150 als logisch Weiß definiert worden.



Originalbild



Bereichsbinarisierung

Abbildung 4.6: Prinzip der Bereichsbinarisierung

Bei der Bestimmung dieser Schwellen kann man sich aller Hilfsmittel der Histogrammanalyse bedienen, wobei die Komplexität natürlich durch mögliche Abhängigkeiten zwischen den Schwellen ansteigt. Durch weitere Verallge-

meinerung kommt man dann schließlich zu Multischwellentechniken, womit sich Objekte in mehreren Grauwertbändern in einem Schritt segmentieren lassen.

Nicht helligkeitsbasierte Schwellen: Abschließend sei darauf hingewiesen, dass die Umwandlung eines Grauwertbildes in ein Binärbild nicht unbedingt auf der Grundlage einer Helligkeitsschwelle stattfinden muss. Allgemein gesagt, handelt es sich ja bei der Einteilung in (logisch) weiße und schwarze Punkte um eine Klassifikation, und nirgendwo steht geschrieben, dass diese Klassifikation nur auf dem Merkmal „Helligkeit“ beruhen kann.

Eine sehr einfache Verallgemeinerung besteht z. B. darin, die Entscheidung zwischen Weiß und Schwarz aufgrund der mittleren Helligkeit in einer Umgebung des Punkts zu treffen. Damit lassen sich kleinere Störungen ausgleichen. Diese und viele andere Verallgemeinerungen lassen sich auch durch eine gewöhnliche Binarisierung mit vorhergehender Filterung des Bildes erreichen, hier mit einem Mittelwertfilter.

Aufwändiger Methoden beruhen auf einer regelrechten Klassifizierung jedes Bildpunkts. Dabei wird eine Vielzahl von Merkmalen für die Umgebung jedes einzelnen Punkts berechnet, neben so einfachen Werten wie der mittleren Helligkeit sind dabei z. B. auch Grauwertstatistiken zweiter Ordnung möglich, also Merkmale, die die räumliche Verteilung der Punkte beschreiben, die dem Menschen als *Textur* der Oberfläche erscheint. Je nach Klassifikationsmethode können dabei auch beliebig komplexe Abhängigkeiten zwischen diesen Merkmalen berücksichtigt werden. Man muss sich allerdings darüber klar sein, dass gerade die Berechnung Textur beschreibender Merkmale ausgesprochen rechenaufwändig sein kann und sich kaum für Echtzeitsegmentierungen eignet.

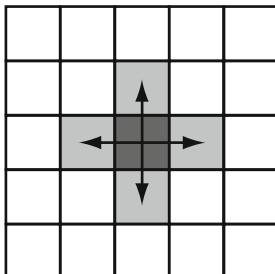
Durch die Verbindung von Methoden zur Vorverarbeitung und Pixelklassifikation mit der gewöhnlichen Binarisierung sind sehr komplexe Segmentierungen möglich. Das resultierende Binärbild, sei es nun durch eine einfache Helligkeitsschwelle oder eine komplexe Klassifikation von Punktmerkmalen entstanden, ist geometrisch kongruent zum ursprünglichen Grauwertbild. Die Beschreibung eines aus dem Binärbild extrahierten Objekts bzw. Arbeitsbereiches in Form von Koordinaten, d. h. als abstrakte Menge von Punkten, ermöglicht dann die Analyse dieses Objekts wahlweise im Binärbild oder im Originalbild. (Russ 2007) gibt eine Einführung in die Verwendung multipler Kriterien zur Binarisierung.

4.3 Konturverfolgung

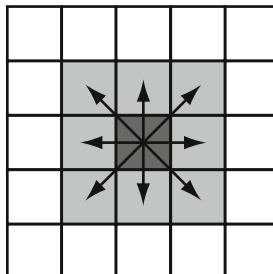
Nachdem im vorigen Abschnitt der erste Aspekt der Binärsegmentierung behandelt wurde, die Unterscheidung zwischen (logisch) weißen und schwarzen Bildbereichen, kommen wir nun zum zweiten Aspekt, der Erzeugung der Objektrepräsentation. Eine verbreitete Methode hierzu, die auch in den vorangegangenen Beispielen benutzt wurde, ist die *Konturverfolgung*. Unter der Kontur eines Objekts versteht man eine geschlossene Linie, die vollständig am Rand des Objekts

verläuft, also ausschließlich aus Randpunkten besteht. Diese Definition ist unmittelbar einleuchtend, aber keineswegs so einfach wie sie sich anhört, denn zunächst muss man einmal erklären was eigentlich ein Randpunkt ist und das führt uns auf den Begriff der *Nachbarschaft*.

4.3.1 Punktnachbarschaft



Vierernachbarschaft



Achternachbarschaft

Abbildung 4.7: Unterschiedliche Definitionen der Punktnachbarschaft

Es gibt in der digitalen Bildverarbeitung zwei gängige Möglichkeiten, die Nachbarn eines gegebenen Bildpunkts zu definieren: die Vierer- und die Achternachbarschaft, dargestellt in Abbildung 4.7. Bei der Vierernachbarschaft werden nur die orthogonal angrenzenden Punkte als Nachbarn eines Bildpunkts betrachtet, bei der Achternachbarschaft sind auch diagonale Verbindungen zugelassen.

An den Begriffen der Punktnachbarschaft und der Kontur, die dem Menschen so unmittelbar einleuchten, lässt sich die fundamentale Bedeutung der Diskretisierung des digitalen Bildes sehr gut veranschaulichen. Im Folgenden zeigen wir einige Erscheinungen, die durch die Diskretisierung in Verbindung mit den verschiedenen Nachbarschaftsdefinitionen entstehen, und durch ihren offensichtlichen Widerspruch zu menschlichen Wahrnehmungsgewohnheiten auch den erfahrenen Anwender industrieller Bildverarbeitungssysteme gelegentlich in Verwirrung stürzen – und den Programmierer eines solchen Systems vor unvorhergesehene Probleme stellen können.

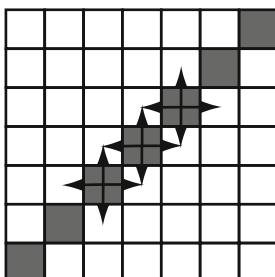


Abbildung 4.8: Diagonale, nicht zusammenhängend in Vierernachbarschaft

Die in 45° -Richtung verlaufende Aneinanderreihung schwarzer Punkte in Abbildung 4.8 ist für das menschliche Auge ganz offensichtlich eine Linie. Bei Verwendung der Vierernachbarschaft ist diese Punktreihe jedoch überhaupt nicht zusammenhängend, da keiner der schwarzen Punkte ein orthogonaler Nachbar eines der anderen ist. Es erscheint naheliegend, dem menschlichen Gefühl für zusammenhängende Linien durch Verwendung der Achternachbarschaft entgegenzukommen. Dies führt aber wiederum zu anderen Problemen, wie Abbildung 4.9 illustriert. Die Kontur des schwarzen Objekts in diesem Bild ist nur bei Verwendung der Achternachbarschaft vollständig zusammenhängend. Allerdings ist dann auch der Hintergrund zusammenhängend, da die weißen Bildpunkte ebenfalls über die Diagonalen verbunden sind. Hintergrund und Objekt durchdringen einander, der Hintergrund fließt praktisch durch die Objektkontur hindurch. Eine hierarchische Suche nach einem hellen Objekt in einem dunklen Objekt kann dann zu erheblichen Abgrenzungsschwierigkeiten führen.

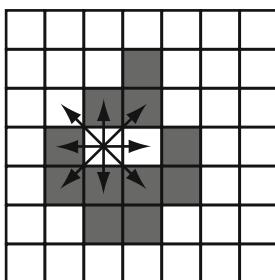


Abbildung 4.9: Durchdringung von Hintergrund und Objekt bei Verwendung der Achternachbarschaft

Schließlich sei noch erwähnt, dass auch die selbstverständliche Vorstellung einander schneidender Geraden im diskreten Raum der digitalen Bildverarbeitung ihre Einfachheit verliert. Die mathematische Definition des Schnittpunkts als des einzigen gemeinsamen Punkts zweier Geraden ist in der diskreten Welt des digitalen Bildes leider nicht anwendbar. Abbildung 4.10 zeigt, dass sich zwei Geraden in Achternachbarschaft ganz offensichtlich schneiden können, ohne einen einzigen Punkt gemeinsam zu haben.

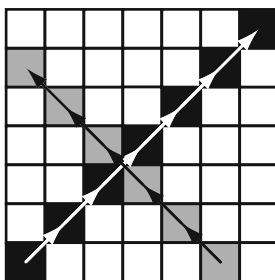


Abbildung 4.10: Geraden, die sich ohne gemeinsamen Punkt schneiden

4.3.2 Erzeugung von Objektkonturen

Das Prinzip der Erzeugung von Objekten mittels Konturverfolgung ist sehr einfach, einen entsprechenden Algorithmus findet man z. B. in (Parker 1994):

1. Im Suchbereich wird nach einem ersten Übergang zwischen Hintergrund und Objekt gesucht, also bei der Erzeugung schwarzer Objekte nach einem Übergang von (logisch) Weiß zu Schwarz, bei der Erzeugung weißer Objekte entsprechend umgekehrt.
2. Sobald der Übergang gefunden ist, wird der nächste Nachbar gleicher (logischer) Farbe in der entsprechenden Nachbarschaft gesucht. Als Nachbar zählt dabei nur ein Randpunkt, d. h. ein Punkt, der mindestens einen Punkt der anderen (logischen) Farbe als Nachbar hat, damit der Algorithmus nicht ins Objektinnere hineinläuft.
3. In derselben Weise wird die gesamte Kontur verfolgt, wobei man dies im Uhrzeigersinn oder entgegen dem Uhrzeigersinn tun kann (man muss sich nur für eine Richtung entscheiden und dabei bleiben).
4. Die Konturverfolgung ist beendet, wenn der Algorithmus wieder am Ausgangspunkt angekommen ist, d. h. dem Punkt, an dem der erste Übergang gefunden wurde.

Ein Hauptvorteil dieses Verfahrens gegenüber vielen anderen Methoden liegt darin, dass immer visuell zusammenhängende, geschlossene Objekte erzeugt werden. Im wirklichen Leben gibt es aber wie immer ein paar besondere Probleme, beispielsweise:

- Als Randpunkt zählen auch Punkte, die direkt am Bildrand liegen, denn sonst könnten keine Konturen gefunden werden, die am Bildrand entlang verlaufen.
- Dasselbe gilt für Punkte am Rand eines Suchbereichs, wenn die Suche auf den Bereich begrenzt werden soll.
- Wenn nicht in geometrisch einfachen Suchbereichen, wie Rechtecken, gesucht wird, sondern entsprechend der Äquivalenz von Arbeitsbereich und Objekt unseres Modells auch in Objekten, die bereits selbst durch eine Konturverfolgung erzeugt wurden, muss man darauf achten, dass in Abbildung 4.9 dargestellte Paradoxon abzufangen.
- Man muss verhindern, dass die Suche an isoliert aus dem Objekt herausragenden Punkten in sich selbst zurückläuft.

Diese Aufzählung ist natürlich keineswegs erschöpfend, sie soll nur einen Hinweis darauf geben, welchen Schwierigkeiten man sich im Einzelfall gegenüber sehen kann.

4.3.3 Konturdarstellung

Die nach dem obigen Verfahren erzeugte Kontur kann in ganz unterschiedlichen Datenstrukturen dargestellt werden. In der Bildverarbeitung hat sich die Darstellung in Form eines Kettencodes durchgesetzt, da sie eine Reihe von algorithmischen Vorteilen bietet. So kann man aus dem Kettencode sehr leicht Umfang und Ausdehnung des Objekts berechnen, ebenso die Krümmung und die erste Ableitung der Kontur. Auch eine Glättung der Kontur, um Rauscheinflüsse zu verringern, ist gut mit dem Kettencode durchzuführen.

Der Kettencode, nach (Freeman 1961) auch als Freeman-Code bezeichnet, stellt eine Kontur als Folge von Richtungsinformationen dar. Die Koordinate des Startpunkts wird separat gespeichert. Die Richtung der Verbindung zum nächsten Konturpunkt wird durch eine Zahl codiert, wie es in Abbildung 4.11 dargestellt ist¹.

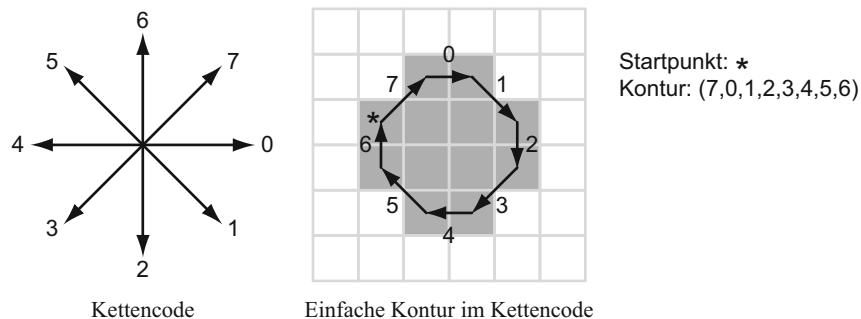


Abbildung 4.11: Kettencode zur Konturbeschreibung

4.4 Template Matching

Bei dem vorgestellten Segmentierungsverfahren aus Binarisierung und Konturverfolgung wurde das Objekt gewissermaßen bottom-up konstruiert. Es wurde eine Transformationen auf ikonischer Ebene vorgenommen, die Binarisierung, und dann auf der Grundlage des transformierten Bildes versucht das Objekt mit der Konturverfolgung zu erzeugen. Das bedeutet, dass zunächst einmal nur festgestellt wurde, ob überhaupt Kandidaten für *irgendwelche* Objekte vorhanden sind. Die Frage liegt nahe, ob man nicht gleich nach ganz bestimmten Objekten suchen könnte, denn dann bliebe einem die Arbeit erspart, aus all den möglichen Objekten die herauszusuchen, um die es eigentlich geht. Tatsächlich ist eine solche Vorge-

¹ Die Zuweisung der Nummern an die einzelnen Richtungen ist im Prinzip beliebig, die hier gezeigte ist aber eine weit verbreitete Konvention, die z. B. auch in (Parker 1994), (Nischwitz 2007), (Gonzalez und Woods 2008) und (Sonka et al. 2008) verwendet wird.

hensweise möglich, einfach ist sie aber auch nicht, denn man kann ja einem Computer nicht sagen, er möge nach einer Zündkerze oder einem Buchstaben im Bild suchen, weiß er doch überhaupt nicht, was diese Dinge sind.

In diesem Abschnitt stellen wir zunächst ein weit verbreitetes Verfahren zur Lokalisierung *bekannter* Objekte in einem Bild vor. Wir werden das Verfahren hier durchgängig bei seiner englischen Bezeichnung *Template Matching* nennen, da uns mögliche deutsche Übersetzungen, wie etwa *Schablonenvergleich*, zu ungewöhnlich erscheinen. Am Ende des Abschnitts werden wir weitere Verfahren skizzieren, die ebenfalls bekannte Objekte in einem Bild suchen.

4.4.1 Funktionsprinzip

Das Grundprinzip des Template Matching besteht darin, ein Bild des gesuchten Objekts über das aktuelle Bild zu schieben und an jeder Position ein Ähnlichkeitsmaß zu berechnen. Wenn eine Position einen hinreichenden Ähnlichkeitswert ergibt, wird davon ausgegangen, dass sich das gesuchte Objekt dort befindet. Dieses Verfahren lässt sich natürlich auf eine beliebige Anzahl gesuchter Objekte und eine beliebige Anzahl gesuchter Positionen erweitern – wobei es allerdings auch beliebig rechenaufwändig werden kann. Zu Optimierungsmöglichkeiten kommen wir gleich noch.

Das *Bild des Objekts* bezeichnet man als *Template*, *vergleichen* heißt auf Englisch *match*, so dass damit auch die Bezeichnung der Methode erklärt ist. Konkretisieren wir nun also die noch recht vage Beschreibung des Verfahrens.

Ähnlichkeitsmaß: Prinzipiell sind sehr verschiedene Ähnlichkeitsmaße verwendbar, aber das mit Abstand am weitesten verbreitete ist die Korrelation (Russ 2007, Sonka et al. 2008).

Der empirische Korrelationskoeffizient einer Reihe von Zahlenpaaren $(g_1, t_1), (g_2, t_2), \dots, (g_N, t_N)$ ist nach (Bronstein et al. 2005) durch Gleichung (4.1) gegeben. Darin bedeuten \bar{g} und \bar{t} die Mittelwerte der Komponenten g_i und t_i , σ_g und σ_t ihre Standardabweichungen².

$$r(g, t) = \frac{\sum_{i=1}^N (g_i - \bar{g})(t_i - \bar{t})}{(N-1)\sigma_g\sigma_t}. \quad (4.1)$$

Die Korrelation eines Bildausschnitts mit einem Template gleicher Größe kann ebenfalls nach dieser Formel berechnet werden, wenn man die Grauwerte des Templates und des Bildausschnitts in der oben gezeigten Form in einen Vektor von Zahlenpaaren schreibe. Die direkte Berechnung aus dem Bild heraus sieht formal etwas komplizierter aus, ist aber nur auf den ersten Blick verwirrend. Ab-

² Wir verwenden hier σ_g im, weil es üblicherweise auch mit der empirischen Standardabweichung assoziiert wird; strenggenommen sollte s verwendet werden.

bildung 4.12 möge die Verhältnisse verdeutlichen: Ein Template t der Breite b und der Höhe h liegt mit seiner linken oberen Ecke auf dem Punkt $(x, y) = (1, 1)$ des Bildes. Dann korrespondiert jeweils der Grauwert $t_{i,j}$ des Templates mit dem Grauwert $g_{x+i,y+j}$ des Bildes. Die Indizes laufen jeweils von 0 bis zu $b - 1$ für die x-Koordinate, $h - 1$ für die y-Koordinate. Der (gedachte) Vektor von Zahlenpaaren lautet folglich $(g_{x,y}, t_{0,0}), (g_{x+1,y}, t_{1,0}), \dots, (g_{x,y+h}, t_{0,h}), \dots, (g_{x+b-1,y+h-1}, t_{b-1,h-1})$. In Abbildung 4.12 ist zur besseren Übersicht nur die Korrespondenz der vier Eckpunkte eingezeichnet.

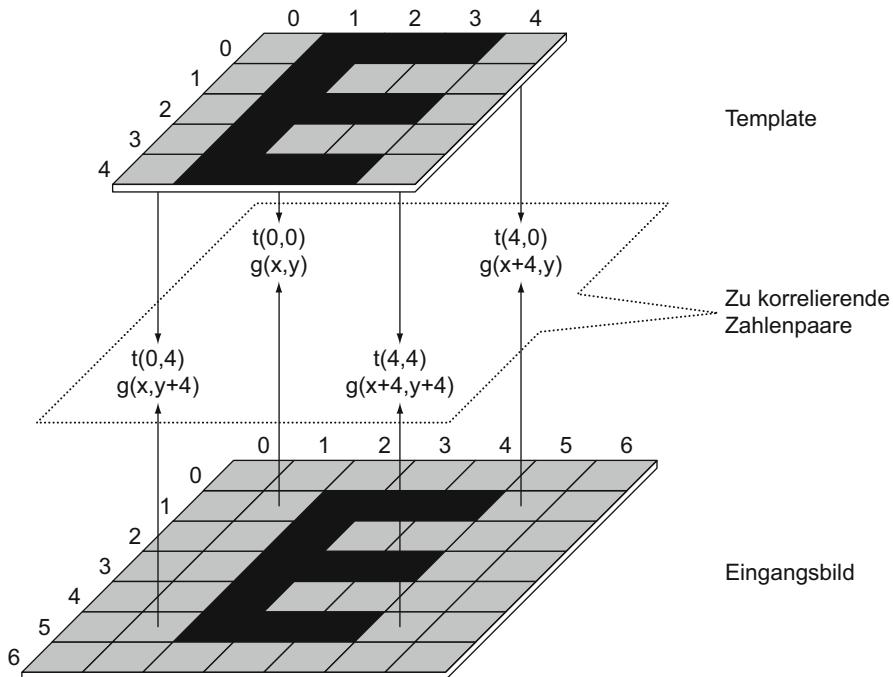


Abbildung 4.12: Prinzip des Template Matchings mit Korrelation

Die Summen für die Berechnung des Korrelationskoeffizienten laufen dann über beide Koordinaten. Mit der Abkürzung N für die Anzahl $b \cdot h$ der Punkte im Template lautet dann die Formel für die Korrelation wie in Gleichung (4.2).

$$r(x, y) = \frac{\sum_{i=0}^{b-1} \sum_{j=0}^{h-1} (g_{x+i,y+j} - \bar{g})(t_{i,j} - \bar{t})}{(N-1)\sigma_g \sigma_t}. \quad (4.2)$$

Die Mittelwerte und Standardabweichungen sind wie folgt zu berechnen:

$$\bar{g} = \frac{1}{N} \sum_{i=0}^{b-1} \sum_{j=0}^{h-1} g_{x+i, y+j},$$

$$\bar{t} = \frac{1}{N} \sum_{i=0}^{b-1} \sum_{j=0}^{h-1} t_{i,j},$$

$$\sigma_g = \sqrt{\frac{1}{N-1} \left(\sum_{i=0}^{b-1} \sum_{j=0}^{h-1} (g_{x+i, y+j} - \bar{g})^2 \right)},$$

$$\sigma_t = \sqrt{\frac{1}{N-1} \left(\sum_{i=0}^{b-1} \sum_{j=0}^{h-1} (t_{i,j} - \bar{t})^2 \right)}.$$

Die Berechnung der Korrelation zwischen Bildausschnitt und Template ist, wie man an diesen Formeln leicht sehen kann, zwar eine prinzipiell einfache und systematische Angelegenheit, aber ziemlich aufwändig. Man muss sich ja auch vor Augen halten, dass bei der einfachsten Form des Template Matchings, bei der das Template einfach Punkt für Punkt über das Bild verschoben wird, diese ganze Berechnung für jeden *Bildpunkt* durchgeführt werden muss (fast jeden; im Allgemeinen lässt man das Template nicht über die Bildränder hinausragen, hört also entsprechend der Templategröße etwas früher auf). In den obigen Formeln lassen sich nur die Werte \bar{t} und σ_t vorab berechnen, alle anderen müssen für jeden neuen Eckpunkt (x, y) neu ermittelt werden. In unserem Modell nach Abschnitt 1.5 werden zwar nur die definierten Arbeitsbereiche abgesucht, nicht das gesamte Bild, aber auch das bedeutet einen erheblichen Rechenaufwand, zumal man üblicherweise nicht bloß nach einem Typ Objekt sucht, sondern nach mehreren Klassen von Objekten und auch noch für jede Klasse zumeist mehrere Muster hat, um kleinere Abweichungen auszugleichen. Es liegt daher nahe, nach Optimierungsmöglichkeiten zu suchen, das ganze Verfahren ein bisschen intelligenter zu gestalten. Bevor wir uns jedoch damit befassen, wollen wir noch einige Eigenschaften des Verfahrens ansprechen, über die man sich vor der Anwendung im Klaren sein sollte.

Eigenschaften: Abbildung 4.13 zeigt ein Bild mit einigen einfachen Figuren. Der Klarheit halber wurden keine Zwischentöne verwendet, nur Weiß und Schwarz. Die Abbildung zeigt weiterhin die Korrelation dieses Bildes, wenn das gerade Kreuz, die zweite Figur von links in der oberen Reihe, als Template benutzt wird. Weiß entspricht hier einer Korrelation von eins, Schwarz einer Korrelation von minus eins, mittleres Grau einer Korrelation von null. Man erkennt Folgendes:

- Die um 45° gedrehte Figur weist keine stärkere Korrelation mit dem Template auf als die anderen Figuren.

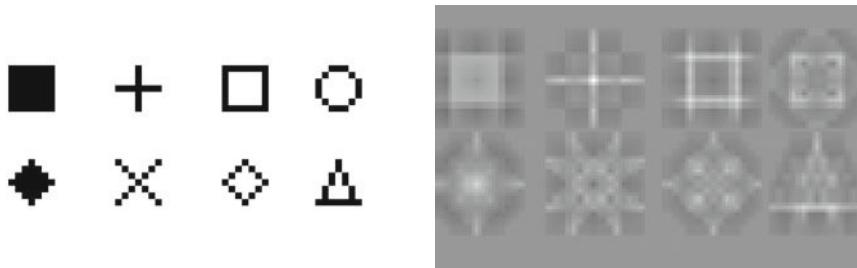


Abbildung 4.13: Einfache Figuren und ihre Korrelation mit dem geraden Kreuz als Template

- Nur an einer einzigen Stelle, wenn das Template genau über dem Ursprungsbjekt liegt, finden wir im Ergebnisbild einen weißen Punkt, also eine Korrelation von eins. Daneben fällt die Korrelation recht scharf ab.
- Es besteht eine gewisse Korrelation zwischen dem Template und den „falschen“ Figuren, am auffälligsten bei dem nicht gefüllten Quadrat, das durch seine langen, schmalen Außenkanten dem Kreuz je nach Matchingposition sehr ähnlich sein kann.
- Auch in der Umgebung einer Figur hat die Korrelation bereits einen sichtbaren Betrag. Das ist am deutlichsten bei dem ausgefüllten Quadrat zu beobachten, da dort die Überdeckung zwischen Template und Bildobjekt allmählich und kontinuierlich ansteigt. Dieser Effekt ist bei höher auflösenden Bildern und ausgedehnteren Figuren natürlich noch erheblich stärker.

Was lässt sich aus diesen Beobachtungen schließen? Template-Matching ist offensichtlich recht empfindlich gegenüber Verzerrungen oder Verdrehungen von Objekten, wie man an der schlechten Korrelation des geraden Kreuzes mit dem diagonalen erkennen kann. Natürlich ist das ein extremes Beispiel, sowohl von der Art der Objekte als auch von Art und Ausmaß der Veränderung her, aber wir haben bewusst ein so drastisches Beispiel gewählt, um diese Schwierigkeit deutlich zu machen.

Ein weiterer wichtiger Gesichtspunkt ist die Art der Objekte. Das Kreuz mit seinen schmalen Armen liefert nur in einer sehr engen Umgebung des Objekts eine nennenswerte Korrelation, während das ausgefüllte Quadrat durch die wesentlich größere Überdeckung schon viel früher hohe Korrelationskoeffizienten ergibt. Andererseits ist das Kreuz als Template selektiver, eben wegen der geringeren Überdeckung.

Besonders wichtig für die spätere Optimierung des Verfahrens ist aber die Beobachtung, dass die Korrelation sich nur selten sprunghaft ändert, sondern bereits in der Umgebung eines Objekts allmählich ansteigt. Das eröffnet die Möglichkeit, das Bild in größeren Schritten nach Kandidatenpositionen abzusuchen. Nur in deren Umgebung wird dann eine punktweise Suche durchgeführt.

Man kann also zusammenfassen, dass das Template Matching sich v. a. für flächige Objekte gut eignet, weniger gut für sehr filigrane Objekte, und dass es empfindlich auf geometrische Veränderungen reagiert.

4.4.2 Optimierungsmöglichkeiten

Das oben vorgestellte Basisverfahren zum Template Matching, bei dem an jedem Punkt des Bildes die vollständige Korrelation mit allen Punkten des Templates durchgeführt ist, führt zu einem enormen Rechenaufwand. Um die Berechnung zu beschleunigen, versucht man deshalb statt dem Basisverfahren geschickt konstruierte Algorithmen mit weniger Rechenaufwand zu verwenden.

Pyramiden suche: Die oben demonstrierten Eigenschaften des Template Matching geben schon die ersten Anhaltspunkte für eine Optimierung. Wir sahen, dass die Korrelation bereits in der Umgebung eines Objekts deutlich ansteigt, um so mehr, je flächiger das Objekt ist. Es wäre also nicht unbedingt nötig, das gesamte Bild Punkt für Punkt abzusuchen, man könnte das Template in größeren Sprüngen bewegen, und bräuchte dann nur in der Umgebung von Gebieten hoher Korrelation eine punktweise Suche durchzuführen. Diese Methode bezeichnet man als Pyramiden suchverfahren (engl. *coarse-to-fine-search*), da man sie in Form einer Pyramide aus Suchschritten darstellen oder auch in Software auf diese Weise implementieren kann.

Dieses Pyramidenverfahren erlaubt erhebliche Beschleunigungen. Wenn sich die Schrittweite für die erste Stufe von ein Pixel auf vier Pixel erhöhen lässt, dann dauert die erste Stufe nur noch 1/16 der Zeit. Natürlich kommt dann die Zeit für die Feinsuche hinzu, aber in vielen Fällen ist sie vergleichsweise vernachlässigbar, je nachdem wie signifikant die Korrelation verläuft und wie viele Positionen tatsächlich gefunden werden müssen.

Man muss allerdings betonen, dass die Möglichkeit, überhaupt eine Pyramiden suche durchzuführen, und der realisierbare Reduktionsfaktor von der Art der Muster abhängen, wie Abbildung 4.14 zeigt. Dort wurde, wie in Abbildung 4.13, die Suche mit dem geraden Kreuz durchgeführt, mit Schrittweiten von ein, zwei, drei und vier Pixeln. Bei einer Schrittweite von vier entsteht überhaupt keine signifikante Korrelation mehr in der Umgebung des Objekts, weil die Schrittweite größer ist als die Strukturbreite des Objekts.

Auf der anderen Seite muss die Einschränkung gemacht werden, dass dieses Musterbild ein extremes Beispiel ist. In realen Bildern verlaufen Korrelationen in der Regel weniger abrupt und bei den Bildauflösungen, die z.B. für sichere Schrifterkennung benötigt werden, ist schon eine deutliche Erhöhung der Schrittweite möglich. Das Grundprinzip bleibt jedoch bestehen: Je filigraner die Objekte sind und je stärker sie auf Kanten anstatt Flächen zur Erkennung angewiesen sind, desto weniger wird sich die Schrittweite erhöhen lassen.

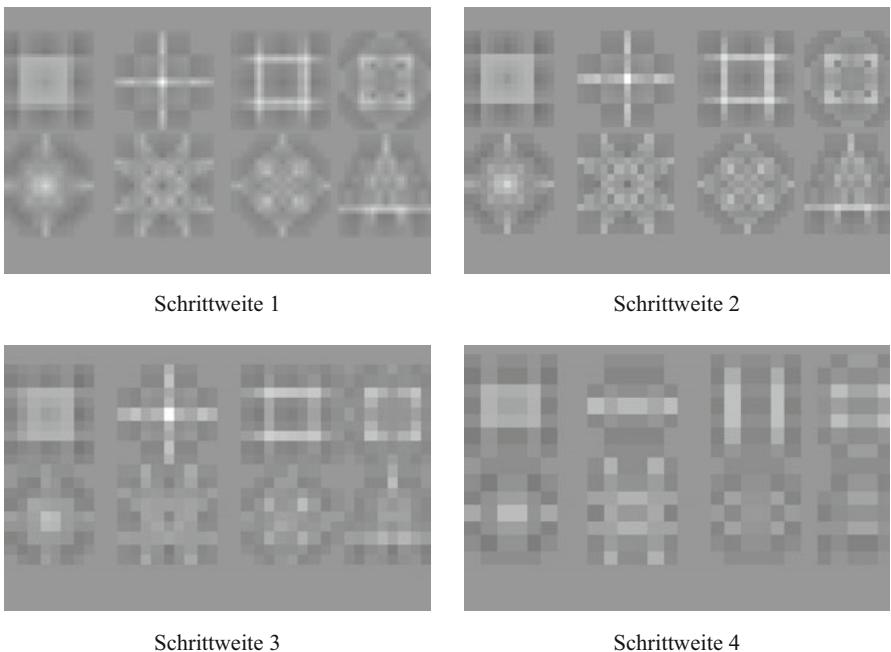


Abbildung 4.14: Suche nach geradem Kreuz mit verschiedenen Schrittweiten

Unterabtastung: Oben wurde eine Vergrößerung der Schrittweite, mit der das Template über das Bild bewegt wird, als eine erste Optimierungsmöglichkeit angeführt. Damit wird die Anzahl der Korrelationsvorgänge reduziert. Jede einzelne Korrelationsberechnung ist aber noch genauso aufwändig, wie es aus Gleichung (4.2) hervorgeht. Dieser Rechenaufwand lässt sich auf ähnliche Weise – und auf der Basis derselben Begründung des relativ stetigen Verlaufs der Korrelation – reduzieren, indem man nicht jeden Punkt des Templates verwendet, sondern z. B. nur jeden zweiten, dritten oder vierten. Der Effekt auf die Rechenzeit ist ähnlich, ebenso die Auswirkungen auf das Ergebnis und die Einschränkungen der Anwendbarkeit.

Es ist durchaus möglich, beide Methoden gleichzeitig zu benutzen. Dabei wird es im Allgemeinen nicht möglich sein, die Suchschrittweite ebenso stark zu erhöhen wie bei Benutzung aller Punkte des Templates, noch wird man das Template ebenso sparsam abtasten können wie bei einer Suchschrittweite von eins, aber damit ist keineswegs gesagt, dass sich die Wirkungen gegenseitig aufheben. Da die mögliche Suchschrittweite in erster Linie von den Eigenschaften des Bildes abhängt, das abgesucht wird, die Unterabtastung in erster Linie von den Eigenschaften des Templates, kann man sich in der Mitte treffen und eine Beschleunigung erreichen, die höher ist, als mit jeder der einzelnen Methoden möglich gewesen wäre. Abbildung 4.15 zeigt die Suchergebnisse mit einer Unterabtastung von zwei bei Suchschrittweiten von eins und zwei. Eine Unterabtastung von zwei bedeutet

eine Beschleunigung um etwa einen Faktor vier³, eine Suchschrittweite von zwei ebenfalls, so dass bei der Kombination eine Beschleunigung um einen Faktor 16 möglich ist. In Abbildung 4.14 war dagegen bei einer Schrittweite von vier Pixeln, was derselben Beschleunigung entspräche, keine Detektion mehr möglich.

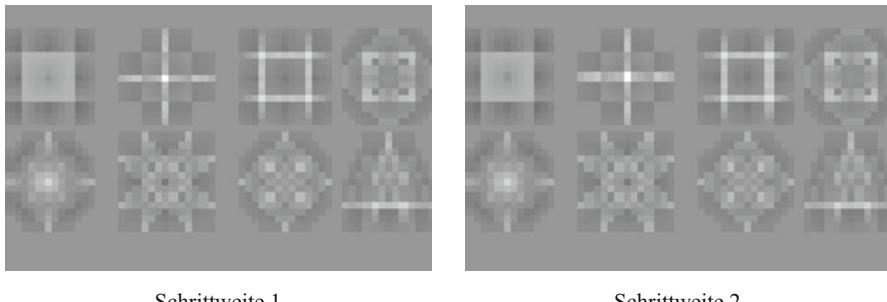


Abbildung 4.15: Suche nach geradem Kreuz mit Unterabtastung zweie

Optimierung der Punktpositionen: Bei der einfachen, äquidistanten Unterabtastung des Templates stößt man auf das Problem, dass die Punkte beliebig schlecht liegen können, beispielsweise unmittelbar auf einer Kante, was den Korrelationsvorgang bis auf ein Pixel von der Position abhängig machen kann. Mit einem auf diese Weise unterabgetasteten Template ist dann praktisch keine Pyramidensuche mehr möglich. Dieses Problem lässt sich beheben, indem man die Positionen der zu verwendenden Punkte nach entsprechenden Kriterien optimiert. Für diese Optimierung kommen vielfältige Verfahren in Frage, die wir hier nicht im Einzelnen aufführen können, z. B. selbstorganisierende neuronale Netze oder kombinatorische Optimierungsverfahren. Je nach Anwendung kann man dabei nur auf möglichst hohe Korrelation mit dem jeweiligen Template optimieren oder auch auf gute Abgrenzung zu anderen Templateklassen.

Als Beispiel zeigt Abbildung 4.16 die Korrelationspunkte für die Detektion einer Ziffer, die mittels Tampondruck auf Gummi aufgebracht wurde. Nicht nur war es in dieser Anwendung möglich, die Korrelationsgeschwindigkeit für dieses filigrane Muster um einen Faktor vier zu erhöhen, auch die Matching-Qualität, d.h. die mittlere Höhe der Korrelationskoeffizienten, konnte gesteigert werden, da die Punkte an aussagekräftigeren Stellen platziert wurden.

Abkürzung der Berechnung: Schließlich soll nicht unerwähnt bleiben, dass es noch eine ganz triviale Möglichkeit der Beschleunigung gibt, die allen oben genannten Verfahren gleichermaßen zugutekommt. Sie besteht darin, die Berechnung abzubrechen, sobald ein Misserfolg des Matchings offensichtlich wird. Im Allgemeinen gibt es sehr viel mehr Stellen im Bild, an denen das Matching erfolg-

³ Selbstverständlich erfolgt die Beschleunigung nicht linear zur Reduzierung der Punktzahl, da es immer einen gewissen, nicht reduzierbaren Overhead gibt, zum Vergleich der Methoden kann man aber näherungsweise davon ausgehen.

los bleibt, als solche, an denen tatsächlich etwas gefunden wird. Wenn man eine Möglichkeit findet, einen Misserfolg frühzeitig zu erkennen, kann man sich an ziemlich vielen Vergleichspunkten einen Teil der Berechnung sparen.

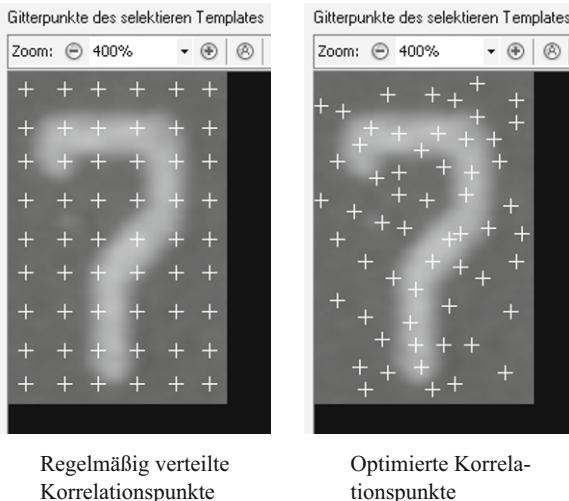


Abbildung 4.16: Optimierung von Korrelationspunkten

Die erste Möglichkeit hierzu ergibt sich daraus, dass in Gleichung (4.2) der Nenner immer positiv ist. Eine gute Kandidatenposition für die Anwesenheit eines Objekts zeichnet sich durch eine hohe positive Korrelation aus, d. h. wenn der Zähler des Korrelationskoeffizienten negativ wird, findet man an diese Stelle ganz sicher keine positive, geschweige denn eine hohe Korrelation und kann die Berechnung der Standardabweichung σ_g des Bildausschnitts unterlassen.

Eine zweite Möglichkeit geht davon aus, dass man üblicherweise eine gewisse Mindestqualität für das Matching verlangt. Normalerweise wird man Objekte mit einer Korrelation unterhalb einer bestimmten Schwelle (oft 0,5) zurückweisen. Nachdem man den Zähler des Korrelationskoeffizienten berechnet hat – und dieser positiv ist – kann man sofort berechnen, wie groß σ_g im Nenner höchstens werden darf, damit diese Schwelle noch erreicht wird. Daraus ergibt sich dann sehr einfach ein höchstzulässiger Wert für den Ausdruck

$$\sum_{i=0}^{b-1} \sum_{j=0}^{h-1} g_{x+i, y+j},$$

der bei der Berechnung der Standardabweichung zunächst einmal anfällt. Sobald während der Summation diese Schwelle überschritten wird, kann die Berechnung für diesen Punkt wieder abgebrochen werden, da man dort dann sicher kein Objekt der erwarteten Qualität vorfinden wird.

4.4.3 Bemerkungen zum Template Matching

Die obigen Ausführungen lassen das Template Matching vielleicht als ein außerordentlich empfindliches und aufwändiges Verfahren erscheinen, und vielleicht fragen Sie sich, wozu man es überhaupt einsetzen kann? Tatsächlich kann es bei entsprechender Konfiguration ausgesprochen leistungsfähig sein. Davon soll hier noch ein wenig die Rede sein.

Flexibilität: Im Gegensatz zur Binärsegmentierung können mit Template Matching auch Objekte segmentiert werden, die visuell nicht zusammenhängen. Das ist vor allem in der Schrifterkennung ein sehr typisches Problem. Einerseits gibt es Buchstaben, wie etwas das i, die von Natur aus nicht zusammenhängend sind. Andererseits gibt es, und das ist der wichtigere Fall, Beschriftungsmethoden, die stets nicht zusammenhängende Zeichen erzeugen. Dazu gehören die im Bereich der Beschriftungen von Verpackungen weit verbreitete Tintenstrahltechnik ebenso wie die Nadelprägetechnik, die häufig bei Fahrzeug- und Maschinenteilen angewandt wird, da derartige Beschriftungen wesentlich haltbarer sind als jede Art von Aufdruck.

Des weiteren müssen die Grenzen eines durch Template Matching gefundenen Objekts keineswegs im Bild visuell abgrenzbar sein (was je nach Anwendung von Vorteil oder Nachteil sein kann. Flächenberechnungen bei solchen Objekten sind wenig nutzbringend, da man nur die Fläche des Templates bestimmen kann).

Die „eingebaute“ Normierung des Korrelationskoeffizienten führt außerdem dazu, dass das Template Matching gegenüber Beleuchtungsschwankungen weniger empfindlich ist als die Binärsegmentierung – vorausgesetzt, die Helligkeitsdifferenz zwischen Objekt und Hintergrund kehrt ihre Richtung nicht um. Selbst dann aber kann relativ einfache Vorverarbeitung ein Matching wieder ermöglichen.

Programmhinweis: Voraussetzung für einen erfolgreichen Einsatz des Template Matching ist es, die vielen verschiedenen Optimierungsmöglichkeiten schnell und flexibel durchprobieren zu können. Die Template-Matching-Funktion von **NeuroCheck** ermöglicht es daher, alle oben genannten Optimierungsparameter (Pyramidenstufe, Unterabtastung, Punktoptimierung) mit verschiedenen Einstellungen durchzuführen.

Korrelation im Frequenzbereich: Eine Optimierungsmöglichkeit wurde bislang nicht erwähnt, da sie bei den typischen Anwendungen des Template Matching in industriellen Anwendungen weniger bedeutend ist, und zwar das Matching nach vorhergehender Fouriertransformation. Die komplizierte Korrelation reduziert sich im Frequenzbereich auf eine einfache Multiplikation, aber diese Optimierungsmethode lohnt sich nur, wenn das Template so groß ist, dass der Aufwand für die normale Korrelation den für die Fouriertransformation übersteigt.

Verallgemeinertes Template Matching: Die oben angesprochene Empfindlichkeit des Template Matchings gegenüber Verdrehungen und Verzerrungen der Objekte lässt sich reduzieren, indem für jedes Objekt mehrere leicht unterschiedliche

Templates gehalten werden. Diese Templates können entweder im Verlauf der Konfiguration einer Anwendung einzeln eingelernt oder automatisch während des Matchings aus einem Referenztemplate erzeugt werden. Dadurch lassen sich auch veränderte Objekte gut finden, allerdings steigt der Rechenaufwand linear mit der Anzahl der Templates.

Schließlich kann man auch dem Template Matching eine Filterung des Bildes vorangehen lassen, durch die dann z. B. größere Schrittweiten bei gleicher Erkennungssicherheiten möglich werden.

4.4.4 Kantenbasierte Objektlokalisierung

Wie bereits angekündigt gibt es neben dem Template Matching weitere Verfahren mit welchen bekannte Objekte in einem Bild gesucht werden können. Während Template Matching die Objekte im Grauwertbild sucht, suchen die Verfahren die hier vorgestellt werden, auf Kantenbildern. D. h. vor der Suche der Objekte wird das Bild zunächst mithilfe von Filteroperatoren, die in Kapitel 2 besprochen werden, transformiert. Abbildung 4.17 zeigt ein Beispiel für das Ergebnis einer Kantenextraktion mit einem relativ einfachen Operator.

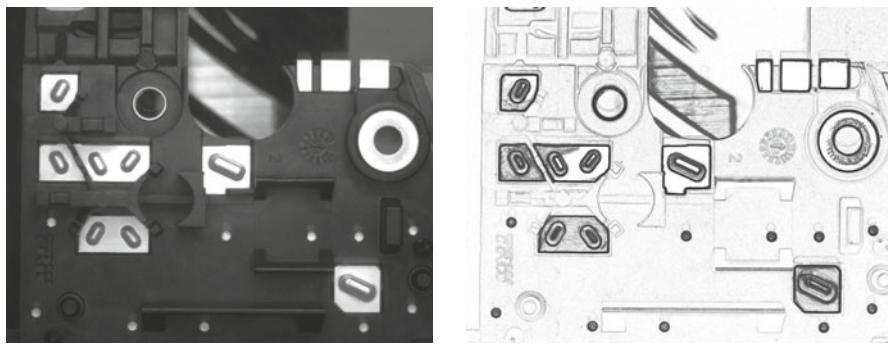


Abbildung 4.17: Kantenextraktion aus einem Grauwertbild

Es dürfte klar sein, dass die Bestimmung von Objekten aus einem solchen Kantenbild keineswegs trivial ist. Die Suche von Objekten in Kantenbildern hat aber auch die Vorteile, dass die Erzeugung von Kanten relativ unempfindlich gegenüber Helligkeitsänderung ist und dass Kantenbilder eine Richtungsinformation in Form der Gradientenrichtung enthalten.

Formbasiertes Matching: Das formbasierte Matching folgt demselben Prinzip wie Template Matching: Das Muster wird über das Bild geschoben und an jeder Position werden Muster und Bild verglichen. Im Unterschied zu Template Matching wird aber nicht die Korrelation als Maß für die Güte der Übereinstimmung

verwendet. (Steger et al. 2008) verwenden als Gütemaß die Summe der Skalarprodukte zwischen den Gradientenvektoren des Bildes und des Templates. Für jeden Punkt des Templates wird somit das Skalarprodukt des Gradientenvektors mit dem Gradientenvektor des korrespondierenden Bildpunkts berechnet. Die Summe der Skalarprodukte aller Template-Punkte ergibt das Gütemaß.

Das Gütemaß r für die Position (x, y) errechnet sich damit gemäß Formel 4.3, wobei zu beachten ist, dass g und t Vektoren sind.

$$r(x, y) = \frac{1}{b \cdot h} \sum_{i=0}^{b-1} \sum_{j=0}^{h-1} \langle \vec{g}_{x+i, y+j}, \vec{t}_{i,j} \rangle \quad 4.3$$

Das Skalarprodukt wird groß wenn die Gradienten in Template und Bild in dieselbe Richtung zeigen und jeweils einen großen Betrag haben, also eine deutliche Kante vorliegt. Wenn dies bei vielen Punkten des Templates der Fall ist, wird das Gütemaß für diese Position groß. An Stellen an denen das Muster nicht ins Bild passt, fallen viele Kantenpunkte des Templates auf Stellen im Bild an der Gradient klein ist. Damit wird auch das Skalarprodukt der Gradient klein werden. An Stellen an denen sich Kanten in Bild und Template schneiden, sind zwar die Beträge der Gradienten groß, aber da die Gradienten in unterschiedliche Richtungen verlaufen, bleibt das Skalarprodukt ebenfalls klein.

Insgesamt ist dieses Verfahren relativ unempfindlich gegenüber der teilweisen Überdeckung des Objekts und Störungen im Bild. Um das Verfahren einsetzen zu können, müssen viele der Optimierungen die man beim Template Matching anwenden kann, auch beim formbasierten Matching verwendet werden. Eine ausführliche Darstellung des Verfahrens einschließlich verschiedener Optimierungen wird in (Steger et al. 2008) gegeben.

Generalisierte Hough-Transformation: Die generalisierte Hough-Transformation arbeitet genau umgekehrt wie Template Matching. Während beim Template Matching für jede Position im Bild geprüft wird, wie gut das Template mit dem Bild übereinstimmt, wird bei der generalisierten Hough-Transformation für jeden Kantenpunkt geprüft, zu welcher Objektposition er passt. Wenn viele Kantenpunkte zu einer Objektposition passen, dann ist dort ein Objekt gefunden worden.

Für die Objektsuche werden nur Kantenpunkte betrachtet, was im einfachsten Fall alle Punkte sind, deren Gradientenbetrag über einem bestimmten Schwellwert liegt. Vor Beginn der Suche wird ein Referenzpunkt im Muster festgelegt und eine sogenannte Referenztabelle (R-table) berechnet. Die Referenztabelle enthält für jeden Kantenpunkt des Musters den Abstand zwischen Kantenpunkt und dem Referenzpunkt und die Richtung vom Kantenpunkt zum Referenzpunkt. Die Richtung zum Referenzpunkt wird dabei relativ zur Gradientenrichtung des Kantenpunkts angegeben.

Bei der eigentlichen Objektsuche wird für jeden Kantenpunkt angenommen er sei ein Punkt des Objekts, für den es einen Eintrag in der Referenztabelle gibt. Aus den Daten aus der Tabelle, der Lage des Punkts und der Richtung des Gradienten

lässt sich bestimmen, wo das Objekt liegen würde. Für jeden Punkt im Bild spielt man alle Einträge der Referenztabelle durch und zählt mit, wie oft jede Objektposition auftritt. Wenn eine Objektposition häufig auftritt, gibt es somit viele Kanten im Bild die zu dem gesuchten Objekt an genau dieser Stellen passen. An dieser Stelle befindet sich somit ein Objekt im Bild.

Das Grundprinzip der generalisierten Hough-Transformation wurde von (Ballard 1981) entwickelt, allerdings sind eine ganze Reihe von Optimierungen notwendig um das Verfahren in der industriellen Bildverarbeitung einsetzen zu können. Eine ausführliche Darstellung des Verfahrens findet sich in (Sonka et al. 2008), und in (Steger et al. 2008) finden sich eine Reihe von Optimierungen des Verfahrens.

Geometrisches Matching: Als letztes Verfahren wollen wir das geometrische Matching vorstellen. Bei diesem Verfahren werden einfache geometrische Formen, meistens Geradenstücke, im Bild gesucht. Für die gefundenen Geradenstücke wird dann geprüft, ob sie zu dem gesuchten Objekt zusammen gesetzt werden können. Der in Abbildung 4.18 dargestellte Schraubenschlüssel hat z. B. sechs Geradenstücke: zwei bilden den Griff und die anderen vier bilden das Maul des Schraubenschlüssels.

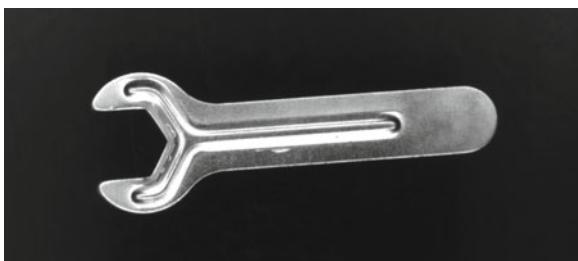


Abbildung 4.18: Objekt für geometrisches Matching

Bei der Objektsuche werden in einem ersten Schritt sämtliche Geradenstücke im Bild gesucht. Im zweiten Schritt wird für jede Kombination aus sechs dieser Stücke geprüft, ob sie die gleiche Anordnung wie die sechs Geradenstücke des Musters haben. Wenn sie die gleiche Anordnung haben, hat man einen Schlüssel gefunden, ansonsten muss man die nächste Anordnung ausprobieren. Aufgrund der vielfältigen Kombinationsmöglichkeiten der gefundenen Geraden ist der Rechenaufwand bei diesem naiven Ansatz natürlich immens.

Mit entsprechenden Optimierungen ist dieses Verfahren dennoch gut geeignet um Objekte mit langen Kantenstücken schnell zu finden. Für filigrane Objekte mit vielen kleinen Kanten sind die vorher beschriebenen Verfahren des formbasierten Matchings und der generalisierten Hough-Transformation besser geeignet. Weitere Details zum geometrischen Matching kann man in (Steger et al. 2008) finden.

4.5 Kantendetektion

In den bisherigen Abschnitten dieses Kapitels ging es vor allem darum, Objekte überhaupt zu finden. In diesem Abschnitt soll es darum gehen, die Position von Kanten, von denen man schon ungefähr weiß wo sie sind, genau zu bestimmen. Die Genauigkeit der Positionsbestimmung ist erforderlich, da diese Kanten oft die Grundlage für eine nachfolgende Vermessung sind.

4.5.1 Kantenantastung in industriellen Bildszenen

Bei dieser Fragestellung ist bereits in etwa bekannt, wo sich die gesuchte Kante befinden muss, es geht also „nur“ noch um die genaue Lokalisierung. Anstelle der im Abschnitt 2.3.4 beschriebenen Umwandlung des gesamten Bildes in ein Kantenbild kann man für solche Anwendungen den Ort der Kante direkt mittels eines Kantenmodells ermitteln. Neben des erheblich geringeren Rechenaufwandes hat diese Methode den Vorteil, dass sich die Beschaffenheit der zu detektierenden Kanten wesentlich einfacher vorschreiben lässt als bei der Verwendung von Kantenfiltern.

Die Suche nach Kanten mit einem Kantenmodell erfolgt normalerweise auf Suchlinien in einer bestimmten Richtung. Die Kriterien für die Detektierung einer Kante ergeben sich aus dem Grauwertprofil entlang dieser Suchlinie. Ein typisches Kantenmodell hat folgende Parameter:

Richtung: Man unterscheidet zwei Kantenrichtungen, ansteigende und abfallende Kanten. Von einer abfallenden Kante spricht man, wenn das Grauwertprofil von hell nach dunkel verläuft, im umgekehrten Fall hat man eine ansteigende Kante vor sich.

Kantenhöhe: Zur Detektierung einer gültigen Kante fordert man, dass eine bestimmte Mindestdifferenz der Grauwerte entlang der Suchlinie auftritt. Dies bezeichnet man als die Kantenhöhe.

Kantenlänge: Die Kantenlänge gibt an, auf welcher Strecke die durch die Kantenhöhe festgelegte Mindestdifferenz der Grauwerte auftreten muss.

Abbildung 4.19 veranschaulicht diese Parameter anhand einer gültigen und einer ungültigen abfallenden Kante. Das Grauwertprofil der ungültigen Kante erreicht zwar sogar eine größere absolute Grauwertdifferenz, benötigt dazu aber mehr Bildpunkte als der Längenparameter zulässt. Die Kante ist also weniger steil. Durch die Kombination dieser beiden Parameter erreicht man eine große Flexibilität bei der Unterscheidung signifikanter Objektkanten von allmählichen Grauwertübergängen, deren Ursache beispielsweise in Beleuchtungsgradienten liegen kann.

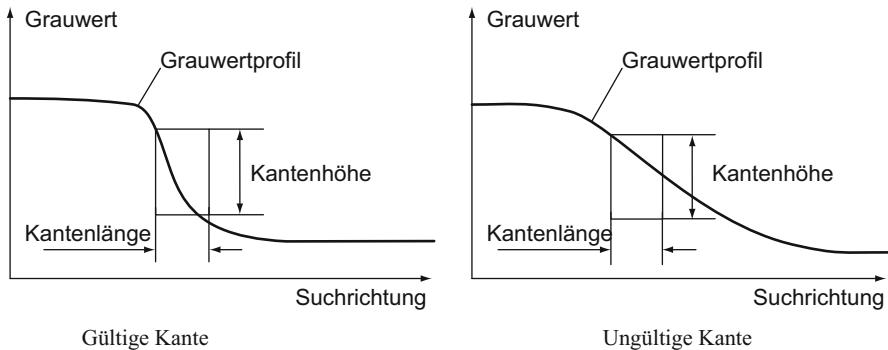


Abbildung 4.19: Detektierung von Kanten mit einem Kantenmodell

Programmhinweis: Der Prüfschritt *Kanten erzeugen* in **NeuroCheck** bedient sich eines derartigen Kantenmodells zur subpixelgenauen Lokalisierung von Kantenpositionen.

4.5.2 Subpixelgenaue Kantenlokalisierung

In Abschnitt 1.4.2 wurde erwähnt, dass die durch Rasterung und Quantisierung verloren gegangene Information in begrenztem Umfang wiedergewonnen werden kann. Die Sensorelemente einer Kamera haben eine endliche Ausdehnung. Sie nehmen also Licht aus einem gewissen Bereich der Bildszene auf. Im späteren Grauwertbild verläuft jede sichtbare Kante genau auf der Grenze zwischen zwei Bildpunkten. Der „wahre“ Ort der Kante in der ursprünglichen Bildszene wird jedoch praktisch niemals mit der Grenze zweier Sensorelemente zusammenfallen, sondern irgendwo innerhalb eines Bildpunkts verlaufen. Jedes Bildelement erhält daher Licht von beiden Seiten der Kante, mal mehr aus dem hellen, mal mehr aus dem dunklen Bereich. Im Grauwertbild äußert sich dies durch einen graduellen Übergang zwischen dem hellen und dunklen Bereich, wie es in Abbildung 4.20 für ein schwarzes Quadrat dargestellt ist. Abbildung 4.21 zeigt einen Querschnitt durch das Quadrat mit dem allmählichen Kantenübergang. Nach einer Interpolation des Kantenprofils lässt sich der wahrscheinlichste wahre Ort der Kante am Wendepunkt der Interpolationsfunktion lokalisieren, der in der Abbildung 4.21 ebenfalls eingezeichnet ist.

Prinzipiell ist es natürlich möglich, ein ganzes Bild so zu interpolieren, aber sowohl der Zeit- als auch der Speicherplatzaufwand verbieten dies in der Regel und es ist für eine genaue Vermessung auch gar nicht erforderlich. Man beschränkt sich daher darauf, den Ort solcher Kanten subpixelgenau festzustellen, die für die Vermessungsaufgabe relevant sind. Nachdem der Ort des Objekts bekannt ist, sei es durch Vorwissen oder vorhergehende Segmentierung, kann eine Kantenantastung durchgeführt werden. In der Umgebung jeder Kante wird dann

die beschriebene Interpolation vorgenommen und der wahre Ort der Kante subpixelgenau ermittelt. Der berechnete Abstand ist dann ebenfalls subpixelgenau.

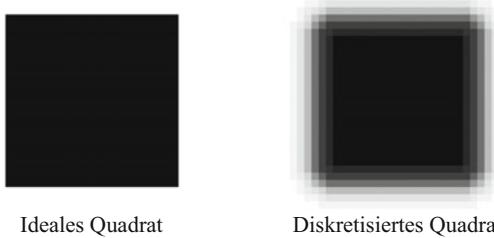


Abbildung 4.20: Allmählicher Kantenübergang infolge von Orts- und Helligkeitsdiskretisierung

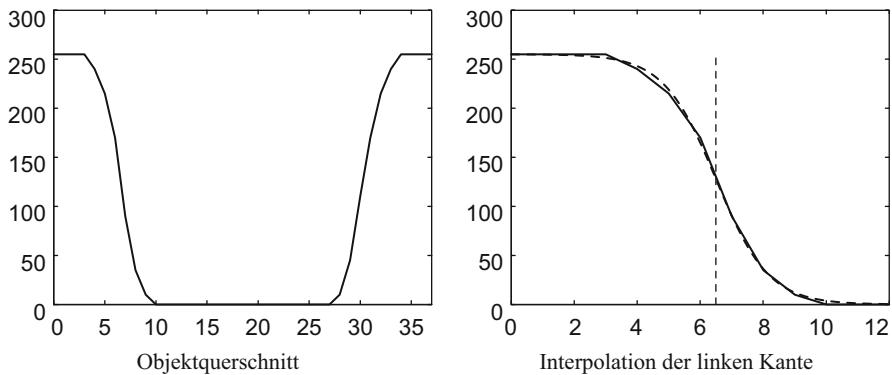


Abbildung 4.21: Prinzip der subpixelgenauen Lokalisierung einer Kante

Mit derartigen Subpixelalgorithmen lässt sich unter Laborbedingungen eine Genauigkeit von 1/20 Pixel erreichen, allerdings setzt dies außerordentliche Sorgfalt bei der Bildaufnahme voraus, was in den Kapiteln 7 und 8 über Vermessung und Beleuchtung noch ausführlich behandelt wird. Im Fertigungseinsatz kann man selbst bei hohem Aufwand eine derartige Stabilität nicht gewährleisten, so dass unter Feldbedingungen mit einer Genauigkeit von 1/5 bis höchstens 1/10 Pixel zu rechnen ist. Die absolute Genauigkeit ist natürlich vom Blickfeld und der Bildauflösung abhängig – und im Extremfall von der Lichtwellenlänge, denn mit entsprechenden Kameras und Optiken kann man ohne weiteres in den Bereich von Beugungseffekten vordringen.

Programmhinweis: Mit **NeuroCheck** lassen sich Kantenpositionen auf zwei Wege subpixelgenau feststellen:

- Der Prüfschritt *Arbeitsbereiche binär erzeugen* verfügt über die Option „Nur Kante“, die z. B. bei der Erzeugung der Gewindekonturen in Abschnitt 1.6 verwendet wurde. Dabei wird für den Ort, an dem die Suchlinie die Kante im Binärbild schneidet, eine Subpixelberechnung durchgeführt. Alle Maße, die

sich auf diesen Punkt beziehen, können mit Subpixelgenauigkeit erzeugt werden.

- Der Prüfschritt *Kanten erzeugen* wurde schon bei der Erklärung von Kantenmodellen erwähnt. Er verwendet keine Binarisierung als Ausgangspunkt, sondern detektiert signifikante Grauwertsprünge entlang einer Suchlinie. Der Ort der Kante wird dann durch Interpolation subpixelgenau bestimmt.

4.6 Zusammenfassung

Segmentierung: Hierunter versteht man die Isolierung von Objekten aus der Bildszene für die weitere Analyse.

Arbeitsbereich: Arbeitsbereiche, auch als *Area of Interest* oder *Region of Interest* bezeichnet, grenzen den Bildbereich ein, der zu analysieren ist. Man kann auf diese Weise Vorwissen über die Position von Objekten in die Verarbeitung einbringen und den Rechenaufwand reduzieren, da nicht das gesamte Bild verarbeitet werden muss.

Objekt: Unter einem Objekt versteht man einen segmentierten Bildbereich, der mit einem realen Objekt in der Bildszene identifiziert werden kann. Im Rahmen unseres Vorgehensmodells sind Objekte und Arbeitsbereiche weitgehend austauschbar. Das ermöglicht einerseits die direkte Analyse fest definierter Bereiche ohne Segmentierung, andererseits die hierarchische Suche innerhalb von Objekten.

Hierarchische Suche: In technischen Anwendungen kommt es häufig vor, dass Objekte innerhalb von Objekten geprüft werden müssen, beispielsweise Schrauben in einem Gehäuse. In solchen Fällen ist es vorteilhaft, innerhalb eines gefundenen Objekts direkt nach weiteren Objekten suchen zu können. In diesem Beispiel würde dann das segmentierte Objekt „Gehäuse“ als Arbeitsbereich für die Suche nach den Schrauben verwendet.

Histogramm: Tabelle der Häufigkeiten der einzelnen Grauwerte in einem Bild oder Bildbereich, meist als Balkendiagramm dargestellt. Eine Histogrammanalyse kann zur Bestimmung optimaler Schwellwerte für die Binärsegmentierung verwendet werden.

Binärsegmentierung: Bei der Binärsegmentierung erfolgt die Unterscheidung zwischen Objekten und Hintergrund anhand der absoluten Helligkeit. Die Grauwertbereiche von Objekten und Hintergrund werden durch die *Binärschwelle* getrennt.

Bestimmung von Binärschwellen: Die Schwellwerte zur Trennung von Objekten und Hintergrund bei der Binärsegmentierung können auf bestimmte Werte gesetzt oder aus einer Analyse der Helligkeitseigenschaften von Bildbereichen bestimmt

werden. Eine häufig verwendete Methode ist die Histogrammanalyse, bei der der Schwellwert zwischen signifikanten Maxima der Grauwertverteilung gesetzt wird.

Lokale Binärschwellen: In vielen Fällen weist ein Bild in verschiedenen Bereichen unterschiedliche Helligkeitseigenschaften auf, so dass lokal unterschiedliche Schwellwerte für die Binärsegmentierung erforderlich sind.

Bereichsbinarisierung: Bei der Bereichsbinarisierung liegen die zulässigen Grauwerte für Objekte (oder den Hintergrund) nicht einfach oberhalb oder unterhalb einer einzigen Schwelle, sondern in einem bestimmten Bereich. Auf diese Weise lassen sich z. B. gleichzeitig helle und dunkle Objekte auf einem Hintergrund mittlerer Helligkeit segmentieren.

Punktnachbarschaft: Man unterscheidet zwei Arten von Punktnachbarschaften. Bei der Vierernachbarschaft werden nur die waagerecht und senkrecht angrenzenden Bildpunkte als Nachbarn betrachtet, bei der Achternachbarschaft auch die diagonal angrenzenden Punkte. Beide Möglichkeiten haben spezifische Vor- und Nachteile für die Beschreibung von Objekten.

Konturverfolgung: Die Erzeugung von Objekten mittels Konturverfolgung beruht auf der Binärsegmentierung. Ausgehend von einem Randpunkt eines Objekts bewegt man sich gemäß der verwendeten Nachbarschaft zum nächsten Randpunkt. Dies wird solange fortgesetzt, bis man wieder am Ausgangspunkt angekommen ist. Dadurch werden immer geschlossene Objekte erzeugt.

Konturdarstellung: Durch Konturverfolgung gefundene Objektbegrenzungen werden häufig in einem Kettencode dargestellt, der jede mögliche Bewegungsrichtung von einem Konturpunkt zum nächsten durch einen Richtungswert angibt.

Template Matching: Beim Template Matching werden Objekte durch direkten Vergleich von Bildausschnitten mit gespeicherten Mustern (Templates) detektiert. Als Maß für die Ähnlichkeit von Bild und Template dient im Allgemeinen der Korrelationskoeffizient. Das Verfahren ist rechenaufwändig und empfindlich gegenüber Verdrehungen der Objekte, ermöglicht aber die Segmentierung nicht visuell zusammenhängender Objekte. Dies ist mit Konturverfolgung nicht möglich. Außerdem ist der Korrelationskoeffizient weitgehend unabhängig von Beleuchtungsschwankungen.

Pyramidensuche: Das Template-Matching-Verfahren lässt sich beschleunigen, indem man zunächst mit einer Schrittweite von mehr als einem Bildpunkt sucht, dann in der Umgebung von Punkten hoher Korrelation eine genauere Lokalisierung durchführt.

Templateoptimierung: Eine weitere Beschleunigungsmöglichkeit für das Template-Matching-Verfahren besteht darin, die Anzahl der für die Korrelation verwendeten Punkte zu reduzieren. Eine Optimierung der Punktpositionen durch kombinatorische Optimierungsverfahren oder selbstorganisierende neuronale Netze ist dabei einer einfachen rasterförmigen Anordnung überlegen.

Kantenbasierte Objektlokalisierung: Durch entsprechende Filterungen lässt sich ein Bild in ein Kantenbild umwandeln. Für die Lokalisierung von Objekten in einem solchen Kantenbild gibt es verschiedene Methoden wie z. B. formbasiertes Matching, generalisierte Hough-Transformation oder geometrisches Matching.

Kantenanastung: Hierunter versteht man die Detektierung signifikanter Objektkanten auf einer Suchlinie anhand eines Kantenmodells, das eine Mindesthöhe und Mindeststeilheit der Grauwertänderung entlang dieser Suchlinie verlangt.

Subpixelgenaue Kantendetektion: Sichtbare Kanten können im digitalen Bild nur genau auf der Grenze von Bildpunkten verlaufen. Dies entspricht im Allgemeinen nicht dem wahren Ort der Kante in der Bildszene. Durch Interpolation des Grauwertverlaufes lässt sich diese Kantenposition mit einer Genauigkeit besser als ein Bildpunkt bestimmen. Voraussetzung hierfür sind aber sehr stabile Beleuchtungs- und Bildaufnahmeverhältnisse.

5. Kennzeichnungsidentifikation

Die Identifikation von Werkstücken, Produkten, Materialien ist ein unverzichtbarer Bestandteil der automatisierten Fertigung, sei es zur Lagerhaltung, Fertigungssteuerung oder Qualitätskontrolle. Es gibt eine Vielzahl von Kennzeichnungsmethoden, optische wie nichtoptische. Magnetische Datenträger sind ein einfaches Beispiel für eine nichtoptische Kennzeichnungstechnik, die vielfach in der Fertigungssteuerung eingesetzt wird. Diese nichtoptischen Kennzeichnungen sind für automatische Systeme sogar leichter identifizierbar als optische. Es gibt jedoch einige Aspekte, die für optische Kennzeichnungen sprechen:

- Nichtoptische Kennzeichnungen sind normalerweise vom Menschen nicht identifizierbar, während optische so ausgelegt werden können, dass sie Menschen und automatischen Systemen gleichermaßen verständlich sind.
- Nichtoptische Kennzeichnungen erfordern im Allgemeinen Zusatzeinrichtungen, die mit dem Werkstück transportiert werden müssen, während optische Kennzeichnungen direkt am Werkstück aufgebracht werden können.
- Nichtoptische Kennzeichnungen können oftmals nur am Werkstückträger angebracht werden, nicht direkt am Prüfteil. Damit ist aber noch nicht sichergestellt, dass sich im Werkstückträger auch tatsächlich das Teil befindet, das von der Kennzeichnung angegeben wurde. Dies stellt ein besonderes Sicherheitsrisiko dar: Wenn der Werkstückträger und seine Kennzeichnung der Fertigungssteuerung dienen und das transportierte Teil nicht der Kennzeichnung entspricht, bleibt eine Fehlmontage unbemerkt.

Es gibt also gute Gründe, optische Kennzeichnungen zu verwenden. In diesem Kapitel werden wir daher die wichtigsten Kennzeichnungsmethoden, Barcode, DataMatrix-Code und Klarschrift, besprechen und außerdem ein Beispiel dafür geben, dass sich auch beliebige andere Codierungen mit den gleichen Methoden identifizieren lassen.

5.1 Barcode-Identifikation

Barcodes sind durch ihre Verwendung im Einzelhandel eine der bekanntesten Methoden zur Kennzeichnung von Teilen jeder Art. Sie sind zwar nur dann vom Menschen lesbar, wenn, wie es im Handel ja üblich ist, die Klarschrift zusätzlich

abgedruckt wird¹, aber dafür sind sie relativ leicht und kostengünstig maschinell identifizierbar. In der Regel wird man dazu allerdings kein Bildverarbeitungssystem benutzen, sondern einen auf diese Aufgabe spezialisierten Scanner. Es gibt jedoch durchaus Fälle, in denen die Barcode-Identifikation mittels eines Bildverarbeitungssystems sinnvoll ist, z. B. wenn

- die Barcode-Identifikation nur eine Teilaufgabe der eigentlichen Prüfung darstellt
- die Kontrastverhältnisse schlecht sind
- nicht standardisierte Barcodes vorkommen
- kreisförmig aufgebrachte Barcodes zu lesen sind
- gebogene, gedrehte oder zerrissene Barcodes erkannt werden müssen

Prinzipiell kann ein Bildverarbeitungssystem einen Barcode nach dem Vorgehensmodell aus Abschnitt 1.5 erkennen, aber üblicherweise wird man nicht versuchen, eine Segmentierung einzelner Objekte durchzuführen und diese dann zu einem Barcode zusammenzusetzen. Statt dessen verwendet man spezielle Algorithmen, die innerhalb eines Suchbereiches direkt nach den charakteristischen Helligkeitsprofilen eines Barcodes suchen. Diese Algorithmen bringen uns im Hinblick auf die sonstigen Anwendungen der Bildverarbeitung, die wir hier besprechen wollen, keine besonderen Erkenntnisse, daher werden wir es bei einigen kurzen Bemerkungen zur Algorithmik belassen, dann Anwendungsbeispiele und weiterführende Literatur vorstellen.

5.1.1 Prinzip der Barcodeerkennung auf Grauwertbasis

Das Prinzip der Barcode-Identifikation ist relativ einfach: Ein definierter Arbeitsbereich wird zeilenweise durchsucht. Die Folge der hellen und dunklen Stücke des Grauwertprofils jeder Zeile wird mit den definierten Charakteristika des gesuchten Barcodes verglichen, um festzustellen, welche Barcodezeichen hier vorliegen.

Natürlich lässt sich dieses Verfahren ohne weiteres verallgemeinern. Man kann den Arbeitsbereich ebenso gut spaltenweise absuchen, um senkrechte Barcodes zu lesen, oder schräg für rotierte Barcodes. Es ist allerdings von Vorteil, bei rotierten Barcodes tatsächlich den Bildausschnitt mit einem Interpolationsverfahren zu drehen, um Unstetigkeiten zu vermeiden. Dabei kann man sich der Methoden zur Bestimmung der Orientierung bedienen, die in Abschnitt 3.3.4 vorgestellt wurden, wie wir später noch sehen werden.

Damit ist das Prinzip vollständig erklärt. Für denjenigen, der tatsächlich einen solchen Algorithmus entwickeln muss, stellt sich das Problem in der Praxis natürlich weit diffiziler dar. Er kann sich im Allgemeinen nicht auf konstante Helligkeitsstufen entlang der Zeilen verlassen, muss also entscheiden, ob eine Stufe

¹ Es soll Menschen geben, die Barcodes lesen können, es dürfte aber zwecklos sein, diese Fähigkeit in einer Stellenanzeige zu fordern.

noch signifikant ist oder nicht. Er wird sich auch nicht auf gleich bleibende Breiten der Striche und Zwischenräume verlassen können, z. B. bei Barcodes, die auf gekrümmten Flächen, wie etwa Flaschen, aufgebracht wurden. Das Lesen derartiger Barcodes erfordert eine dynamische Anpassung der Modulbreiten, die aber wieder zu anderen Unsicherheitsfaktoren führt, weil unter Umständen ungültige Module dann nicht mehr erkannt werden. Daher ist in einem solchen Fall oft eine spezielle Vorverarbeitung angebrachter, die das Bild passend entzerrt – vorausgesetzt das entsprechende Vorwissen ist vorhanden. Andere Probleme sind der Umgang mit Bildstörungen, Fehlstellen und Kratzern im Barcode usw.

Man sieht, dass auch auf diesem so abgeschlossen und beherrscht erscheinenden Gebiet die Entwicklung robuster Leseverfahren keineswegs einfach ist, wenn trotz Störungen verschiedenster Art zuverlässige Erkennungsleistungen erbracht werden sollen. Wir wollen hier nicht ausführlich auf die Entwicklung von Barcode-Lesealgorithmen eingehen, die aufgrund ihrer hohen Spezialisierung wenig zum Verständnis anderer Bildverarbeitungsanwendungen beitragen. Ausführliche Informationen über Barcodes und Matrixcodes finden Sie z. B. in (Lenk 2005) oder (<http://www.adams1.com>).

5.1.2 Barcodetypen

Dieser Abschnitt stellt einige der gängigen Barcodes vor, die von den meisten Bildverarbeitungssystemen identifiziert werden können.

EAN-Codes: Die EAN-Codes werden europaweit für die Kennzeichnung von Einzelhandelsprodukten verwendet. Es gibt sie 8- und 13-stellig, wobei die 13-stellige Form deutlich häufiger ist. Der Abbildung 5.1 zeigt beide Formen. Beachten Sie die verlängerten Balken an beiden Enden und in der Mitte des 13-stelligen Codes. Die EAN-Codes haben eine feste Vorschrift zur Prüfsummenberechnung. Insgesamt bestehen die EAN-Codes aus vier verschiedenen, ineinander verschachtelten Modulbreiten, was ihre Identifikation relativ schwierig macht.



Code EAN8



Code EAN13

Abbildung 5.1: Barcodetypen EAN8 und EAN13

UP-Codes: Die Universal Product Codes (abgekürzt UP-Codes) in Abbildung 5.2 sind den EAN-Codes eng verwandt, allerdings fehlt oft die Verlängerung der seitlichen Balken. Tatsächlich kann jeder EAN-Scanner auch UP-Codes lesen. Sie werden in den USA in ähnlicher Weise verwendet wie die EAN-Codes in Europa.

Der lange zwölfstellige Code wird als UPC-A bezeichnet, der kurze achtstellige als UPC-E.



Code UPC-E



Code UPC-A

Abbildung 5.2: Barcodetypen UPC-A und UPC-E

Code 2/5 Interleaved: Der Code kann ausschließlich Zahlen darstellen und optional eine Prüfsumme enthalten. Im Gegensatz zu den UP- und EAN-Codes ist bei diesem Code nicht festgelegt, wie viele Zeichen er enthält, nur dass die Zeichenzahl immer gerade ist. Der Code ist platzsparend, da jeweils zwei Zeichen in den Strichen und Zwischenräumen ineinander verschränkt sind (deshwegen muss auch die Zeichenzahl immer gerade sein). Im Gegensatz zu den EAN- und UP-Codes hat dieser Code nur zwei Modulbreiten, daher ist er sehr robust identifizierbar und wird gern in industrieller Umgebung eingesetzt.



Abbildung 5.3: Code 2/5 Interleaved

Code 39 und PZN: Der Code 39 ist einer der wenigen Barcodes, die auch Buchstaben darstellen können. Aufgrund seines umfangreichen Zeichenvorrates benötigt er vergleichsweise viel Platz und wird deshalb meist nur dort eingesetzt, wo tatsächlich Buchstaben benötigt werden. Der Code kann beliebig viele Zeichen enthalten. Wegen dieser Flexibilität in Format und Zeichensatz wird er oft zur Identifikation von Teilen zwischen verschiedenen Firmen benutzt.

Der Pharma-Zentralnummer-Code (abgekürzt PZN-Code) ist Standard auf deutschen Arzneimittelpackungen. Es handelt sich hierbei tatsächlich um einen Code 39 mit einigen Zusatzvorschriften. Er besteht grundsätzlich aus einem Minuszeichen, sechs Ziffern und einer Prüfsumme.



Abbildung 5.4: Code 39 und PZN

5.1.3 Beispiele für industrielle Barcode-Identifikation

Barcode auf Metall: In industrieller Umgebung sind Barcodes oftmals nicht von so hoher Qualität wie man das vom Einzelhandel gewohnt ist – und selbst dort erlebt man ja oft genug das Versagen der Scanner. Besonders wenn ein Barcode direkt auf Metall aufgebracht wird, sind Beschädigungen häufig. Abbildung 5.5 zeigt zwei auf Metall gedruckte Barcodes vom Typ 2/5 Interleaved. In diesem Fall lässt sich auch der durch einen Kratzer beschädigte Code noch identifizieren, da die Beschädigung senkrecht zu den Strichen des Codes orientiert ist. Eine Möglichkeit zum Ausgleich einer solchen Beschädigung besteht darin, übereinander liegende Zeilen zu integrieren, vorausgesetzt der Barcode ist genau waagerecht orientiert, denn schon bei einer geringen Drehung führt eine solche Integration über mehrere Zeilen zu einer Verschmierung der Barcodekanten. In diesem Fall könnte man Teilstücke der Zeilen zusammensetzen, um zumindest eine vollständig lesbare Zeile zu erhalten.

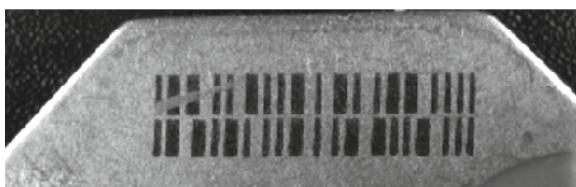


Abbildung 5.5: Beschädigter Barcode auf Metalloberfläche

In jedem Fall führt aber eine Beschädigung zu einer Minderung der Erkennungssicherheit und der Lesegeschwindigkeit, da bei geringer Qualität des Barcodes entsprechend aufwändige Algorithmen erforderlich sind, die vorhandenen Informationsbruchstücke sinnvoll zusammenzusetzen. In diesem Zusammenhang ist vielleicht noch interessant, dass die sogenannte *Erstleserate* von Scannern weit unter 100 % liegt. Was an Scannerkassen im Einzelhandel nur ein Ärgernis darstellt, kann in industriellen Anwendungen zum Problem werden, da jeder zusätzliche erforderliche Leseversuch Zeit kostet. Auch benötigen Scanner in der Regel stärkere Kontraste zwischen den hellen und dunklen Bereichen eines Barcodes als

ein Bildverarbeitungssystem, das selbst bei Kontrasten von nur fünf Prozent noch einwandfreie Leseergebnisse liefern kann.

Programmhinweis: Sie finden dieses Beispiel im Download im Verzeichnis BeispieleIdentifikation. Beachten Sie die hoch eingestellte Glättung im Prüfschritt *Barcode identifizieren* und die Tatsache, dass jede einzelne Zeile gescannt wird, nicht nur jede fünfte oder sogar zehnte Zeile, wie man es zur Beschleunigung bei qualitativ hochwertigen Barcodes tun kann.

Zweistufige Barcodeerkennung: Ausreichende Auflösung des Barcodes ist von großer Wichtigkeit für die Erkennung. Im Allgemeinen wird man davon ausgehen müssen, dass eine dünne Linie ebenso wie ein dünner Zwischenraum wenigstens drei Bildpunkte breit sein sollte, wenn mit Störungen zu rechnen ist, besser vier bis fünf, da die Kamerasensorik ansonsten nicht in der Lage ist, ausreichend klare Helligkeitsübergänge abzubilden. Das bedeutet andererseits, dass lange Barcodes, wie der oben angesprochene Code 39, eine erhebliche Bildauflösung verlangen. Zehn Zeichen im Code 39 erfordern bei einer Mindestlinienbreite von drei Pixeln 570 Bildpunkte, womit man sich schon fast an der Bildgröße von Kameras mit niedriger Auflösung befindet, und dabei sind die vorgeschriebenen und für eine sichere Detektierung erforderlichen sogenannten Ruhezonen links und rechts des Codes noch unberücksichtigt.

Bei Anwendungen im Bereich der Lagerhaltung verwendet man oft ein zweistufiges Vorgehen mit einer sogenannte Schwenk-Neigekopf-Zoomkamera (abgekürzt SNZ-Kamera). Wenn beispielsweise ein auf einem Behälter oder Container aufgebrachter Barcode gelesen werden soll, kennt man die Stelle, an der der Barcode aufgebracht ist, normalerweise nicht genau. Zunächst muss also der gesamte Behälter nach dem Barcodelabel abgesucht werden. Dazu muss das Bildfeld der Schwenk-Neigekopf-Zoomkamera die ganze Behälterfront enthalten. Bei dieser Auflösung ist es aber völlig ausgeschlossen, den Barcode zu lesen, weshalb er zunächst nur als Objekt gesucht wird. Aus den Koordinaten dieses Objekts wird dann eine Einstellung für die Schwenk-Neigekopf-Zoomkamera berechnet. Wenn die Kamera die berechnete Einstellung angefahren hat, wird der Barcode fast formatfüllend abgebildet, so dass er problemlos gelesen werden kann.

Gedrehte Barcodes: Zum Abschluss wollen wir hier noch Hinweise darauf geben, wie sich gedrehte Barcodes identifizieren lassen. Abbildung 5.6 zeigt denselben Barcode in drei verschiedenen Winkeln. Es handelt sich dabei um einen zu Lagerhaltungszwecken auf einem Bohrer aufgebrachten zweistelligen Code 2/5. Bei der leichten Verdrehung im zweiten Bild ist noch eine einwandfreie Lesung möglich, die stärkere Verdrehung im dritten Bild führt hingegen zu Fehlidentifikationen.

Prinzipiell kann man dieses Problem lösen, indem man die Vorzugsrichtung des Barcodes feststellt und dann den Scan in der entsprechenden Richtung durchführt. Dabei stehen allerdings nur die unmittelbaren Pixelinformationen zur Verfügung, wodurch je nach Auflösung die Übergänge mehr oder weniger stark verfälscht werden können. Stattdessen kann man auch aufgrund der Vorzugsrichtung

des Barcodes einen Bildausschnitt rotieren und dabei die Grauwerte interpolieren. Danach ist der Barcode einwandfrei lesbar.

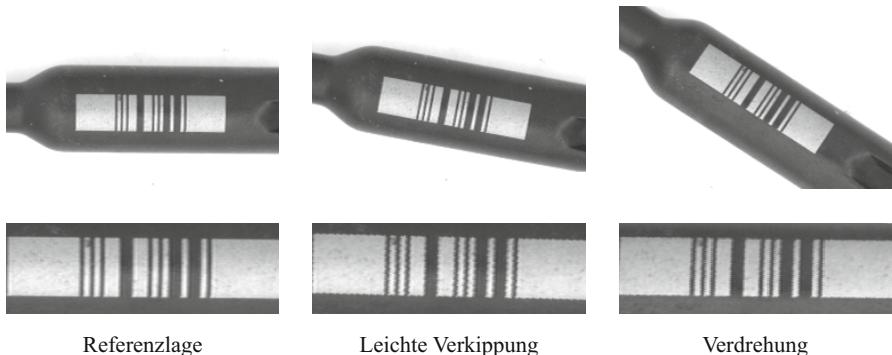


Abbildung 5.6: Barcode in verschiedenen Rotationswinkeln, original und nach Orientierungsbestimmung zurückgedreht

Zur Bestimmung der Vorzugsrichtung kann man gut die Methoden zur Positions-korrektur verwenden, die in Abschnitt 3.3.4 vorgestellt wurden. Zunächst wird dazu der gesamte Bohrer gesucht, danach auf diesem alle hellen Objekte ausreichen-de Größe (beachten Sie, dass hier tatsächlich der Hintergrund des Barcodes gedruckt wurde, nicht die eigentlichen Striche, die genau der Bohreroberfläche entsprechen, also auch nicht als Objekte von dieser segmentiert werden können). Auf diese Weise entstehen zehn Objekte, die, als zusammenhängende Gruppe betrachtet, eine sehr genau Positions- und Rotationsbestimmung ermöglichen. Man sieht leichte Diskretisierungseffekte nach der Drehung des Bildes, die aber die Erkennbarkeit nicht beeinträchtigen.

Programmhinweis: Sie finden dieses Beispiel im Download im Verzeichnis Beispiele\Identifikation. Die Drehung des Bildausschnitts mit dem Barcode erfolgt durch Definition einer Suchlinie, die genau in der Mitte des Barcodes verläuft und mit dem Prüfschritt *Arbeitsbereiche positionieren* der aktuellen Ausrichtung angepasst wird. Danach kann mit dem Prüfschritt *Arbeitsbereich abwickeln* ein Streifen beiderseits dieser Suchlinie herausgeschnitten und in die Horizontale gedreht werden. Beachten Sie in dem Prüfschritt *Arbeitsbereich abwickeln* die Einstellung „Feste Länge“. Bildpunkte sind in der Diagonalen um $\sqrt{2}$ länger als in der Horizontalen, daher würde das Ergebnisbild je nach Winkel unterschiedlich groß, was die weitere Verarbeitung verkomplizierte. Aus diesem Grund bietet der Prüfschritt die Möglichkeit, auf eine vorgegebene Länge zu skalieren.

5.1.4 Zweidimensionale Codes

Zweidimensionale Codes sind natürlich streng genommen keine Barcodes, weil sie nicht aus einer linearen Folge von Strichen und Zwischenräumen bestehen. Sie

beruhen jedoch auf derselben Grundidee, einer geometrischen Anordnung heller und dunkler Flächen, und manche Typen haben sich direkt aus bestimmten „linearen“ Barcodes entwickelt. Daher behandeln wir sie hier im Zusammenhang mit Barcodes.

Die Bedeutung zweidimensionaler Codes in der industriellen Fertigung und vor allem der Logistik nimmt stetig zu, da sich auf kleinerem Raum eine weit größere Informationsmenge codieren lässt als mit linearen Barcodes. Viele zweidimensionale Codes bieten außerdem mathematisch ausgefeilte Methoden der Fehlerkorrektur, so dass sich auch beschädigte Codes noch einwandfrei identifizieren lassen.

Einer der verbreitetsten Typen von Matrixcodes ist der von der Internationale Organisation für Normung (kurz ISO) als ISO/IEC 16022 standardisierte Data-Matrix-Code ECC-200. Das ECC-200 dient zur Unterscheidung von nicht normierten, veralteten DataMatrix-Code Varianten. Der Code besteht aus einer meist quadratischen, gelegentlich auch rechteckigen, Anordnung heller und dunkler Quadrate mit zwei durchgezogenen und zwei unterbrochenen Rändern, wie es Abbildung 5.7 zeigt. Jedes der Quadrate – mit Ausnahme der vier Randlinien – repräsentiert ein Codebit. Die acht Bits jedes Codebytes sind über die gesamte Fläche des Codes verteilt, damit nicht durch eine Beschädigung ein Byte komplett ausfällt.

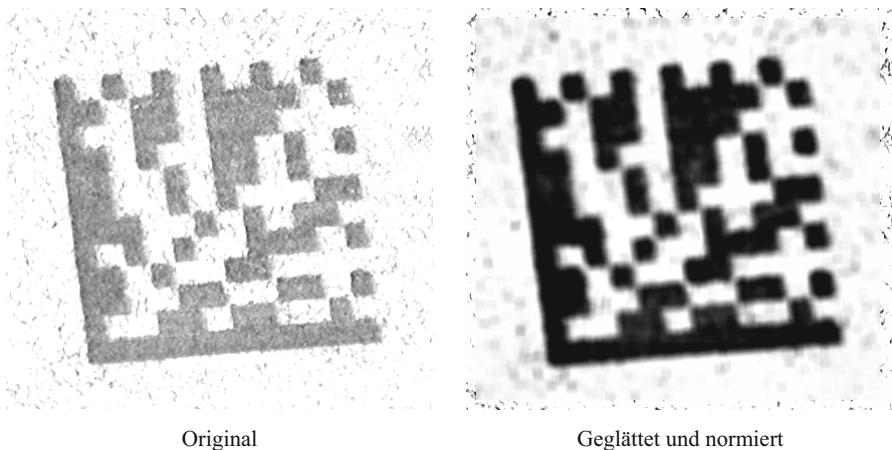


Abbildung 5.7: DataMatrix-Code auf Aluminiumguss

Je nach Codegröße sind etwa 40 bis 60 % der Bytes Prüfinformation, die restlichen Nutzinformation. Als Berechnungsverfahren für die Prüfinformation wird der Reed-Solomon-Algorithmus verwendet. Es gibt eine Reihe standardisierter Codierungen, mit denen ein umfangreicher Zeichenvorrat dargestellt und gleichzeitig eine Datenkompression erreicht werden kann. Die höchste Kapazität ergibt sich bei rein numerischen Informationen, wobei in jedem Nutzbyte des Codes zwei Ziffern codiert werden. Auch bei textuellen Informationen ist es jedoch

durch festgelegte Kompressionsverfahren möglich, mehr Zeichen darzustellen als der Code Nutzbytes enthält.

Abbildung 5.7 zeigt einen in Aluminiumguss lasergravierten DataMatrix-Code der Größe 12×12 (der Rand wird bei der Größenangabe gewöhnlich mitgezählt). Damit entsteht ein Code-Bereich von $10 \times 10 = 100$ Bit oder 12 Byte. Davon sind fünf Nutzbytes und sieben Korrekturbytes, womit sich eine Maximalkapazität von zehn Ziffern ergibt.

Das Bild zeigt eine für Aluminiumgussteile typische Hintergrundstruktur, die zu erheblichen Ausfransungen der Kanten des Codes führt. Am besten kommt der Code hier bei einer Hellfeldbeleuchtung zur Geltung, da die durch die Laserbehandlung veränderten Oberflächenstrukturen der Codemodule das Licht unterschiedlich gut in Richtung der Kamera reflektieren. Der Kontrast bleibt jedoch relativ schwach.

Der Code ist bei dieser Bildqualität durchaus identifizierbar. Durch einige Vorverarbeitungsschritte lässt sich die Zuverlässigkeit und Stabilität jedoch noch erhöhen. Eine Glättung des Bildes sorgt dafür, dass die Oberflächenstruktur nicht zu Unterbrechungen innerhalb der Codemodule führt, also hellen Einschlüssen in dunklen Quadranten und umgekehrt. Danach kann durch eine Normierung des Bildes in Form einer Histogrammspreizung eine Kontrastverbesserung erreicht werden. Es ist vorteilhaft, die Vorverarbeitung in dieser Reihenfolge vorzunehmen, da die Glättung Rauschen unterdrückt, das sonst durch die Normierung verstärkt würde. Abbildung 5.7 zeigt den Code vor und nach den beiden Bearbeitungsschritten.

Nach dieser Vorverarbeitung kann der Code ohne weiteres stabil auch bei schwankender Oberflächenbeschaffenheit identifiziert werden. Ähnlich wie bei linearen Barcodes ist auch hier eine Segmentierung und Weiterverarbeitung einzelner Module als Objekte nicht sinnvoll, dazu sind die einzelnen Schritte des Algorithmus zu spezialisiert. Daher wird man diese Decodierung gewöhnlich in einer Funktion kapseln. Tabelle 5.1 zeigt die Realisierung dieser Prüfanwendung mit **NeuroCheck**-Prüfschritten.

Beispielbeschreibung 5.1. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 5.1.4 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Gerichtetes Auflicht als Hellfeldbeleuchtung (Kamera in Richtung der Totalreflexion des einfallenden Lichts platziert)
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, ein Rechteck, das den DataMatrix-Code großzügig umschließt. 3. <i>Bildbereiche glätten</i> 4. <i>Bild normieren</i> 5. <i>DataMatrix-Code identifizieren</i>

5.2 Klarschrifterkennung

Die Erkennung normaler, für den Menschen lesbarer Schrift ist eine der anspruchsvollsten Aufgaben der industriellen Bildverarbeitung, sowohl hinsichtlich der „eigentlichen“ Verarbeitung des Bildes, die zur Isolierung der einzelnen Zeichen im Bild führt, als auch hinsichtlich der dann in unserem Modell folgenden Bewertung der Objekte, die hier in der Erkennung jedes Objekts als eines bestimmten Zeichens besteht.

Die verfügbare wissenschaftliche Literatur zu diesem Thema entspricht der Schwierigkeit der Aufgabe, sie ist praktisch nicht zu überblicken. Allerdings befasst sich ein Großteil dieser Literatur mit Handschrifterkennung, die in der industriellen Fertigung praktisch keine Rolle spielt. Gegenüber der Handschrifterkennung hat man bei gedruckten oder sonst maschinell erzeugten Schriften den Vorteil, dass die Form der Zeichen stabiler ist. Dennoch kann das Erscheinungsbild der Zeichen je nach Aufbringungsmethode auch innerhalb einer Fertigungscharge erheblich variieren. Solche Variationen können beispielsweise durch Verschleiß von Druckklischees oder Prägenadeln entstehen oder auch einfach aufgrund von Unterschieden in der Beschaffenheit der beschrifteten Oberfläche.

Die Hauptschwierigkeit besteht bei industriellen Beschriftungen daher häufig in der Isolierung der Zeichen von der Teileoberfläche. Dieser Schritt ist bei der Handschrifterkennung oder der Dokumentenverarbeitung typischerweise einfacher, da sich dort die Schrift meist mit gutem Kontrast von einem homogenen Hintergrund wie Papier abhebt. Die Identifikation der gefundenen Zeichen ist dann aufgrund der konstanteren Form oft einfacher als bei Handschrift, vor allem aus zwei Gründen:

- Maschinell aufgebrachte Zeichen werden separat geschrieben und sind entweder nicht kursiv oder haben konstante Neigungswinkel, so dass sie leichter von einander zu trennen sind als handgeschriebene Zeichen (völlig problemlos ist das trotzdem nicht, wie wir in Abschnitt 5.3 sehen werden).
- Maschinell aufgebrachte Zeichen stammen gewöhnlich aus einem einzigen Zeichensatz, so dass das Beschreibungsverfahren weniger Variationen erfassen muss als bei der außerordentlichen Variabilität der Handschriften vieler Menschen.

Dieser unterschiedliche Schwerpunkt gegenüber der Handschrifterkennung wird sich an den nachfolgenden Beispielen deutlich zeigen, bei denen der Positionierung und Segmentierung eine sehr wichtige Rolle zukommt.

5.2.1 Lasergravierte Schrift auf einem IC

Abbildung 5.8 zeigt eine Platine für ein Steuergerät, das in der Kfz-Elektronik eingesetzt wird. Eine der Prüfaufgaben bestand darin, sicherzustellen, dass an der

mit einem Pfeil markierten Stelle der richtige IC eingesetzt wurde. Da es sich um ein sicherheitsrelevantes Bauteil handelt, muss hier eine 100 %-Prüfung erfolgen.

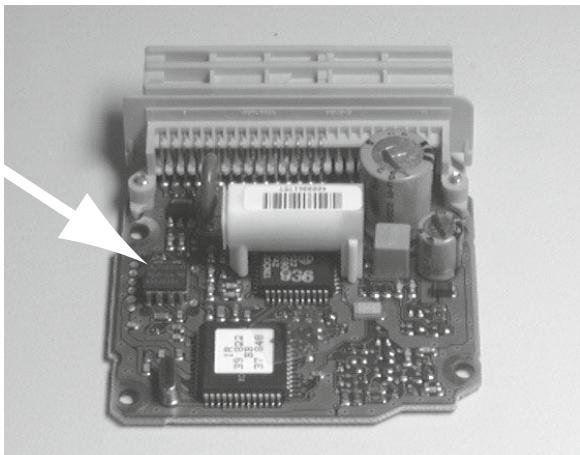


Abbildung 5.8: Platine aus der Kfz-Elektronik mit zu prüfendem IC (mit freundlicher Genehmigung der Robert Bosch GmbH, Stuttgart)

Die Schrift wird auf dem nicht einmal fingernagelgroßen IC mit einem Laser aufgebracht. Der IC ist schwarz, die Schrift ein dunkles, fast orangefarbenes Gelb. Der Kontrast ist daher nicht besonders gut, so dass eine starke, auf den IC konzentrierte Beleuchtung erforderlich ist.

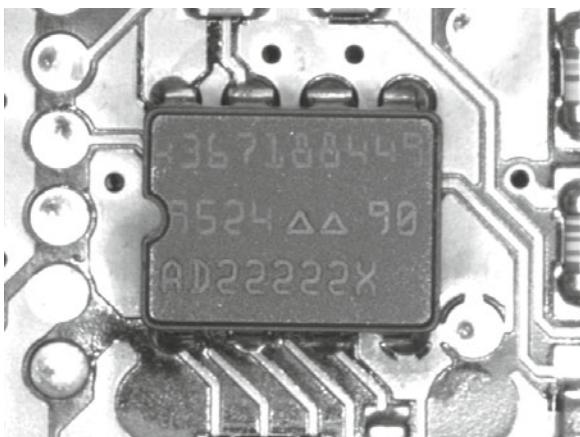


Abbildung 5.9: IC unter Ringlicht

Um die Zeichen deutlich hervortreten zu lassen, muss das Licht direkt in die Kamera zurückreflektiert werden, so dass sich eine ringförmige Beleuchtung anbietet. Damit entsteht aber das Problem starker Reflexionen, sowohl vom IC selbst

als auch von der Platine, und eines so geringen Kontrastes zwischen IC und Beschriftung, dass eine Segmentierung praktisch unmöglich wird, wie Abbildung 5.9 zeigt. Die Lösung dieses Problems ist in Abbildung 5.10 dargestellt: eine kreuzpolarisierte Beleuchtung. Das Licht aus der Ringleuchte fällt durch einen Polarisationsfilter, wodurch alle direkten Reflexionen Licht zurücksenden, das in einer einheitlichen Polarisationsebene schwingt. Ein um 90° gedrehter Polarisationsfilter vor der Kamera lässt dieses Licht nicht durch, so dass ein sehr klares, reflexionsfreies Bild entsteht, wie in Abbildung 5.11.

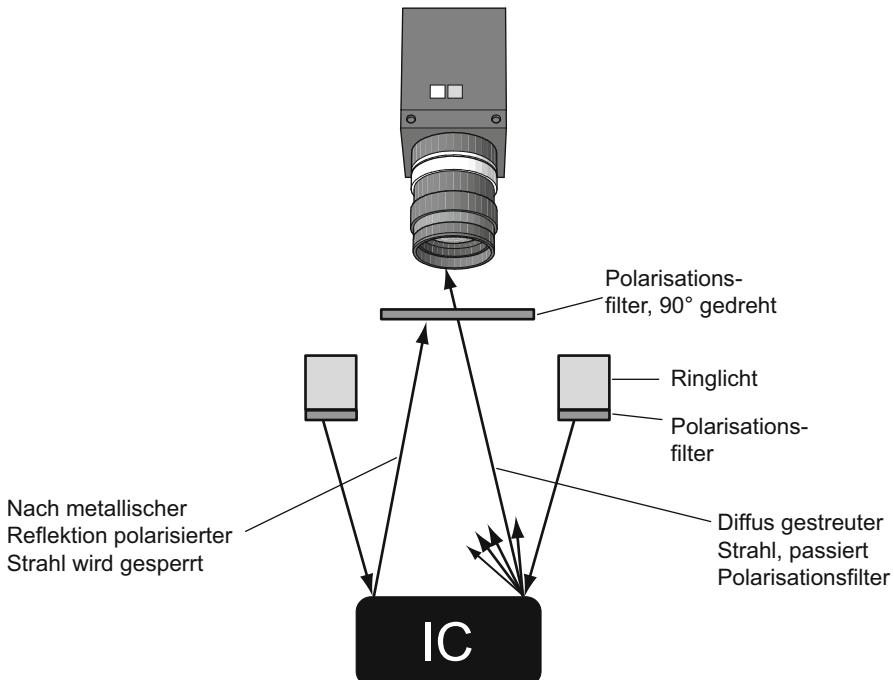


Abbildung 5.10: Beleuchtungsaufbau für IC-Prüfung

5.2.2 Basiskonfiguration der IC-Schrifterkennung

Wie schon eingangs erwähnt ist die Schrifterkennung eines der anspruchsvollsten Gebiete der industriellen Bildverarbeitung; daher werden wir diese Anwendung schrittweise aufbauen. Wir beginnen mit den unbedingt erforderlichen Elementen, erklären dann das Prinzip von Anwendungen, in denen Methoden der Mustererkennung verwendet werden, und fügen schließlich die Feinheiten hinzu, die für eine Fertigungsanwendung nötig sind. Wir stellen den Aufbau der Anwendung in folgenden Stadien dar:



Abbildung 5.11: IC unter kreuzpolarisiertem Licht

1. Zunächst gehen wir davon aus, dass die Schrift von gleich bleibender Qualität ist und der IC absolut präzise positioniert ist, um den Grundaufbau der Anwendung klarer herauszustellen. Darauf aufbauend wird dann die Vorgehensweise bei Klassifikationsanwendungen erklärt.
2. Im zweiten Stadium verwenden wir die Methoden aus Kapitel 3, um die unvermeidlichen Positionstoleranzen auszugleichen.
3. Im dritten Stadium befassen wir uns mit dem Problem schwankender Schriftqualität.

Im Folgenden werden die einzelnen Schritte kurz beschrieben, danach die für die Mustererkennung entscheidenden Punkte der Beschreibung und Klassifikation von Objekten näher erläutert.

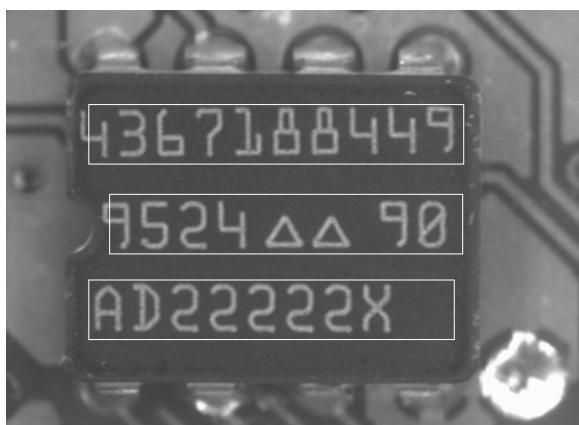


Abbildung 5.12: Arbeitsbereiche für zeilenweise Suche

1. Bildaufnahme

2. *Setzen der Arbeitsbereiche:* Für jede Zeile wird ein rechteckiger Suchbereich definiert, wie es in Abbildung 5.12 angedeutet ist. Dadurch kann die weitere Prüfung dem Aufbau des Prüfteils entsprechend strukturiert werden. Außerdem kann so die Objektsuche schneller durchgeführt werden, als wenn die gesamte Oberfläche des ICs abgesucht würde. Beachten Sie, dass die Arbeitsbereiche ganz bewusst etwas kürzer definiert sind als die Zeile. Man kann so einen gewissen Abstand zu verbleibenden Reflexionen von der abgeschrägten Kante des ICs halten. Für die Konturverfolgung ist es gleichgültig, an welcher Stelle der erste Konturpunkt gefunden wird.
3. *Objekterzeugung:* Da die Zeichen guten Kontrast zum Hintergrund aufweisen und auch nicht die Gefahr besteht, dass sie in einzelne Bruchstücke zerfallen, können sie problemlos mit einer Konturverfolgung anhand einer Binärschwelle segmentiert werden. Abbildung 5.13 zeigt das Binärbild mit den umschreibenden Rechtecken der segmentierten Zeichen.

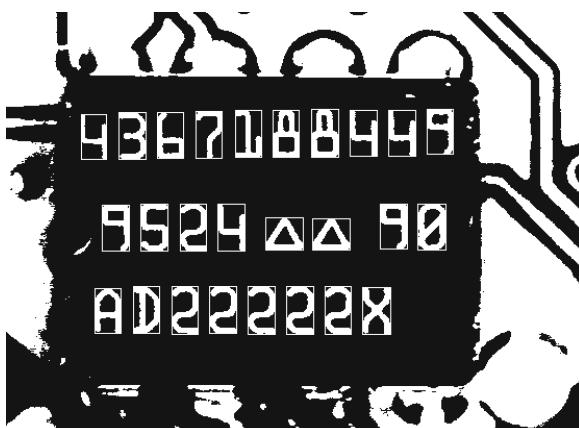


Abbildung 5.13: IC mit binär segmentierten Zeichen

4. *Berechnung von Objekteigenschaften:* An dieser Stelle liegen zwei Darstellungen für jedes Zeichen vor, das Bild selbst und die in der Objektsuche ermittelte Kontur, die z. B. wie in Abschnitt 4.3 als Kettencode dargestellt werden kann. Beide Darstellungen eignen sich nicht für die Aufgabe, die im nächsten Schritt zu bewältigen ist, die Erkennung, um welches Zeichen es sich handelt. Warum das so ist und wie man zu einer verwendbaren Darstellung kommt, werden wir gleich noch sehen; Abbildung 5.14 zeigt zunächst das Ergebnis der hier verwendeten Merkmalsberechnung: Für jedes einzelne Zeichen wird eine großen- und kontrastnormierte Darstellung erzeugt, ein sogenanntes *Rasterbild*. Das Prinzip erklärt Abbildung 5.17.
5. *Entscheidung:* Im letzten Schritt erfolgt die eigentliche Zeichenerkennung. Dazu werden die zuvor segmentierten und durch ihre Merkmale beschriebenen

Zeichenobjekte klassifiziert, d. h. jedes Objekt wird aufgrund seiner Eigenchaften in eine der möglichen Klassen eingeteilt. In unserem Fall handelt es sich dabei um die Ziffern 0–9, die Buchstaben A und D sowie das Dreieck, das zweimal in der mittleren Zeile vorkommt.

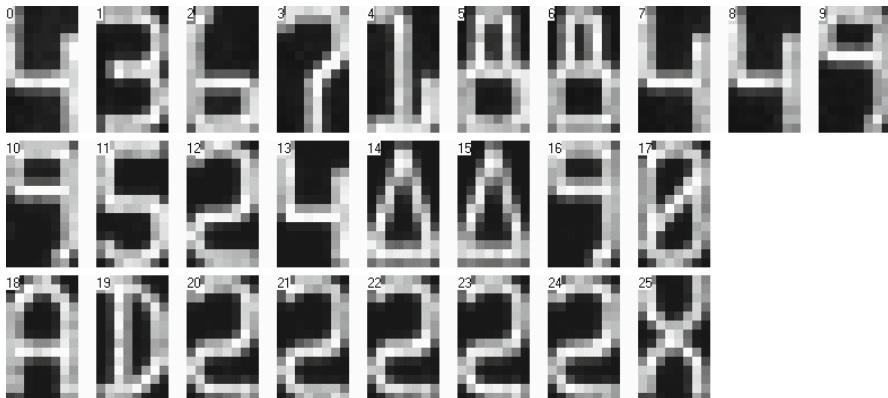


Abbildung 5.14: Gerasterte Zeichen mit einer Auflösung von 8×14

Beispielbeschreibung 5.2. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 5.2.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Ringlicht mit Kreuzpolarisation zur Unterdrückung von Reflexen
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, zeilenweise 3. <i>Binärschwelle erzeugen</i> 4. <i>Arbeitsbereiche binär erzeugen</i> 5. <i>Arbeitsbereiche rastern</i> zur Erzeugung des normierten Eingangsbildes für den Klassifikator 6. <i>Arbeitsbereiche klassifizieren</i>, benutzt ein neuronales Netz, um jedes segmentierte Objekt aufgrund seiner Rasterbilddarstellung in eine Klasse einzusortieren.

5.2.3 Prinzipieller Aufbau einer Klassifikationsanwendung

Wir haben im vorigen Abschnitt die Erzeugung der Merkmale für die Klassifikation einfach vorausgesetzt, um die ganze Anwendung zunächst im Zusammenhang vorstellen und die folgenden Ausführungen motivieren zu können. Dieser Abschnitt reicht die zum Verständnis notwendigen Erklärungen des Prinzips nach,

auf die theoretischen Grundlagen der Klassifikation gehen wir dann ausführlich in Kapitel 6 ein.

Vorgehensweise bei einer Klassifikationsanwendung: Abbildung 5.15 zeigt das prinzipielle Vorgehen beim Aufbau einer Anwendung die, wie die Schrifterkennung, einen Klassifikator verwendet.

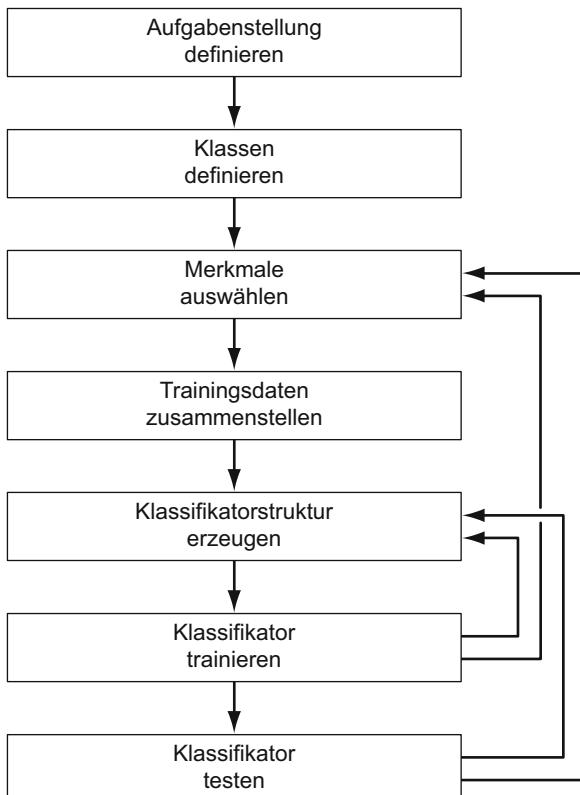


Abbildung 5.15: Vorgehensweise bei Klassifikationsanwendungen

Zu den einzelnen Schritten:

1. Zunächst muss man sich, wie immer, über die genaue Aufgabenstellung im Klaren sein.
2. Aus der Aufgabendefinition ergibt sich dann normalerweise automatisch, welche Klassen zu erkennen sind. In unserem Beispiel ist aus dem Herstellungsprozess bekannt, dass außer den zehn Ziffern nur das Dreieck und die Buchstaben A, D und X vorkommen können. Dadurch kann man die Anzahl der Ausgänge des Klassifikators auf 14 beschränken, was den Klassifikator deutlich einfacher macht, als wenn er das gesamte Alphabet erkennen müsste.

3. Die Auswahl geeigneter Merkmale ist häufig keineswegs trivial. Es gibt umfangreiche Forschungen zu diesem Thema, die hier auch als bloße Literaturhinweise nicht wiedergegeben werden könnten. In Werken zur statistischen Modellierung und Klassifikation sind weitere Hinweise hierzu zu finden, z. B. in (Smith 1993), (Bishop 1995). In der Praxis wird man im Allgemeinen aus den vom verwendeten System vorgegebenen Merkmalen experimentell die für die jeweilige Aufgabenstellung geeigneten auswählen, gegebenenfalls auch Merkmale, die durch den Charakter der Aufgabe naheliegend erscheinen, hinzu programmieren (falls das System solche Erweiterungen zulässt).
4. Der nächste Schritt ist die Zusammenstellung von Trainingsdaten, anhand derer ein Klassifikator entworfen werden kann. Hierzu benötigt man eine Sammlung von Musterteilen, in unserem Beispiel ICs, die die zu erkennenden Muster in verschiedenen Qualitäten enthalten. Um im späteren Betrieb eine sichere Klassifikation zu gewährleisten, sollten auch Grenzmuster von gerade noch akzeptabler Qualität enthalten sein. An dieser Stelle liegt übrigens eines der größten Probleme industrieller Klassifikationsanwendungen: Es ist häufig überhaupt nicht möglich, eine solche Sammlung von Teilen zu bekommen, da man es oft mit Vorserienteilen zu tun hat oder Teilen aus dem Produktionsanlauf, so dass noch gar nicht abzusehen ist, welche Qualitäten das System im Laufe der Zeit zu sehen bekommt. Es ist daher von großer Wichtigkeit bei der Auswahl eines Systems, dass sich Trainingsdaten während des Betriebes ergänzen lassen, um Prozessdrift ausgleichen zu können. Das erlaubt zudem, das System ohne störende Produktionsunterbrechungen zu optimieren.
5. Danach kann die Klassifikatorstruktur erzeugt werden. Die Ein- und Ausgangsstruktur liegt durch die Zusammenstellung von Klassen und Merkmalen an dieser Stelle bereits fest, aber die meisten Klassifikatortypen besitzen eine Reihe freier Parameter, die ihr Verhalten beeinflussen. Dieser Schritt lässt sich kaum verallgemeinern, da Art und Umfang dieser Parameter sich je nach Klassifikatortyp und verwendetem System stark unterscheiden.
6. Als nächstes wird der Klassifikator „trainiert“, d. h. die vorhandene Struktur wird anhand der Trainingsdaten so optimiert, dass die gestellte Aufgabe einwandfrei gelöst wird. Es gibt Klassifikatortypen, wie z. B. den *Nächster-Nachbar-Klassifikator*, bei denen es zwischen dem Aufbau der Struktur und dem Training keinen Unterschied gibt, aber die meisten aufwändigeren Klassifikatoren verwenden diesen zweistufigen Ablauf. Wenn das Training fehlschlägt, wird man zunächst versuchen, die Klassifikatorstruktur anzupassen. Wenn auch dann noch keine Lösung gefunden wird, kann man versuchen, durch Veränderungen der Merkmalsauswahl eine Verbesserung zu erzielen, z. B. durch Hinzufügen anderer Merkmale oder auch durch Beseitigen redundanter Merkmale, die den Trainingsvorgang stören könnten.
7. Nach erfolgreichem Training sollte der Klassifikator stets auf unabhängigem Material getestet werden, d. h. mit Teilen, die nicht für das Training verwendet wurden. Falls die Leistungen des Klassifikators in diesem Test nicht hinreichen, kann wiederum an beiden Stellen, der Klassifikatorstruktur und der Merkmalsauswahl, optimiert werden.

Der soeben geschilderte Ablauf zeigt, dass der Aufbau einer Klassifikationsanwendung unter Umständen recht aufwändig sein kann. In vielen Fällen aus der industriellen Praxis kommt man aber mit etwas Erfahrung schnell zum Erfolg. Im Folgenden machen wir noch einige Bemerkungen zu den hier im Beispiel verwendeten Merkmalen und dem Klassifikator.

Beschreibung von Zeichen zur Klassifikation: Es gibt sehr viele Möglichkeiten, Zeichen – oder sonstige Muster – für die automatische Erkennung zu beschreiben. Besonders in der Handschrifterkennung wird eine große Zahl teils sehr komplizierter Merkmale benutzt, um der großen Variationsbreite von Handschriften Herr zu werden. Bei der Erkennung gedruckter Zeichen genügt wegen der großen Ähnlichkeit von Zeichen derselben Klasse hingegen oft eine bildliche Darstellung. Es ergibt sich dabei jedoch ein Problem, das auf dem Funktionsprinzip eines Klassifikators beruht.

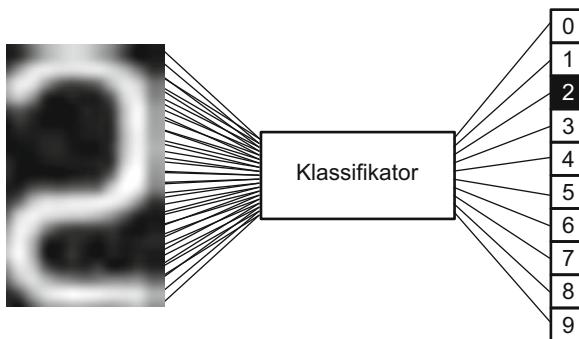


Abbildung 5.16: Klassifikator als signalverarbeitender Baustein

Kapitel 6 wird sich ausführlich mit den theoretischen Grundlagen von Klassifikatoren befassen, daher gehen wir hier nichts ins Detail, sondern betrachten den Klassifikator nur von außen, als eine Blackbox, die gewisse Eingangssignale erhält und als Ausgang die Klasseninformation liefert, vergleichbar einem signalverarbeitenden Bauteil. In unserem Beispiel heißt das konkret, dass der Klassifikator aus den Eigenschaften der einzelnen segmentierten Objekte ermittelt, um welches Zeichen es sich handelt. Abbildung 5.16 illustriert dieses Prinzip anhand einer reinen Ziffernerkennung. Prinzipiell könnte man jetzt, wie es auch in Abbildung 5.16 angedeutet ist, einfach die einzelnen Bildpunkte, die im umschreibenden Rechteck eines Zeichens liegen, als Eingangssignale verwenden, oder auch die einzelnen Elemente des Kettencodes, der die Zeichenkontur beschreibt. Aus der Analogie mit einem elektronischen Baustein zur Signalverarbeitung ergibt sich aber ein Problem: Ein solcher Baustein besitzt eine gewisse Anzahl von Eingängen, die alle belegt werden müssen. Weder die Konturbeschreibungen noch die Bilder der einzelnen Zeichen bestehen aber immer aus gleich vielen Werten. Die 2 auf dem IC ist deutlich breiter als die 1 und besteht aus entsprechend mehr Bildpunkten. Bei der Konturbeschreibung kommt zur unterschiedlichen Länge noch

erschwerend hinzu, dass die Kontur völlig verschieden aussehen kann, je nachdem an welchem Punkt mit der Konturverfolgung begonnen wurde.

Man muss also dafür sorgen, dass das „Bild“ jedes Zeichens, das dem Klassifikator zur Erkennung vorgelegt wird, stets aus derselben Anzahl von Bildpunkten besteht. Zu diesem Zweck führt man eine Größennormierung durch, bei der benachbarte Bildpunkte zusammengefasst werden, so dass schließlich ein Bild mit einer vorgegebenen Höhe und Breite entsteht, wie es Abbildung 5.17 zeigt. Vor der Rasterung ist die 2 etwa doppelt so breit wie die 1, hinterher haben beide Zeichen die gleichen Abmessungen.

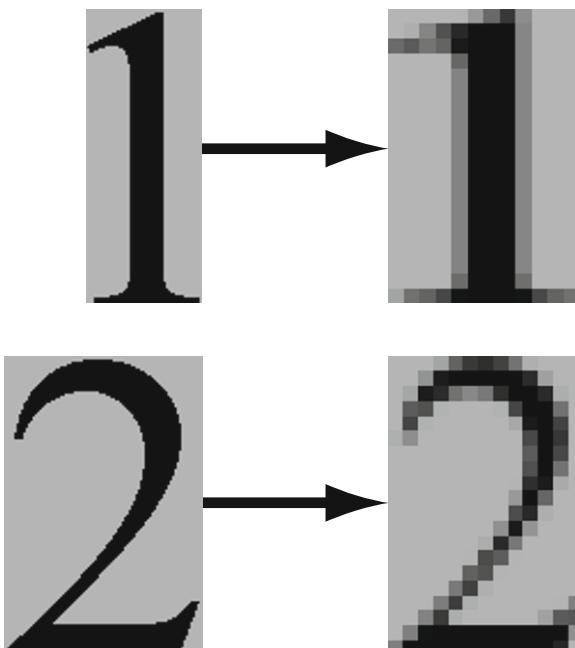


Abbildung 5.17: Prinzip der Rasterung von Zeichen für die Klassifikation

Umfangreiche Untersuchungen an industriell gebräuchlichen Beschriftungen, (Schwenk 1991), haben gezeigt, dass eine Auflösung von 8×14 Punkten für die Unterscheidung von Ziffern einen guten Kompromiss zwischen Fehlerrate und Verarbeitungsgeschwindigkeit darstellt. Feinere Auflösungen können die Fehlerrate weiter senken, unter Umständen aber den Klassifikator auch empfindlicher machen, da er sich an einen höherdimensionalen Merkmalsraum anpassen muss. Wie die Art der verwendeten Merkmale sind auch solche Einstellungen von der jeweiligen Anwendung abhängig und müssen experimentell ermittelt werden.

Neben der Größennormierung empfiehlt es sich, auch eine Kontrastnormierung vorzunehmen. Um bei der Analogie mit der elektrischen Signalverarbeitung zu bleiben: Signale haben begrenzte Wertebereiche – bei einem Grauwertbild als Eingangssignal ist dies das Intervall [0, 255]; je besser dieser Wertebereich ausge-

nutzt wird, desto besser wird auch das Verarbeitungsergebnis sein und desto leichter wird es auch dem Klassifikator fallen, die Problemstellung zu erlernen. Beispielsweise stellt dasselbe Zeichen mit verschiedenen Grundhelligkeiten für den Klassifikator ganz verschiedene Signale dar, deren Klassenzugehörigkeit er separat erlernen müsste. Nach einer Kontrastnormierung sind die Zeichen hingegen identisch, wie Abbildung 5.18 zeigt. Damit wird auch die Menge unterschiedlicher Signalkombinationen, die der Klassifikator erlernen muss, erheblich verringert.

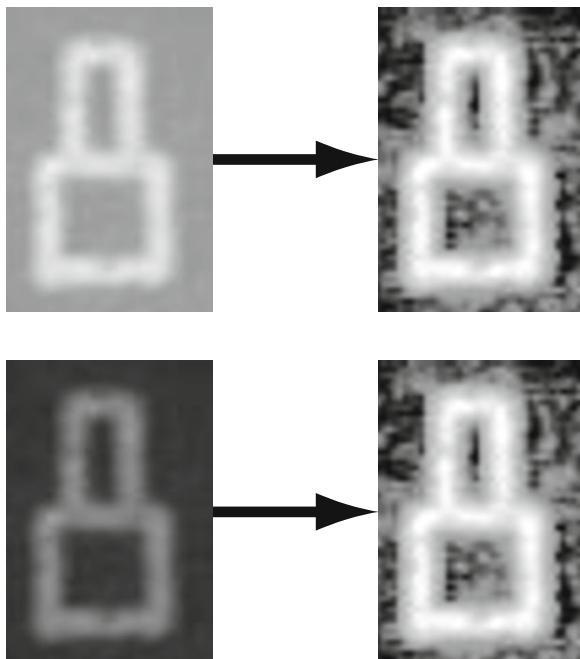


Abbildung 5.18: Kontrastnormierung eines Zeichens mit unterschiedlicher Grundhelligkeit

Erstellung eines Klassifikators: Zur Erstellung eines Klassifikators werden gemäß dem Ablauf in Abbildung 5.15 Trainingsdaten benötigt. Ein solcher Datensatz besteht aus einer Reihe von Trainingsmustern; jedes Trainingsmuster enthält die Merkmale eines Objekts, in unserem Beispiel also die Rasterdarstellung eines Zeichens, und die zugehörige Klasseninformation. Die Merkmale kann das System bereits jetzt selbst berechnen, die Klasseninformation muss jedoch für die Trainingsphase noch vom Menschen vorgegeben werden. Nach dem Abspeichern der Merkmale einer repräsentativen Auswahl von Objekten folgt also die manuelle Klassifikation all dieser Muster – wobei das verwendete System durchaus Hilfestellungen geben kann, z. B. in Form einer Vorsortierung nach Ähnlichkeiten von Mustern.

Wenn der Trainingsdatensatz fertig ist, kann der Klassifikator erzeugt und optimiert werden. Wie die Zusammenstellung von Trainingsdaten und die Erzeugung und Optimierung eines Klassifikators im Einzelnen vor sich gehen, ist je nach

verwendetem Bildverarbeitungssystem sehr unterschiedlich, daher können wir hier nur auf die Dokumentation des jeweiligen Systems verweisen.

Programmhinweis: Die Erstellung von Trainingsdaten und Klassifikatoren in **NeuroCheck** ist in der Online-Dokumentation ausführlich beschrieben.

5.2.4 Positions korrektur am IC

Kehren wir nach diesem Ausflug in die Grundlagen zu unserem Anwendungsbeispiel zurück. In Abschnitt 5.2.2 waren wir von der unrealistischen Ausnahme ausgegangen, dass der IC immer exakt positioniert ist. Selbst in einem Werkstückträger lässt sich diese Voraussetzung nicht erfüllen, vor allem wenn man sich vor Augen hält, dass der annähernd sieben Millimeter breite IC sich in Abbildung 5.11 über eine Breite von ca. 375 Pixeln erstreckt, ein Bildpunkt also etwa 0,02 mm entspricht. Eine Positionstoleranz von nur einem Millimeter bedeutet dann im Bild eine Verschiebung von 50 Bildpunkten, die eine vernünftige Segmentierung natürlich unmöglich macht. Rotationen stellten bei dieser Anwendung ein wesentlich kleineres Problem dar, mit mehr als zwei bis drei Grad Verdrehung war keinesfalls zu rechnen.

Positionierung durch Kantensuche: Aufgrund der dunklen Leiterbahneinfassungen lässt sich der Korpus des ICs nicht stabil von der Umgebung trennen, aber der gute Kontrast zwischen dem dunklen Material des ICs und dem Platinenhintergrund in Bild 5.11 legt den Gedanken nahe, die Positions korrektur in derselben Form vorzunehmen wie in Abschnitt 3.1.2, d. h. anhand der Objekt kanten.

Bei einer Positionierung an Objekt kanten benötigt man eine senkrechte Kante für die Positionierung in X-Richtung und eine waagerechte Kante für die Y-Richtung. Die waagerechte Kante musste dabei genau zwischen zwei der Löt punkte am oberen Ende gefunden werden, um eine stabile Y-Referenz zu bilden. Dazu war zunächst eine genaue X-Positionierung erforderlich, d. h. es wurde eine hierarchische Positions korrektur in folgender Weise durchgeführt:

1. Im ersten Schritt wird eine senkrechte Kante gesucht. Mit deren Hilfe wird der Suchbereich für die waagerechte Kante positioniert.
2. In diesem Suchbereich wird dann die waagerechte Kante gesucht und beide Ergebnisse zusammen werden für die Positionierung der endgültigen Suchbereiche für die Zeichen benutzt.

Bei der Feststellung der ersten Referenzkante tritt dabei die Schwierigkeit auf, dass in diesem Stadium kaum Anhaltspunkte für die Positionierung vorliegen. Je nach Position der Platine können unterschiedlich viele Leiterbahnen im Suchbereich liegen, so dass zwischen gültigen und ungültigen Kanten unterschieden werden muss. Da der IC insgesamt dunkler ist als die Leiterbahnen, ist es durch entsprechende Einstellung der Binärschwelle zwar möglich, die ungültigen Kanten von vornherein von der Segmentierung auszuschließen, aber wie man in Bild 5.19 erkennt, beginnt dadurch auch die Kontur der Kante zu zerfasern. Das liegt an der

Abschrägung am Rand des ICs, die die Kante stellenweise aufhellt. Damit eignete sich die Kante kaum noch für eine exakte Positionsbestimmung. Die Abbildung zeigt übrigens auch, dass selbst bei dieser niedrigen Binärschwelle der IC noch immer mit Teilen der Leiterbahnen verschmilzt, vor allem im Bereich der unteren Lötpunkte. Da diese Effekte niemals wirklich gleich bleibend und stabil sind, führen sie zu unvorhersehbaren Veränderungen der Form des ICs im Binärbild, so dass man ihn nicht als Ganzes zur Positionsbestimmung benutzen kann.



Abbildung 5.19: Zerfaserung der IC-Kante nach Ausblenden der Leiterbahnen durch die Binärschwelle

Schon in Abschnitt 3.1.2 wurde ein Glättungsfilter benutzt, um die Positionierung anhand einer Kante zu stabilisieren. Mit demselben Verfahren lässt sich auch hier eine hinreichend gerade und stabile Kante erzeugen, wie Abbildung 5.20 zeigt.



Abbildung 5.20: Kantenglättung zur Verbesserung der Positionierung

Mit diesen Verfeinerungen konnte die Positions korrektur anhand der IC-Kanten mit Erfolg eingesetzt werden. Im Laufe der Zeit erwies sich der relativ komplexe und von vielen Parametern mit geringen Toleranzen abhängige Verarbeitungsablauf jedoch als zu starr und störanfällig. Neben Positionstoleranzen, die so groß waren, dass teilweise schon die Detektierung der ersten Referenzkante fehlschlug, waren es vor allem Umstellungen in der Produktion, die zu Problemen führten. Änderungen des Platinenlayouts machten häufige Rekonfigurationen erforderlich, die bei diesem Ablauf große Sorgfalt erforderten. Gravierender wirkte sich aber der Umstand aus, dass das Platinenmaterial sich immer wieder änderte – je nachdem, welches Material gerade auf dem Markt verfügbar war. Der Kontrast zwischen IC und Platine änderte sich dadurch ebenfalls je nach Charge.

Positionierung mit Template Matching: Aufgrund der Probleme der Kantendetektion wurde ein neuer Ansatz entwickelt, der sich auf das garantierte Vorhandensein der beiden Dreiecke in der zweiten Zeile stützte. Diese Dreiecke dienen zur Kontrolle der Beschriftung beim Chiphersteller selbst, was bedeutet, dass die restliche Beschriftung relativ zu diesen beiden Dreiecken immer gleich positioniert ist. Die Dreiecke eignen sich sehr gut für eine Segmentierung mit dem Template-Matching-Verfahren, und da die Oberfläche des ICs von stabiler Helligkeit ist, gibt es hier auch keine Kontrastprobleme.

Es drängt sich natürlich die Frage auf, warum dieser Ansatz nicht von vornherein gewählt wurde, angesichts der Probleme der kantenorientierten Methode. Man darf dabei nicht übersehen, dass auch einige Punkte für die Verwendung der Kanten als Positionsreferenz sprechen. Einerseits ist Template Matching ein rechenintensives Verfahren, das je nach Taktzeit und verfügbarer Hardware durchaus Zeitprobleme aufwerfen kann, während die Kantendetektion entlang einer Suchlinie sehr schnell abläuft. Andererseits geht man ja bei der Lösung einer Ingenieursaufgabe immer von einer Spezifikation aus, die zu erfüllen ist. Im vorliegenden Fall war, als die Aufgabe ursprünglich spezifiziert wurde, weder die Abschrägung an den IC-Kanten vorhanden, noch war absehbar, dass das Platinenlayout und -material solchen Schwankungen unterworfen sein würden. Es ist keineswegs selten, dass eine im Rahmen der ursprünglichen Spezifikation gute Lösung durch Produktionsumstellungen und andere Umstände später nicht mehr brauchbar ist und ein neuer Weg gefunden werden muss. Wenn es sich also nicht um Teile handelt, die jahrelang in völlig unveränderter Form von einem Band laufen, ist es für die Auswahl eines Prüfsystems von großer Bedeutung, wie schnell und flexibel man damit auf solche Änderungen reagieren kann und welchen Aufwand eine Umkonfiguration erfordert.

Es erwies sich als die beste Vorgehensweise, nicht nach zwei einzelnen Dreiecken zu suchen, sondern gleich das ganze Paar als Template zu definieren. Die große Ausdehnung und die Dicke der Dreiecks kanten ermöglichen eine große Schrittweite für die in 4.5.2 beschriebene Pyramidensuche, so dass sich der zusätzliche Zeitaufwand in Grenzen hielt. Das Beispiel ist in Beschreibung 5.3 zusammengefasst.

Beispielbeschreibung 5.3. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 5.2.4 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Ringlicht mit Kreuzpolarisation zur Unterdrückung von Reflexen.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, gesamtes Bild 3. <i>Template Matching</i> zur Detektierung des Dreieckspaares als Positionsreferenz 4. <i>Arbeitsbereiche Position bestimmen</i> 5. <i>Arbeitsbereiche definieren</i>, zeilenweise passend für den IC im Referenzbild für die Positionierung 6. <i>Arbeitsbereiche positionieren</i>, entsprechend der Position des Dreieckspaares 7. Nach der Positionierung liegen die Arbeitsbereiche genau über den Zeilen des aktuellen ICs, daher entspricht der weitere Ablauf genau Beispiel 5.2.

Die Aussage aus Abschnitt 4.5.1, dass Template Matching empfindlich auf Verdrehungen reagiert, bleibt natürlich bestehen, aber wie oben schon erwähnt, sind in dieser Anwendung keine größeren Rotationen zu erwarten. Verdrehungen bis zu $\pm 4^\circ$ werden bei der Suche nach dem Dreieckspaar ohne weiteres toleriert, wie Abbildung 5.21 zeigt. Der schwarze Rahmen markiert das Dreieckspaar, das mit einer Korrelation von über 0,9 sicher gefunden wurde.

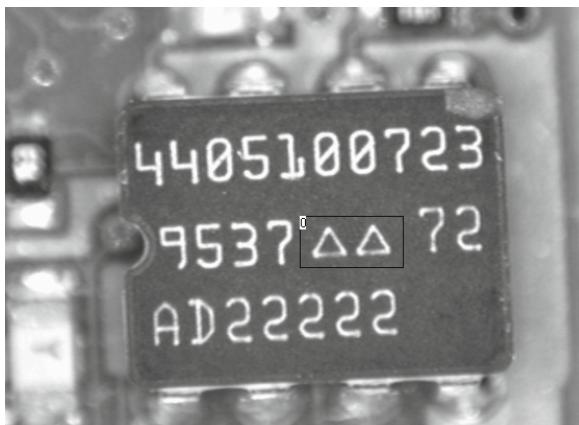


Abbildung 5.21: Template Matching auf gedrehtem IC

Eine Kleinigkeit muss man allerdings im Fall einer Verdrehung noch beachten. Segmentierungsverfahren finden Objekte nicht unbedingt in einer Reihenfolge, die dem Menschen sinnvoll erscheint. Jeder Mensch sieht sofort die Zeilenstruktur der IC-Beschriftung, das Bildverarbeitungssystem weiß davon aber nichts. Es hängt

von der Konstruktion des Segmentierungsalgorithmus' ab, in welcher Reihenfolge die Zeichen gefunden werden. Aus programmtechnischen Gründen sucht man Bilder in der Regel immer innerhalb der Arbeitsbereiche zeilenweise waagerecht ab. Prinzipiell könnte man selbstverständlich auch parallel zu den Kanten eines gedrehten Rechtecks suchen, aber es müsste dann nicht nur die Verdrehung bestimmt werden, es wäre auch außerordentlich langsam, da man nicht einfach Bildpunkt für Bildpunkt vorgehen kann, sondern nach jedem Punkt aufgrund des Drehwinkels den nächstfolgenden berechnen müsste.

Man findet dann bei dem rotierten IC aus Abbildung 5.21 die weiter rechts liegenden Zeichen zuerst, weil sie durch die Verdrehung höher im Bild liegen und daher zuerst von Suchlinien getroffen werden. Abbildung 5.22 zeigt die Reihenfolge, in der die Zeichen bei dieser Verdrehung gefunden werden.



Abbildung 5.22: Reihenfolge der Zeichen bei zeilenweiser Suche

Für die nachfolgende Klassifikation stellt diese wenig intuitive Reihenfolge insoweit ein Problem dar, als eine Kontrollinstanz, z. B. eine SPS oder eine interne Funktion zur Klassenauswertung, gar nicht mehr überprüfen kann, ob die Zeichen überhaupt in der richtigen Reihenfolge auf dem IC stehen. Die Abhilfe ist natürlich sehr einfach, man sortiert die Zeichen in jedem Arbeitsbereich einzeln nach ihrer X-Koordinate – womit übrigens auch klar ist, warum für jede Zeile ein einzelner Arbeitsbereich definiert wurde: Abgesehen davon, dass weniger Fläche abgesucht werden muss, erspart man es sich auch, zunächst die Zeilen nach Y-Koordinaten unterscheiden zu müssen, bevor man korrekt Zeile für Zeile sortieren kann. Die Darstellung des kompletten Ablaufs mit Sortierung überspringen wir hier, Sie finden sie in der Beispielbeschreibung 5.4, da sich dort nur noch eine Kleinigkeit ändert.

5.2.5 Verbesserung der Zeichenqualität

Ein weiteres Problem entstand im Laufe der Zeit dadurch, dass die Beschriftung mit zwei Lasern aufgebracht wird: Die erste Zeile und die vordere Hälfte der zweiten Zeile wurden mit einem Laser geschrieben, der Rest mit einem zweiten Laser. Durch Unterschiede in Alter und Justierung der beiden Laser entstehen stark unterschiedliche Kontraste in diesen Bereichen, wie Abbildung 5.23 zeigt. Dies kann zu Problemen bei der Binärsegmentierung führen.

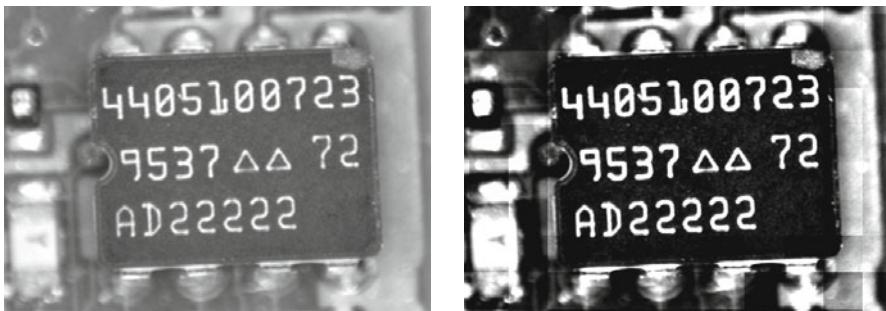


Abbildung 5.23: IC mit unterschiedlichem Schriftkontrast, vor und nach Normierung

Dies gilt insbesondere in der zweiten Zeile, in der der Kontrast dann mitten im Arbeitsbereich wechselt, so dass auch eine separate Binärschwelle für jeden Arbeitsbereich das Problem nicht beseitigt.

Eine Lösung besteht darin, für die zweite Zeile zwei Arbeitsbereiche zu verwenden, was aber wiederum die Positionierung empfindlicher macht. Einfacher ist es in diesem Fall, einen Vorverarbeitungsschritt einzuführen, der den Kontrast ausschnittweise normiert. Das Bild wird dabei in rechteckige Sektoren zerlegt. In jedem Sektor werden nach einer Histogrammanalyse die Grauwerte so modifiziert, dass der volle Grauwertbereich von [0...255] ausgenutzt wird, wie Abbildung 5.23 zeigt.

Abgesehen vom Kontrast unterscheiden sich die Zeichen auch in der Strichdicke. Grundsätzlich ist ein solcher Defekt durch eine morphologische Filterung (siehe Abschnitt 2.5) zu beheben, allerdings gibt es dabei die Schwierigkeit, dass man zunächst entscheiden muss, wo der Filter anzuwenden ist, da er ja nur bei den zu dünnen Zeichen zur Wirkung kommen soll. Andernfalls bliebe der Unterschied zwischen den Zeichen bestehen, da sich nur die absolute Strichdicke verändern würde. Leider kann man sich auch nicht darauf verlassen, dass immer der zweite Laser die dünneren Zeichen erzeugt, wie Abbildung 5.24 zeigt.

Zur Entscheidung, in welchem Bereich die Zeichen gefiltert werden müssen, wäre eine Strukturanalyse erforderlich, die die mittlere oder minimale Strichbreite ermittelt. Diese Analyse ist nicht nur rechnerisch aufwändig, man müsste die Objekte auch erst einmal finden, um sie analysieren zu können. Es ist in diesem Fall wesentlich einfacher, in den Trainingsdatensatz für den Klassifikator Zeichen beider Typen aufzunehmen. Die leistungsfähigen Klassifikatoren, die heute z. B. in

Form neuronaler Netze zur Verfügung stehen, sind ohne weiteres in der Lage, mit dieser Varianz zurechtzukommen. Beispielbeschreibung 5.4 stellt noch einmal den kompletten Ablauf zusammen.



Abbildung 5.24: IC-Beschriftung mit falsch justiertem ersten Laser

5.2.6 Optimierung im Betrieb

Wir wiesen schon im Abschnitt 5.2.4 im Zusammenhang mit den Problemen bei der kantenorientierten Positionierung darauf hin, dass es bei industriellen Bildverarbeitungsanwendungen selten möglich ist, (a) eine Spezifikation zu erhalten, die alle späteren Eventualitäten und Modifikationen berücksichtigt, und (b) einen umfassenden Satz seriengleicher Musterteile zu erhalten. Da Prüfanlagen heute bereits oft integraler Bestandteil der Planung einer Fertigungslinie sind und nicht mehr nachträglich installiert werden, liegen noch gar keine Serienteile vor, wenn die Bildverarbeitung entwickelt wird. Man ist also gezwungen, die Prüfung an Vorserienteilen oder bestenfalls Teilen aus dem Produktionsanlauf zu entwickeln. Das bedeutet andererseits, dass es unumgänglich ist, das Bildverarbeitungssystem im Betrieb zu optimieren und auf die Veränderungen im Produktionsprozess abzustimmen.

Eine Klassifikationsanwendung im Prozess zu optimieren, erfordert zuerst einmal, mögliche Fehlerquellen zu erfassen, d. h. das Bildverarbeitungssystem muss sich selbst daraufhin überwachen, ob unsichere Ergebnisse entstehen. Das ist in der Tat möglich, wie in Kapitel 6 noch eingehender besprochen wird. Alle Zeichen, die das System mit einer Sicherheit unterhalb einer einstellbaren Rückweitungsschwelle erkannte, wurden automatisch in einer Datei gesammelt und von Zeit zu Zeit vom Anlagenpersonal durchgesehen. Darunter befanden sich natürlich auch echte Fehlermuster, bei denen die Zurückweisung durch den Klassifikator berechtigt war. Solche Fehlermuster entstanden zum Beispiel infolge von Ver-

schmutzung durch Staub, die bei der geringen Zeichengröße durchaus zur Unlesbarkeit führen kann. Solche Muster wurden aussortiert, denn es wäre natürlich gefährlich, den Klassifikator auf die Erkennung derart veränderter Zeichen zu trainieren, da er dann echte Fehler möglicherweise nicht mehr feststellen kann. Die restlichen Muster, die nur deswegen zurückgewiesen wurden, weil sie sich zu stark von den bis dahin benutzten Trainingsmustern unterschieden, wurden dem Trainingsdatensatz hinzugefügt, der Klassifikator dann mit der kompletten Sammlung alter und neuer Muster neu trainiert. Dadurch lernte das System allmählich mit der ganzen Variationsbreite umzugehen, die in der Serienproduktion vorkam.

Beispielbeschreibung 5.4. Die folgende Tabelle fasst das Beispiel aus Abschnitt 5.2.5 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Ringlicht mit Kreuzpolarisation zur Unterdrückung von Reflexen.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Bild normieren</i>, sorgt für vollen Kontrast in jedem Bildausschnitt. 3. <i>Arbeitsbereiche definieren</i>, gesamtes Bild 4. <i>Template Matching</i> zur Detektierung des Dreieckspaares als Positionsreferenz 5. <i>Arbeitsbereiche Position bestimmen</i> 6. <i>Arbeitsbereiche definieren</i>, zeilenweise passend für den IC im Referenzbild für die Positionierung 7. <i>Arbeitsbereiche positionieren</i>, entsprechend der Position des Dreieckspaares 8. <i>Binärschwelle erzeugen</i> 9. <i>Arbeitsbereiche binär erzeugen</i>, findet die Zeichen bildzeilenweise, d. h. nicht unbedingt in der intuitiv richtigen Reihenfolge. 10. <i>Arbeitsbereiche Merkmale erzeugen</i>, ermittelt die X-Koordinaten der Arbeitsbereiche; diese stehen nach der Binärsegmentierung nicht unmittelbar zur Verfügung, da nicht sicher ist, dass die Kontur bei der minimalen X-Koordinate gefunden wird. 11. <i>Arbeitsbereiche sortieren</i>, sortiert die Zeichen in jeder Zeile separat nach X-Koordinate, damit später die Richtigkeit der Beschriftung beurteilt werden kann. 12. <i>Arbeitsbereiche rastern</i>, stellt das großen- und kontrastnormierte Eingangsbild für den Klassifikator her. 13. <i>Arbeitsbereiche klassifizieren</i>, ermittelt zu jedem Objekt, welcher Klasse es angehört, d. h. welches Zeichen es darstellt.

Programmhinweis: Beachten Sie bei dem im Download enthaltenen Beispiel bitte, dass der Klassifikator ICO.CLA nur mit Zeichen aus dem Bild ICO.BMP trainiert wurde. Er kennt daher nur die Schriftqualität dieses einen ICs und von manchen Zeichen tatsächlich nur ein einziges Muster. Seine Leistungen bei den

Zeichen aus den anderen Bildern sind entsprechend schlecht; es werden zwar alle Zeichen richtig erkannt, aber häufig mit sehr geringer Sicherheit. Wenn Sie das Beispiel auf Ihre Festplatte kopieren, können Sie problematische Zeichen automatisch in der Trainingsdatei abspeichern lassen, indem Sie im Prüfschritt *Arbeitsbereiche klassifizieren* die Option „*Unsicher klassifizierte Muster automatisch abspeichern*“ aktivieren und eine Rückweisungsschwelle von beispielsweise 60 % einstellen. Erneutes Trainieren des Klassifikators führt dann zu erheblich besseren Leistungen – auch auf Bildern, aus denen keine zusätzlichen Muster abgespeichert wurden, weil der Klassifikator allmählich die gesamte Bandbreite des Problems kennen lernt.

5.3 Erkennung genagelter Schrift auf Metall

In vielen Fertigungsprozessen müssen Kennzeichnungen auf Werkstücken angebracht werden, die von nachfolgenden Bearbeitungsschritten nicht angegriffen werden. Auch gesetzliche Vorschriften können, beispielsweise bei Kraftfahrzeugen, solche beständigen Kennzeichnungen erfordern.

Die Aufbringung dieser Kennzeichnungen erfolgt häufig mit Schlagstempeln oder Nadelprägemaschinen, die Vertiefungen im Material erzeugen. Aufgrund dieser dreidimensionalen Struktur sind die entstehenden Zeichen häufig schwierig zu beleuchten und zu segmentieren. Je nach Qualität der Aufbringung kann die Tiefe eines Zeichens an verschiedenen Stellen unterschiedlich sein, so dass die Reflexionseigenschaften nicht gleichmäßig sind. Das Zeichen kann dann optisch zerfallen, wodurch es nicht mehr mittels Konturverfolgung segmentierbar ist.

5.3.1 Beleuchtung

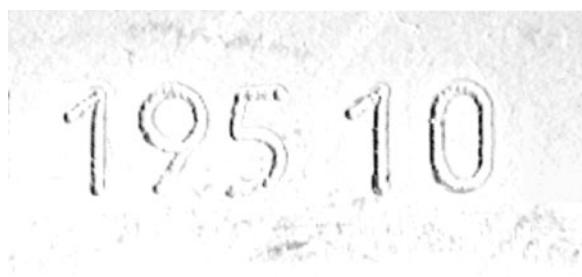


Abbildung 5.25: Geprägte Ziffern unter normaler Auflichtbeleuchtung

Abbildung 5.25 zeigt mit einem Nadelpräger erzeugte Ziffern auf einer Metalloberfläche, die unter einer gewöhnlichen Auflichtbeleuchtung aufgenommen wurden. Hier ist praktisch keine stabile Segmentierung möglich, da zum Teil über-

haupt kein Unterschied und keine Begrenzung zwischen Ziffer und Hintergrund erkennbar ist.

Das Problem kann durch eine Dunkelfeldbeleuchtung gelöst werden, die in Abbildung 5.26 schematisch dargestellt ist. Von einer glatten, spiegelnden Oberfläche wird dabei kein Licht zur Kamera reflektiert, deren Bild dann dunkel erscheint. Vertiefungen in der Oberfläche erscheinen dagegen hell, da die der Beleuchtung zugewandten Kanten das Licht zur Kamera reflektieren. Im Gegensatz dazu hat eine stark strukturierte Oberfläche, wie sie durch bestimmte Bearbeitungsverfahren entstehen oder auch durch Verunreinigungen hervorgerufen werden kann, sehr viele Mikrovertiefungen, die Licht zur Kamera streuen. Eine solche Oberfläche erscheint daher unter dieser Beleuchtung durchgängig hell, während die der Beleuchtung abgewandten Kanten der Zeichen dunkel hervortreten. Unabhängig von der Oberflächenstruktur erzeugt diese Art der Beleuchtung also einen deutlichen Unterschied zwischen der Oberfläche und den Zeichen.

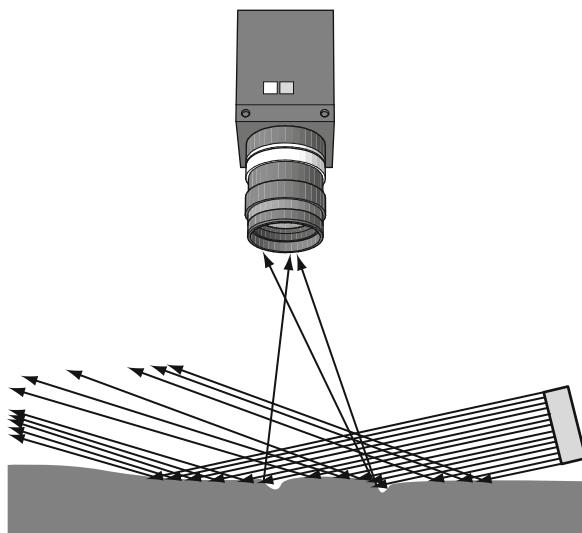


Abbildung 5.26: Dunkelfeldbeleuchtung für geprägte Zeichen

5.3.2 Vorverarbeitung

In der oberen Bildreihe in Abbildung 5.27 sind Zeichen verschiedener Prägequalität auf unterschiedlichen Oberflächen unter der oben beschriebenen Dunkelfeldbeleuchtung gezeigt. Man sieht, dass die Segmentierung der Zeichen auch unter dieser Beleuchtung noch ein Problem aufwirft. Es besteht zwar überall ein Unterschied zwischen Zeichen und Hintergrund, aber einmal ist das Zeichen heller, an anderen Stellen der Hintergrund. Da man nicht vorher wissen kann, welche Teile eines Zeichens aufgrund ihrer Prägetiefe heller erscheinen als die Oberflä-

che, welche dunkler, ist es noch nicht ohne weiteres möglich, die Zeichen zu segmentieren. Aus diesem Grund unterzieht man den Bildausschnitt einer Vorverarbeitung mit einem Differenzfilter. Dieser hebt jede Art von Unterschied in der Umgebung eines Bildpunkts hervor, unabhängig vom Vorzeichen des Unterschiedes. Die untere Bildreihe in Abbildung 5.27 zeigt das Ergebnis einer Filterung mit einem Differenzfilter der Größe 5×5 . Trotz des stark unterschiedlichen Ausgangsmaterials erscheinen alle Ziffern hell auf dunklem Grund.

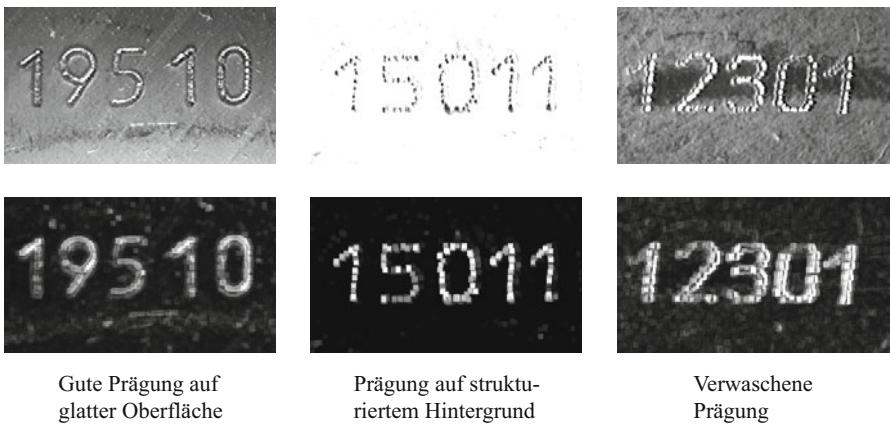


Abbildung 5.27: Geprägte Ziffern vor und nach Differenzfilterung

Programmhinweis: Den Differenzfilter finden Sie in **NeuroCheck** im Prüfschritt *Bild filtern* unter dem Namen „Kontrast“, da er die Helligkeit jedes Punkts im Ergebnisbild entsprechend dem Kontrast in einer Umgebung dieses Punkts im Eingangsbild setzt

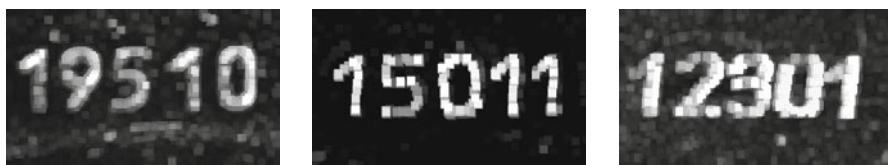
5.3.3 Segmentierung und Klassifikation

Nach der Differenzfilterung erscheinen in Abbildung 5.27 alle Zeichen hell auf dunklem Hintergrund, da die Oberfläche selbst kaum Kontrast aufweist. Man sieht jedoch auch, dass die unterschiedliche Qualität der Ziffern und des Hintergrunds auch nach der Differenzfilterung noch zu einem unterschiedlichen Erscheinungsbild führt. Auf der stark strukturierten und daher in der Dunkelfeldbeleuchtung hell erscheinenden Oberfläche sind die einzelnen Vertiefungen als kleine dunkle Punkte auszumachen, während die tiefe, verwaschene Prägung zu sehr dicken, fast verschmelzenden Zeichen führt. Eine Binärsegmentierung wird also trotz des guten Kontrastes nicht zum Erfolg führen, da entweder keine zusammenhängenden Konturen gefunden oder die Zeichen nicht getrennt werden können. Aus diesem Grund muss Template Matching zur Segmentierung verwendet werden. Da jede dieser stark unterschiedlichen Zeichenqualitäten für eine sichere Lokalisierung ei-

gene Templates erfordert, lohnt es sich die Suche zu beschleunigen. Dies lässt sich folgendermaßen erreichen:

Groblokalisierung mit Binärsegmentierung: Wenn die Suche der Zeichen in größeren Bildausschnitten erfolgen muss, kann nach der Differenzfilterung zumindest die ungefähre Position der gesamten Zeichengruppe durch Binärsegmentierung ermittelt werden. Anhand dieser Position kann dann ein relativ kleiner Suchbereich gesetzt werden, der wesentlich schneller durchsucht werden kann als das gesamte Bild.

Dilatationsfilterung: Eine Beschleunigung des Template Matching selbst findet ihre Grenzen in der geringen Strukturbreite schlecht geprägter Ziffern. Eine Dilatationsfilterung ermöglicht eine größere Suchschrittweite für die erste Stufe der pyramidalen Suche, wie sie in Abschnitt 4.5.2 beschrieben ist. Abbildung 5.28 zeigt die Bilder aus Abbildung 5.27 nach der Grauwertdilatation.



Gute Prägung auf glatter Oberfläche

Prägung auf strukturiertem Hintergrund (Rost)

Verwaschene Prägung

Abbildung 5.28: Geprägte Ziffern nach Dilatation

Beide Möglichkeiten zur Beschleunigung der Prüfung lassen sich gemeinsam nutzen. Wenn die Grob lokalisierung vor der Dilatation durchgeführt wird, ergibt sich als zusätzlicher Vorteil, dass nur der Suchbereich für das Template Matching dilatiert werden muss, nicht das gesamte Bild. Alle weiteren Schritte laufen genauso ab wie bei der Schrifterkennung auf dem IC in den vorangegangenen Abschnitten, daher werden wir jetzt nicht näher darauf eingehen. Die Beschreibung des Beispiels finden Sie in Tabelle 5.5.

Programmhinweis: Beachten Sie bei dem im Download enthaltenen Beispiel, dass die Trainingsdatei des Klassifikators keine Beispiele aus dem qualitativ schlechtesten Bild mit den zerfallenden Zeichen auf hellem Hintergrund enthielt. Daher werden diese Zeichen nur mit geringer Sicherheit erkannt. Das Template Matching hingegen verwendet Muster aller Qualitäten.

Beispielbeschreibung 5.5. Die folgende Tabelle fasst das Beispiel aus Abschnitt 5.3 zusammen. Ein erheblicher Teil der Prüfzeit entfällt auf das Template Matching, das für die unterschiedlichen Schriftqualitäten jeweils unterschiedliche Templates benötigt, und die beiden Filterstadien.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Flache Dunkelfeldbeleuchtung zur sicheren Erzeugung von Kontrasten an den Kanten der Prägepunkte.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Bild filtern</i>, hebt die Unterschiede zwischen Prägepunkten und Hintergrund hell hervor, während die homogene Oberfläche mangels Kontrast dunkel erscheint. 3. <i>Bild filtern</i>, vergrößert die Strukturbreite der Zeichen durch Grauwertdilatation, um die mögliche Schrittweite für das Template Matching zu erhöhen. Vorher kann eine Grob lokalisierung des Zeichenfeldes erfolgen, um den Bereich einzuzgrenzen, der gefiltert und durchsucht werden muss. 4. <i>Arbeitsbereiche definieren</i>, setzt einen Suchbereich um das Zeichenfeld. 5. <i>Arbeitsbereiche sortieren</i>, ordnet die gefundenen Ziffern nach x-Koordinaten sortiert an. 6. <i>Arbeitsbereiche rastern</i>, stellt normierte Eingangsbilder für den Klassifikator her. 7. <i>Arbeitsbereiche klassifizieren</i>, erkennt die gefundenen Zeichen anhand der Rasterbilder mittels eines neuronalen Netzes.

5.4 Blockcodierungen auf Filmrollen

Die Anwendung von Klassifikatoren muss sich keineswegs auf die Erkennung von Schrift beschränken. Prinzipiell lässt sich jede Art von Muster auf diese Weise erkennen, wenn die richtigen Merkmale zur Verfügung stehen. Als Beispiel dafür soll hier die Erkennung von Lichtempfindlichkeitscodes auf Kleinbildfilmen dienen. Im Zeitalter der digitalen Fotografie geraten Kleinbildfilme zwar langsam in Vergessenheit, aber zum Einen trifft man sie noch gelegentlich bei Liebhabern an und zum Anderen illustriert dieses Beispiel sehr schön die Erkennung von nicht schriftlicher Kennzeichnung.

Die Codes lassen sich auch nichtoptisch identifizieren, da sie leitfähig sind. So erkennen Kleinbildkameras nach dem Einlegen des Films die Lichtempfindlichkeit, wobei typische Lichtempfindlichkeiten 100, 200 und 400 ASA sind, die in Abbildung 5.29 zu sehen sind. Innerhalb der Fertigung der Filmrollen vor deren Verpackung war jedoch eine schnellere und vor allem berührungslose Prüfung gefragt.



Abbildung 5.29: Filmrollen mit 100, 200 und 400 ASA Lichtempfindlichkeitscodes (von links nach rechts)

Die Herausforderung dieser Aufgabe lag dabei vor allem in der verlangten Prüfgeschwindigkeit von 15 Teilen pro Sekunde, daher ist der Hardwareaufbau hier der interessanter Aspekt, während der Prüfablauf trotz der High-Speed-Verarbeitung relativ schlicht erscheint – wenn man nur die endgültigen Funktionsbausteine betrachtet und die unter der Oberfläche verborgene komplexe Algorithmik außer Acht lässt.

Hardwareaufbau: Bei einer Anlage mit derart schnell bewegten Teilen muss die Bildaufnahme zu einem definierten Zeitpunkt durch einen Trigger ausgelöst werden. Außerdem muss man so kurz wie möglich belichten, um geschwindigkeitsbedingte Verzerrungen auszuschließen. Dies wiederum macht eine entsprechend leistungsstarke Beleuchtung erforderlich, die in der kurzen Zeit genügend Licht auf den Sensor bringt, mit anderen Worten: ein Blitzlicht.

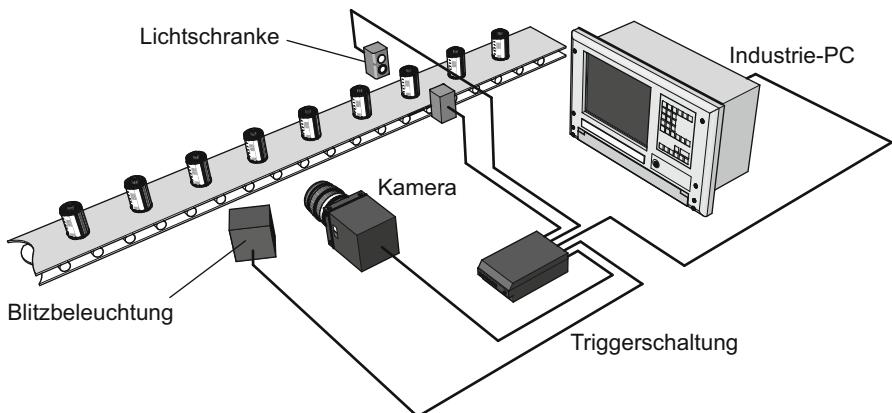


Abbildung 5.30: Hardwareaufbau der Filmrollenprüfung

Die Realisierung ist in Abbildung 5.30 gezeigt. Die Filmrollen durchlaufen eine Lichtschranke, die einen Trigger auslöst, der folgende Ereignisse auslöst:

- die Kamera startet die Bildaufnahme und öffnet den Shutter für eine Millisekunde. Danach wird das aufgenommene Bild an den Rechner übertragen;
- der Blitz beleuchtet nach einer minimalen Verzögerung, in der sich die Kamera auf die Aufnahme vorbereitet, das vorbeilaufende Teil für etwa zehn Mikrosekunden, so dass ein scharfes Standbild der Filmrolle erzeugt wird.

Die gesamte Szene erhält also nur so lange Licht, wie der Blitz tatsächlich leuchtet; die Belichtungszeit ist damit noch einmal deutlich kürzer als die Belichtungszeit der Kamera, so dass Verzerrungen durch die Teilebewegung ausgeschlossen sind. Voraussetzung für das Funktionieren dieses Ablaufs ist natürlich ein sorgfältiger mechanischer Aufbau und die genaue Abstimmung der Triggerlaufzeiten.

Prüfablauf: Der algorithmisch interessante Aspekt dieser Aufgabe liegt in der Anwendung eines Klassifikators auf einen beliebigen Satz von Symbolen anstelle von Klarschrift. Eine Objektsuche ist dabei nur zur Positionskorrektur erforderlich, was der Geschwindigkeit der Anwendung zugutekommt.

Aufgrund der hohen Geschwindigkeit ist trotz der Triggerung die Position der Filmrolle im Bild nicht immer präzise identisch, einige Pixel Abweichung sind unvermeidlich, wobei der mechanische Aufbau keine Variationen in Y-Richtung zulässt. Man benötigt also für die Positionierung nur eine senkrechte Kante, hier die stets deutlich kontrastierende Vorderkante der Filmrolle. Die Feststellung der Kantenposition geht sehr schnell vor sich, da nur auf einer einzigen Linie nach dem Übergang gesucht und ein Stück aus der Kante herausgeschnitten werden muss. Darüber hinaus ist keine Histogrammanalyse erforderlich, da die Blitzbeleuchtung für sehr stabile Kontrastverhältnisse sorgt. Die Binärschwelle für die Kantendetektierung kann also einfach fest eingestellt werden.

Aufgrund der festgestellten Kantenposition wird dann ein Arbeitsbereich über dem Lichtempfindlichkeitscode positioniert. Da der Arbeitsbereich nach der Positionierung genau den zu lesenden Code enthält, ist keine Objektsuche mehr erforderlich, vielmehr wird einfach der Inhalt des Arbeitsbereiches klassifiziert. Wie die Buchstaben und Ziffern in Abschnitt 5.2.2 wird er dafür zunächst in ein Rasterbild umgewandelt. Die Rasterung sorgt hier übrigens nicht in erster Linie dafür, dass der Klassifikator immer gleich viele Eingangssignale erhält. Da der Arbeitsbereich nicht durch Segmentierung gefunden wurde, ist er natürlich immer gleich groß und könnte im Prinzip direkt für die Klassifikation verwendet werden. Dann aber brauchte man einen Klassifikator mit mehreren 10 000 Eingängen, der sehr langsam wäre, da jeder Bildpunkt ein Eingangssignal darstellt. Die Hauptaufgabe der Rasterung liegt hier also in einer Datenreduktion. Abbildung 5.31 zeigt den Lichtempfindlichkeitscode vor und nach der Rasterung mit 10×20 Rasterpunkten.

Nach der Klassifikation des Lichtempfindlichkeitscodes wird dann ein zweiter Arbeitsbereich entsprechend der ermittelten Vorderkante über dem Barcode posi-

tioniert und dieser gelesen. Es handelt sich dabei um einen Code 2/5 Interleaved mit sechs Zeichen.

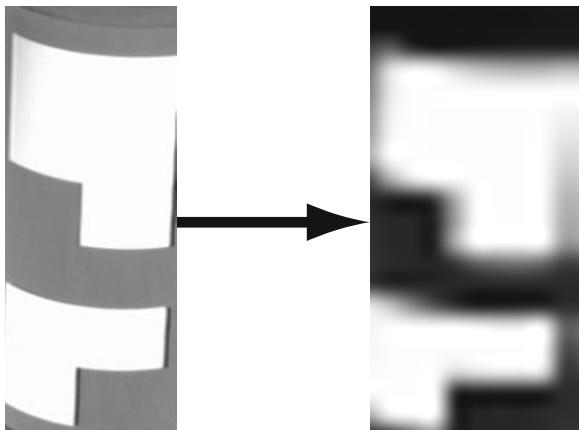


Abbildung 5.31: Lichtempfindlichkeitscode eines 200-ASA-Filmes vor und nach der Rasterung

Geschwindigkeit: Die Aufnahme- und Übertragungszeit lässt sich senken, wenn nur ein Teil der Bildzeilen verwendet wird, was bei der Auflösung des Blockcodes in dieser Anwendung die Lesbarkeit nicht einschränkt. Voraussetzung dafür ist, dass die Kamera eine teilweise Übertragung der Bilddaten erlaubt. Sollte selbst dann die Summe aus Aufnahme-, Übertragungs- und Verarbeitungszeit noch zu groß sein, kann die Kamera bereits das nächste Bild aufnehmen, während der Prozessor die Auswertung des vorherigen Bildes vornimmt. Damit wird die Prüfgeschwindigkeit nicht mehr von der Summe dieser Zeiten limitiert, sondern nur noch von dem langsameren der beiden Vorgänge. Beispielbeschreibung 5.6 fasst diese Anwendung noch einmal zusammen.

Programmhinweis: Für die oben erwähnte Parallelisierung von Bildaufnahme und Verarbeitung ist in **NeuroCheck** der Prüfschritt *Bild aufnehmen parallel* zuständig, der die erforderliche Synchronisation übernimmt.

5.5 Druckqualitätskontrolle

Als Abschluss dieses Kapitels über Kennzeichnungsidentifikation wollen wir noch einige Bemerkungen zu einem Aufgabengebiet machen, in dem man oft mit ähnlichen Prüfteilen zu tun hat, jedoch andere Ziele verfolgt. Bei der Druckqualitätskontrolle geht es nicht darum, die Kennzeichnungen zu lesen, sondern festzustellen, ob sie mit hinreichender Qualität aufgebracht wurden. Derartige Anwendungen findet man typischerweise dort, wo sehr viele identische Teile bedruckt werden und aus Gründen der Lesbarkeit, der Sicherheit oder auch des Fir-

menimages sichergestellt werden muss, dass die Bedruckung einem Idealmuster entspricht. Sehr viele Beispiele sind hierfür in der Pharma- und Lebensmittelindustrie zu finden, wo es meist um Markennamen, Firmenlogos und Produktbezeichnungen geht, ebenso in der Elektronik- und der Automobilindustrie bei der Bedruckung von Bauteilen, Schaltern und Armaturen.

Beispielbeschreibung 5.6. Die folgende Tabelle fasst das Beispiel aus Abschnitt 5.4 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Identifikation
Beleuchtung	Getriggerte Blitzbeleuchtung.
Kamera	Asynchronkamera, Belichtungszeit 1 ms.
Triggerung	Lichtschranke, löst die Bildaufnahme aus und mit minimaler Verzögerung den Blitz.
Prüfschritte	Das Beispiel verwendet folgenden Prüfschrittablauf: <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, Suchlinie zur Kantendetektierung zwecks Positions korrektur 3. <i>Binärschwelle erzeugen</i>, zur Beschleunigung fest eingestellt (möglich wegen stabiler Kontrast Verhältnisse) 4. <i>Arbeitsbereich binär erzeugen</i>, Detektierung der Vorderkante zur Positions korrektur 5. <i>Arbeitsbereiche Position bestimmen</i>, ermittelt X-Position der Vorderkante. 6. <i>Arbeitsbereiche definieren</i>, Rechteck von passender Größe für den Licht empfindlichkeitscode 7. <i>Arbeitsbereiche positionieren</i>, verschiebt das Rechteck entsprechend der Kantenposition. 8. <i>Arbeitsbereiche rastern</i>, erzeugt datenreduziertes Eingangsbild für den Klassifikator. 9. <i>Arbeitsbereiche klassifizieren</i>, erkennt den Lichtempfindlichkeitscode aus dem Rasterbild. 10. <i>Arbeitsbereiche definieren</i>, Rechteck von passender Größe für den Barcode 11. <i>Arbeitsbereiche positionieren</i>, verschiebt das Rechteck entsprechend der Kantenposition. 12. <i>Barcode identifizieren</i>, liest sechs Zeichen im Code 2/5 Interleaved.

5.5.1 Vorgehensweise

Die typische Vorgehensweise bei der Druckqualitätskontrolle unterscheidet sich erheblich von den bisher in diesem Kapitel besprochenen Identifikationsanwendungen. Das Prinzip ist sehr einfach: Man speichert ein Idealmuster, das sogar computergeneriert sein könnte, also nicht von einem real existierenden Meisterteil aufgenommen werden muss, und führt dann einen Bildvergleich durch. Dieser er-

zeugt ein Bild, das alle Abweichungen des aktuellen Prüfteils zu diesem Idealmuster enthält. Dazu kann man das Referenzbild vom aktuellen Bild subtrahieren.

Soweit zum Prinzip. Wenn man jedoch versucht, diese Methode so schlicht in einer realen Anwendung zu benutzen, wird man entdecken, dass stets erhebliche Abweichungen zwischen Musterteil und Prüfteil auftreten. Das ist eine Folge der unvermeidlichen Positionsabweichungen und sonstiger Ungenauigkeiten der Bildaufnahme und -übertragung. Eine Positionsänderung von einem einzigen Bildpunkt erzeugt entlang sämtlicher Kanten auffällige Fehler. Abbildung 5.32 zeigt diesen Effekt. Man beachte, dass das zwischen dem Referenzbild und dem Prüfbild außer der falschen Positionierung keinerlei Unterschied besteht. Es gibt verschiedene Strategien mit diesem Problem umzugehen. In jedem Fall ist aber die erste Bedingung für eine erfolgreiche Druckqualitätskontrolle eine sehr sorgfältige Positionierung des Referenzbildes relativ zum aktuellen Prüfbild (oder umgekehrt).

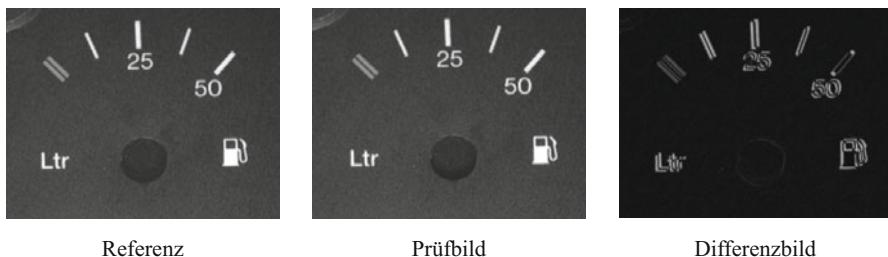


Abbildung 5.32: Effekt kleiner Positionsänderungen auf die Bildsubtraktion

Das weitere Vorgehen ist größtenteils davon abhängig, wie der etwas schwammige Begriff „Druckqualität“ in der jeweiligen Anwendung spezifiziert wurde, welche Fehler also detektiert werden müssen. Es ist daher sehr schwierig, hier allgemeine Angaben zu machen. Einige mögliche Spezifikationen zulässiger und unzulässiger Abweichungen vom Idealmuster sind:

Verdünnung von Zeichen: Solange sie sich in einem gewissen Rahmen halten, sind Verdünnungen von Zeichen, wie etwa das Fehlen von ein oder zwei Pixelspalten am senkrechten Strich einer 1, häufig tolerierbar, da sie die Lesbarkeit nicht beeinträchtigen. Das System muss dann in der Lage sein, solche Abweichungen zu ignorieren.

Löcher in Zeichen: Weiße Flecken in einem schwarz gedruckten Zeichen oder Firmenlogo müssen so gut wie immer als Fehler betrachtet werden. Dasselbe gilt für quer durchlaufende Kratzer, durch die die Zeichen sogar zerfallen.

Flecken im Hintergrund: Diese Art von Fehlern kommt nicht selten bei Tintenstrahldruckern oder Tampondruckern mit verschlissenen Schwämmen vor. Auch derartige Fehler können normalerweise nicht akzeptiert werden.

Es ist, wie gesagt, sehr schwierig, dieses Aufgabengebiet zu verallgemeinern, da die Vorgehensweise sehr stark von den Wünschen des jeweiligen Auftraggebers, den Vorgaben für die Druckqualität und auch dem Druckverfahren abhängt. In vielen Fällen kann man eine solche Aufgabe aber als spezielle Form der Anwesenheitskontrolle auf einem Differenzbild betrachten. Dabei werden die Fehlstellen aus dem Differenzbild als Objekte segmentiert und einzeln bewertet, wodurch man erhebliche Flexibilität beim Umsetzen der aufgabenspezifischen Anforderungen erreicht. Besondere Sorgfalt muss dabei natürlich der Herstellung des Differenzbildes gelten.

Die Bewertung von Objekten im Rahmen einer Anwesenheitskontrolle wird ausführlich im Kapitel 9 besprochen, daher werden wir im Folgenden einfach ein Beispiel für die Herstellung des Differenzbildes und die nachfolgende Bewertung geben.

5.5.2 Druckqualitätskontrolle in Einzelbereichen

Wir wollen kurz am Beispiel der Benzinuhr aus Abbildung 5.32 einen Eindruck vom Ablauf einer Druckqualitätsprüfung geben. Ein möglicher Ansatz ist die Definition einzelner Suchbereiche für jedes zu prüfende Zeichen. Diese können mittels einer Positions korrektur nach demselben Muster wie in den bisherigen Anwendungen dieses Kapitels dem aktuellen Prüfteil nachgeführt werden. Als Referenzobjekt bietet sich dabei der kreisförmige Ausschnitt für die Zeigerachse an, der sich mit Template Matching trotz des geringen Kontrastes sehr leicht segmentieren lässt. In einfachen Fällen kann diese Positionierung bereits genügen, so dass man danach innerhalb der Suchbereiche ein Differenzbild erzeugen und dieses bewerten kann.

Im Allgemeinen wird man aber mit kleinen relativen Verschiebungen innerhalb der Bedruckung rechnen müssen, so dass man für jeden Suchbereich noch einmal einen separaten Positions ausgleich durchführen muss. Dazu kann man sich derselben Methode bedienen wie beim Template Matching: Das Referenzmuster für jeden Suchbereich wird über dem aktuellen Prüfbild verschoben und an jeder Position die Korrelation zum Prüfbild berechnet, bis das Muster an der passenden Stelle mit der höchsten Korrelation „einrastet“. Dabei kann man sich, im Gegensatz zum Template Matching bei der Segmentierung, auf eine kleine Umgebung der Referenzposition dieses Suchbereichs beschränken. Bildlich gesprochen wackelt man solange mit dem Referenzmuster hin und her, bis es optimal zum Bild passt. Abbildung 5.33 zeigt das Ergebnis dieses Verfahrens für ein Benzinuhrmuster, das nicht nur gegenüber dem Referenzmuster global verschoben ist, sondern auch einige echte Fehler aufweist:

- Das „r“ von „Ltr“ fehlt.
- Das Fenster der Zapfsäule ist teilweise verschmiert.
- Der 25-Liter-Strich ist etwas zu schmal.

Die Rahmen der Suchbereiche sind im Bild des fehlerhaften Prüfteils und im Differenzbild weiß eingezeichnet. Man sieht, dass alle Fehler einwandfrei erkannt werden.

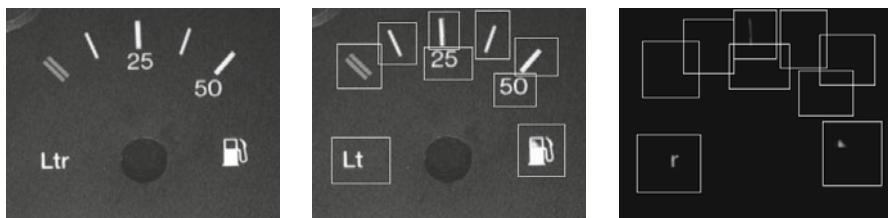


Abbildung 5.33: Effekt kleiner Positionsänderungen auf die Bildsubtraktion

Im nächsten Schritt kann man die Fehler dann als Objekte segmentieren und nach ihren Eigenschaften bewerten. Wir machen hier einen Vorgriff auf das Kapitel 9 über Anwesenheitskontrolle sowie das Kapitel 10 über Objektmerkmale. Als Merkmale eignen sich hier beispielsweise Radien und horizontale/vertikale Ausdehnung und die Fläche. Wenn leichte Verdünnungen eines Zeichens, wie bei dem 25-Liter-Strich, nicht als Fehler angesehen werden sollen, bieten sich Maße wie das Verhältnis von Bogenlänge zu Breite an, aber auch der mittlere Grauwert und der Kontrast, da in den Randbereichen des Drucks gewöhnlich der Farbauftrag geringer ist, so dass sich diese Randfehler deutlich von Fehlern im Innern eines Zeichens unterscheiden.

5.5.3 Druckqualitätskontrolle mit automatischer Teilung

Ein bestimmter Typ von Fehlern lässt sich mit dem eben beschriebenen Ansatz nicht feststellen: zusätzlicher Farbauftrag außerhalb eines der Suchbereiche. Wenn mit derartigen Effekten zu rechnen ist, kann man den gesamten Prüfbereich – in diesem Fall praktisch das ganze Bild – als Suchbereich definieren und diesen Bereich schrittweise immer weiter unterteilen. Mit jedem dieser neuen Unterbereiche kann ebenso verfahren werden wie mit den kleinen Suchbereichen in Abbildung 5.33, d. h. sie werden in kleinen Schritten um ihre ursprüngliche Position herumbewegt, bis die maximale Übereinstimmung mit dem Bild gefunden ist. Durch die Vorgabe des Teilungsfaktors (z. B. Teilung in 2×2 oder 3×3 Unterbereich in jedem Schritt) und eine minimale Größe für diese Bereiche kann die Fehlergröße eingestellt werden, die detektiert werden soll. Abbildung 5.34 zeigt das Ergebnis beider Verfahren, wenn zu den oben beschriebenen Fehlern noch ein heller Fleck links vom Zeigerausschnitt hinzukommt. In den einzelnen zeichenweisen Suchbereichen kann dieser Fehler nicht gefunden werden.

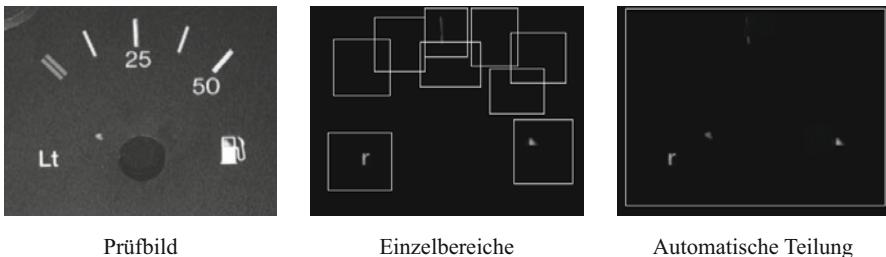


Abbildung 5.34: Druckqualitätskontrolle mit automatischer Teilung in Unterbereiche

Die weitere Prüfung kann dann auf dieselbe Weise in Form einer Anwesenheitskontrolle durchgeführt werden wie zuvor. Beide Methoden, d. h. die globale Positionierung einzelner Suchbereiche und die automatische Teilung jedes Suchbereichs lassen sich verbinden, so dass man sehr flexibel auf unterschiedliche Fehlertypen reagieren kann.

Dies soll als Eindruck der typischen Eigenschaften einer Druckqualitätskontrolle genügen. Wie gesagt, ist diese Gebiet aufgrund der sehr unterschiedlichen Arten von Fehlern kaum zu verallgemeinern. Die Auswertestrategie wird man daher von Anwendung zu Anwendung entscheiden müssen. Es sei noch erwähnt, dass sich die Methodik der Druckqualitätskontrolle auch auf völlig andere Bereiche anwenden lässt, beispielsweise die Kontrolle hybrider Schaltelemente oder Leiterbahnen, also allgemein auf Prüfaufgaben, bei denen die Übereinstimmung des visuellen Erscheinungsbildes eines bekannten Objekts mit einer Referenz sichergestellt werden muss.

5.6 Zusammenfassung

Barcode-Identifikation: Bildverarbeitungssysteme sind flexibler als Barcode-scanner und dadurch in der Lage, Codes auch unter Bedingungen zu erkennen, bei denen ein Scanner versagen würde (schlechter Kontrast, Beschädigungen, geometrische Verzerrungen). Dem stehen jedoch ein höherer Einstandspreis und erheblicher Entwicklungsaufwand für robuste Lesealgorithmen gegenüber, wenn kein fertiges Bildverarbeitungssystem mit eingebauten Verfahren zur Erkennung von Barcodes verwendet wird.

DataMatrix-Code: Wegen der höheren Datendichte und der Möglichkeit zur Fehlerkorrektur werden in der Industriellen Fertigung und der Logistik häufig zweidimensionale Code wie der DataMatrix-Code zur Kennzeichnung eingesetzt. DataMatrix-Codes können mit Bildverarbeitungssystemen auch unter schlechten Bedingungen zuverlässig identifiziert werden.

Entwurf von Klassifikationsanwendungen: Beliebige Zeichen und Symbole können anhand geeigneter Merkmale von einem Klassifikator erkannt werden. Der Aufbau einer Klassifikationsanwendung erfordert die folgenden Schritte:

1. Aufgabenspezifikation
2. Klassendefinition
3. Merkmalsauswahl
4. Zusammenstellung von Trainingsdaten
5. Entwurf der Klassifikatorstruktur
6. Optimierung des Klassifikators (Training)
7. Test des Klassifikators

Merkmalsauswahl, Klassifikatorentwurf, Training und Test bilden im Allgemeinen einen iterativen Prozess.

Ablauf von Klassifikationsanwendungen: Prüfaufgaben aus diesem Bereich folgen meist einem typischen Ablauf (unter Vernachlässigung möglicher Vorverarbeitungsschritte):

1. Setzen von Suchbereichen, ggf. nach vorheriger Positionsbestimmung.
2. Segmentierung, bei nicht zusammenhängenden Objekten mit Template Matching.
3. Beseitigung ungültiger Objekte aufgrund einfacher Eigenschaften und Sortierung.
4. Erzeugung von Merkmalen, die sich für die Klassifikation eignen, beispielsweise eine normierte Bilddarstellung.
5. Klassifikation.

Klassifikator: Ein Klassifikator kann analog zu einem signalverarbeitenden Baustein mit einer definierten Anzahl von Eingängen für die Merkmale und Ausgängen für die Klassen betrachtet werden.

Zeichendarstellung: Für die Erkennung gedruckter Zeichen ist im Allgemeinen eine großen- und kontrastnormierte Bilddarstellung der Zeichen gut geeignet. Handschriften erfordern komplexere Merkmale, die in diesem Rahmen nicht dargestellt werden können.

Beliebige Symbole: Klassifikatoren können nicht nur dem Menschen vertraute Zeichen erkennen, sondern beliebige Symbole.

Vorverarbeitung: Einfache Vorverarbeitungsverfahren, wie z. B. Kontrastnormierungen oder Filterungen, können die Aufgabe des Klassifikators durch Verringerung der Variationen im Erscheinungsbild der Zeichen erheblich vereinfachen.

Optimierung im Betrieb: Änderungen der Aufgabenspezifikation, mangelnde Verfügbarkeit von Serienteilen während der Bildverarbeitungsentwicklung, Produktionsumstellungen und Prozessdrift (z. B. durch Verschleiß oder Dejustierung von Beschriftungsanlagen) erfordern nachträgliche Optimierungen des Prüfablaufs

und des Klassifikators. Hierzu muss die Klassifikationssicherheit ständig überwacht werden können.

Positionierung durch Template Matching: Bei wechselnden Kontrastverhältnissen und Hintergrundeigenschaften sind charakteristische Zeichen, die mit Template Matching gefunden werden können, oft besser zur Ermittlung einer Positionsreferenz geeignet als Objektkanten, da man sich auf eine größere Zahl von Bildpunkten stützen kann.

Nicht zusammenhängende Zeichen: Konturverfolgung erfordert zusammenhängende Zeichen. Für eine sichere Segmentierung von Zeichen, die aus Einzelpunkten zusammengesetzt sind (wie beim Tintenstrahldruck oder Nadelprägung in Metall) muss man daher entweder Template Matching oder durch eine Filterung zusammenhängende Zeichen herstellen.

Druckqualitätskontrolle: Hierunter versteht man die Überprüfung bereits bekannter gedruckter Zeichen oder anderer Objekte mit vorgeschriebenem Erscheinungsbild, wie etwa Schaltelemente, Leiterbahnen, Verpackungen und Produktlabel. Die Anwendungsstruktur entspricht meist einer Anwesenheitskontrolle mit vorheriger Erzeugung eines Differenzbildes. Dabei ist eine sehr genaue Positionierung des gesamten Referenzbildes und/oder Teilen davon unerlässlich. Die genaue Vorgehensweise hängt stark von der jeweiligen Aufgabe und den Fehlerspezifikationen ab.

6. Querschnitt: Klassifikation

Dieses Kapitel gibt eine kurze allgemeine Einführung in das Gebiet der Klassifikation, verbunden mit einer Übersicht über einige wichtige Klassifikatortypen. Es ist natürlich nicht möglich, dieses Gebiet hier auch nur annähernd erschöpfend zu behandeln, zu vielfältig und umfangreich ist die Forschung, die besonders in den letzten 40 Jahren auf diesem Gebiet von so verschiedenen Disziplinen wie der Psychologie und Biologie auf der einen, der Informatik auf der anderen Seite betrieben wurde. Ebenso können wir nicht alle Klassifikatortypen im Detail behandeln und ihre mathematischen Eigenschaften ausführen. Dennoch wollen wir die wichtigen Typen wenigstens erwähnen, um dem Leser eine gewisse Orientierung in diesem für Mustererkennungsaufgaben so wichtigen Bereich zu geben. Wir werden einen dieser Klassifikatortypen, das neuronale Netz vom Multilayer-Perzeptron-Typ, ausführlicher behandeln, da dieser Klassifikator in den in Kapitel 5 beschriebenen Mustererkennungsanwendungen zum Einsatz kam.

6.1 Was ist Klassifikation

Klassifikation ist der zentrale Vorgang bei der Erkennung von Mustern, d. h. dem Bemühen, beobachtete Erscheinungen zu benennen, von der bloß beschreibenden Ebene physikalischer, geometrischer oder sonstiger Merkmale zu einer symbolischen Ebene zu gelangen. Im Rahmen der für uns hier wesentlichen Mustererkennung im technischen Bereich können wir Klassifikation definieren als die Zuordnung eines durch einen Satz von Merkmalen beschriebenen Objekts zu einer von mehreren vorgegebenen Klassen.

Diese Definition macht gewisse Einschränkungen. Es gibt durchaus auch den Fall, dass die möglichen Klassen nicht von vornherein bekannt sind, sondern sich aufgrund von Ähnlichkeitsbeziehungen erst im Laufe der Zeit herausbilden. Die Definition entspricht aber dem für technische Anwendungen typischen Fall. Ihre Bedeutung versteht man vielleicht am besten anhand einiger Beispiele für technische Klassifikationsaufgaben, die in Tabelle 6.1 aufgeführt sind.

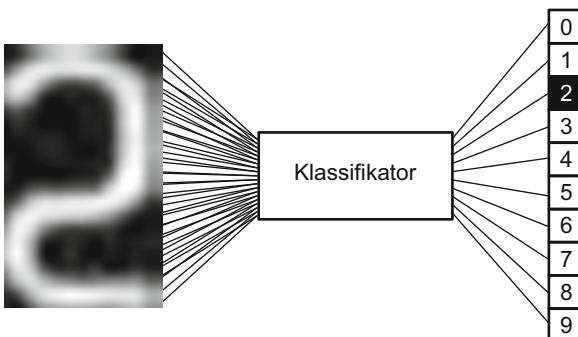
Man sieht, dass es schon im technischen Bereich sehr unterschiedliche Arten von Klassifikationsanwendungen gibt. Ihnen allen ist aber gemeinsam, dass eine Verringerung der Datenmenge stattfindet. Die Beschreibung ist normalerweise recht umfangreich (das Bild eines Zeichens sollte, um gut erkennbar zu sein, wenigstens 400 Bildpunkte aufweisen; die Spektraleigenschaften akustischer Signale werden gewöhnlich mit wenigstens 256 Spektrallinien beschrieben). Jede einzelne Beschreibungskomponente bezeichnet man als Merkmal. Jede Kombination von

Merkmalswerten beschreibt ein bestimmtes Objekt. Die Anzahl der Klassen ist dagegen normalerweise deutlich geringer, es findet also eine Datenreduktion statt. Um Einwänden vorzubeugen: Es kann durchaus auch der umgekehrte Fall vorkommen, beispielsweise lassen sich Schrauben aufgrund eines einzigen Merkmals, ihres Durchmessers, in sehr viele verschiedene Größenklassen einteilen. Beim Einsatz von Klassifikatoren in der industriellen Bildverarbeitung ist eine erhebliche Verringerung der Datenmenge in mehreren Schritten, vom Eingangsbild über die Merkmale bis zur Klassenzuordnung, typisch.

Tabelle 6.1. Beispiele für technische Klassifikationsaufgaben

Anwendung	Merkmale	Klassenbeschreibung
Unterscheidung von Gegenständen (z.B. Schrauben)	<ul style="list-style-type: none"> • Bild des Gegenstandes • Maße des Gegenstandes • Formparameter (z.B. Verhältnis von Umfang und Fläche, Strukturbreite, ...) • ... 	Bezeichnung des Gegenstandes
Unterscheidung von Oberflächenfehlern	<ul style="list-style-type: none"> • Bild der Fehlstelle • Maße der Fehlstelle • Helligkeitseigenschaften der Fehlstelle • ... 	Bezeichnung der Fehlerart (Kratzer, Lunker, ...) und/oder Einstufung des Fehlers als noch zulässig oder unzulässig.
Schrifterkennung	<ul style="list-style-type: none"> • Bild des Zeichens • Fourierkoeffizienten des Zeichens • Formmerkmale des Zeichens (wie Strecken, Krümmungen, ...) • ... 	Bezeichnung des Buchstabens oder Nummer des Zeichens in einem gegebenen Zeichensatz.
Geräuschprüfung eines Elektromotors	<ul style="list-style-type: none"> • Zeitverlauf des Schallsignals • Spektraleigenschaften des Schallsignals • ... 	Einstufung des Motors als einwandfrei oder fehlerhaft

Wir sagten eingangs, dass man unter einer Klassifikation die Zuordnung eines Objekts, das durch einen Satz von Merkmalen beschrieben wird, zu einer bestimmten Klasse versteht. Man kann dies auch formulieren als die Durchführung einer Abbildung aus einem Merkmalsraum in einen Klassenraum. Aus der Klassifikation wird so im mathematischen Sinne eine Abbildungsfunktion, die mittels einer Rechenvorschrift durchgeführt werden kann. Abbildung 6.1 veranschaulicht diesen Zusammenhang noch einmal am Beispiel der Ziffernerkennung.



Merkmalsraum → Abbildungsfunktion → Klassenraum

Abbildung 6.1: Klassifikation als Abbildung zwischen Vektorräumen

Im Folgenden werden wir zunächst einige theoretische Grundlagen einführen, ohne dabei allzu tief in mathematische Details zu gehen. Dann werden wir verschiedene Klassifikatortypen vorstellen und uns schließlich mit einem davon, neuronalen Netzen eines bestimmten Typs, etwas ausführlicher beschäftigen.

6.2 Klassifikation als Funktionsapproximation

Dieser Abschnitt fasst einige theoretische Grundlagen aus dem Bereich der statistischen Mustererkennung zusammen. Wie wir im vorigen Abschnitt ausführten, kann man eine Klassifikation als Abbildung aus einem Merkmals- in einen Klassenraum betrachten, d. h. als eine Funktion. Diese Funktion ist normalerweise nicht bekannt, denn wir wissen nicht, wie ein Mensch aus dem Bild eines Buchstabens die Information gewinnt, um welchen Buchstaben es sich handelt. Bei der automatischen Klassifikation mithilfe eines Computers wird also versucht, diese unbekannte Funktion mit numerischen Mitteln anzunähern.

Hier stellt sich sogleich die Frage, welche Funktion eigentlich angenähert werden soll. Je nach Aufgabenstellung kann es darauf verschiedene Antworten geben. In Bildverarbeitungsanwendungen besteht die Aufgabe jedoch üblicherweise darin, eine vom Menschen vorgegebene Zuordnung maschinell nachzuvollziehen. Von einem Schrifterkennungssystem wird man beispielsweise erwarten, dass es dem Bild eines Zeichens genau das Zeichen zuordnet, das auch der Mensch in diesem Bild erkennen würde. Entsprechend wird man auch erwarten, dass ein Klassifikationssystem für Oberflächenfehler einen zu großen Kratzer ebenso zurückweist, wie dies ein menschlicher Prüfer tun würde. Die Aufgabe eines Klassifikators besteht demnach darin, eine vom Menschen vorgeschriebene Zuordnung zwischen Eingangsmerkmalen und Klassenbezeichnung nachzuvollziehen. Damit wissen wir aber noch immer nicht, wie wir zu einer Klassifikations-

funktion kommen, die ein Computer berechnen kann. Vor der Klärung dieser Frage müssen wir einige Grundbegriffe einführen.

6.2.1 Grundbegriffe

Es gibt in der Technik sehr viele Aufgaben, bei denen die zulässigen Merkmalsbereiche leicht zu quantifizieren sind; die Toleranzen auf einer technischen Zeichnung und die Definitionen für die Oberflächengüte lassen sich relativ leicht nachprüfen. Viele der Bewertungen und Klassifikationen, die Menschen ganz selbstverständlich vornehmen, lassen sich hingegen praktisch nicht formalisieren, schon allein, weil Menschen im Allgemeinen gar nicht wissen oder genau beschreiben können, welche Kriterien sie dabei anwenden.

Wie beschreibt man etwa den Unterschied zwischen einem 'F' und einem 'E' so, dass eine Maschine ihn versteht? Kein Problem, das 'F' hat einen Strich weniger, wird man vielleicht sagen, aber das ist auch nicht die Lösung, denn nun muss man dem Computer klarmachen, dass die scharfe 90°-Ecke am 'E' einen neuen Strich einleitet. Nach einer Segmentierung ist das ganze Zeichen ja zunächst einmal ein zusammenhängendes Objekt. Menschen hingegen formen die Abbildungsvorschriften, die sie bei dieser Klassifikation verwenden, durch Lernen aus Beispielen. Man muss nur oft genug ein 'E' und ein 'F' gesehen haben, um sie auseinander halten zu können – und kommt dann von ganz allein zu dem Schluss, dass dem einen Buchstaben ein Strich fehlt. Sollte es also nicht möglich sein, die Ableitung einer Abbildungsvorschrift aus Beispielen auch maschinell durchzuführen?

In der Tat ist dies durchführbar, und man bezeichnet diesen Vorgang als *maschinelles Lernen*. Dieser Lernvorgang beruht auf einer Sammlung von Beispielen für die korrekte (heißt: vom Menschen vorgegebene) Zuordnung zwischen Objektbeschreibungen und Klassen. Der maschinelle Klassifikator „lernt“, diese Zuordnungen nachzuvollziehen und auf neue Objekte zu übertragen. Führen wir zunächst einige Begriffe ein:

Muster: Unter einem Muster verstehen wir die Beschreibung eines zu klassifizierenden Objekts. Die Repräsentation eines Musters kann je nach Aufgabenstellung ganz verschieden ausfallen, es kann sich z. B. um ein normiertes Bild eines Zeichens handeln, aber auch um einen Satz vergleichsweise unanschaulicher Messwerte.

Merksmal: Unter einem Merkmal verstehen wir einen Wert innerhalb eines Musters. Wenn das Objekt durch ein Bild beschrieben wird, so ist die Helligkeit eines bestimmten Punkts in diesem Bild ein Merkmal. Fläche, Umfang, mittlere Krümmung sind andere Beispiele für Merkmale.

Merkmalsvektor: Alle Merkmale, die ein Objekt beschreiben, werden in einem Merkmalsvektor zusammengefasst. Merkmalsvektor ist also in diesem Fall gleichbedeutend mit *Muster*. Da der Merkmalsvektor den Eingang des Klassika-

tors darstellt, bezeichnen wir ihn mit i für *Input*. Ein Muster wird also durch einen Vektor aller Merkmale dargestellt:

$$\mathbf{i} = \begin{pmatrix} i_0 \\ i_1 \\ \vdots \\ i_N \end{pmatrix}. \quad (6.1)$$

Klassenindex: Der Klassenindex ist die Nummer der Klasse, der ein Eingangsmuster zugeordnet wurde, sei es von dem Menschen, der als Trainer fungiert, oder vom Klassifikator selbst.

Zielvektor: Der Zielvektor ist ein Vektor, der ausschließlich Nullen enthält, mit einer Ausnahme: Das Element, das dem Klassenindex der Klasse des Eingangsmusters entspricht, ist eins. Wir bezeichnen den Zielvektor mit t für *target*. Ein Zielvektor sieht also folgendermaßen aus:

$$\mathbf{t} = (0, \dots, 0, 1, 0, \dots, 0), \quad (6.2)$$

wobei das Element mit dem Wert eins gerade das Element t_k ist, also das dem Klassenindex entsprechende Element.

Ergebnisvektor: Der Ergebnisvektor enthält das Ergebnis des Klassifikationsvorgangs. Die meisten Klassifikatoren liefern (zumindest intern) für jede der möglichen Klassen einen Ausgangswert, der angibt, mit welcher Wahrscheinlichkeit das Muster der betreffenden Klasse angehört. Im Idealfall ist dieser Vektor genau gleich dem Zielvektor, gewöhnlich wird er jedoch gewisse Abweichungen zeigen. Wir bezeichnen den Ergebnisvektor mit \mathbf{o} für *Output*.

Lernstichprobe: Die Lernstichprobe, oftmals auch als *Trainingsdatensatz* bezeichnet, ist eine Sammlung von Mustern mit zugehörigen Zielvektoren (oder Klassenindizes), die angeben, welche Klasse das jeweilige Muster *tatsächlich* angehört. Die Lernstichprobe beschreibt also die Aufgabe, die der Klassifikator zu erlernen hat.

Eine Klassifikation ist entsprechend diesen Begriffen eine Abbildung aus einem Raum von (kontinuierlichen) Merkmalen in einen Raum von (diskreten) Klassen. Die Beschreibungen „kontinuierlich“ und „diskret“ sind in Klammern gesetzt, da es hier weniger um die tatsächliche mathematische Form der Werte geht als darum, wie solche Aufgaben typischerweise betrachtet werden. Man verwendet durchaus diskrete Merkmale, in der Bildverarbeitung sogar sehr häufig, denn die Grauwerte eines digitalen Bildes nehmen ja auch nur diskrete Werte an. Andererseits können Klassifikatoren auch kontinuierliche Ausgangswerte liefern. Bei der üblichen Herangehensweise behandelt man jedoch die Merkmale als reelle Zahlen, auch dann, wenn sie nur ganz bestimmte Werte annehmen können, d. h. man

rechnet ganz einfach mit diesen Werten. Die Ergebnisdarstellung hingegen erfolgt letztlich in Form eines Klassenindexes, der nur diskrete Werte annehmen kann, obwohl der Klassifikator selbst als Rechenergebnis einen beliebigen Wahrscheinlichkeitswert liefert hat.

6.2.2 Statistische Grundlagen

Wir sagten im vorigen Abschnitt, dass das Ergebnis eines Klassifikators angibt, mit welcher Wahrscheinlichkeit ein Muster einer der gegebenen Klassen angehört, und haben uns auch sonst einer statistischen Terminologie bedient. Klassifikation und Mustererkennung scheinen demnach in den Bereich der mathematischen Statistik zu fallen. Warum ist das so?

Wie wir oben gesehen haben, ist es in der Regel nicht möglich, die Abbildung aus dem Merkmals- in den Klassenraum formal direkt zu beschreiben. Daher wird man versuchen, diese Abbildung anhand von Beispielen anzunähern. Diese Beispiele geben aber nur einen Ausschnitt des Problems wieder. Wenn Sie vor der Aufgabe stehen, einen Klassifikator zu bauen, der anhand eines Bildes aus nur 200 Punkten einen Buchstaben identifiziert, so gibt es bereits 256^{200} Kombinationen unterschiedlicher Grauwerte in diesem Bild. Das ergibt etwa $4 \cdot 10^{481}$ unterschiedliche Eingangsmuster, bei weitem mehr als das Universum Moleküle enthält. Natürlich sind sich die meisten dieser Muster sehr, sehr ähnlich und die bei weitem überwiegende Mehrheit wird mit keinem Buchstaben auch nur die geringste Ähnlichkeit haben. Es erweist sich aber als praktisch undurchführbar, mehr als einen winzigen Ausschnitt des Problems in den Trainingsmustern zu erfassen. Das gilt noch mehr für Aufgabenstellungen mit tatsächlich kontinuierlichen Merkmalswerten, bei denen dann die Zahl der möglichen Kombinationen nicht nur praktisch, sondern auch mathematisch unendlich groß ist.

Es ist daher erforderlich, eine Auswahl zu treffen, die Lernstichprobe, und aus dieser Stichprobe auf das Verhalten des Gesamtproblems zurück zu schließen. Hierbei kommt es zwangsläufig zu Unsicherheiten, da die Lernstichprobe das Problem ja nicht vollständig beschreiben kann. Natürlich wird man versuchen, sie so zu wählen, dass sie das Problem möglichst gut präsentiert, aber der Klassifikator wird stets gezwungen sein, bei neuen Mustern zu extrapolieren, aus seinem Wissensschatz zu schließen, mit welcher Wahrscheinlichkeit das neue Muster in eine der bekannten Klassen gehört.

Die folgenden Ausführungen wollen keine Einführung in die statistische Entscheidungstheorie sein, dies würde den Rahmen dieses Buches sprengen. Vielmehr sollen sie ein Gefühl für die allgemeinen Eigenschaften maschinelaler Lern- und Klassifikationsprozesse geben. Wir argumentieren dabei bewusst eher qualitativ und verzichten auf eine Reihe mathematischer Details.

Die Lernstichprobe ist ein Ausschnitt des Gesamtproblems. Wir können dieses Gesamtproblem als einen Prozess betrachten, der andauernd Muster erzeugt. Eine beliebige Kombination von Merkmals- und Zielvektor kommt dabei mit einer Wahrscheinlichkeit vor, die eine Funktion beider Vektoren ist, der Verbund Wahr-

scheinlichkeit¹ $P(i, o)$. Viele dieser Wahrscheinlichkeiten werden natürlich null sein, tatsächlich sogar die allermeisten.

Bei der Klassifikation geht es nun darum, auf der Grundlage eines Merkmalsvektors i zu entscheiden, welcher Klasse das Muster angehört, d. h. welcher Zielvektor o diesem Merkmalsvektor entspricht. Die Grundlage hierfür bildet die *bedingte Wahrscheinlichkeit* $P(o|i)$, die angibt, mit welcher Wahrscheinlichkeit die durch o beschriebene Klasse vorliegt, unter der *Bedingung*, dass der Eingangsvektor i auftritt. Man bezeichnet diese Wahrscheinlichkeit auch als *a posteriori* oder *Rückschlusswahrscheinlichkeit*, da man durch sie aus dem Merkmalsvektor auf die Klasse zurückschließen kann. Diese bedingte Wahrscheinlichkeit ist mit der Verbindewahrscheinlichkeit über den Satz von Bayes folgendermaßen verknüpft:

$$P(i, o) = P(o | i) \cdot P(i). \quad (6.3)$$

In Worten heißt das, die Wahrscheinlichkeit für eine beliebige Kombination aus Merkmals- und Zielvektor ergibt sich als Produkt der Wahrscheinlichkeit, dass der Merkmalsvektor i auftritt und der Wahrscheinlichkeit, dass unter dieser Bedingung die Klasse o vorliegt.

6.2.3 Konstruktion von Klassifikatoren

Die Aufgabe, der sich der Konstrukteur eines Klassifikators gegenüber sieht, ist, dafür zu sorgen, dass der Klassifikator die gewünschte Abbildung vom Merkmalsraum in den Klassenraum möglichst gut durchführt. Die Abbildung ist durch die Lernstichprobe vorgegeben. Bleibt zu klären, was *möglichst gut* hier bedeutet.

Der Klassifikator realisiert eine Entscheidungsfunktion $d(i)$ (für *Decision*), die aus dem Merkmalsvektor i eine Schätzung \hat{o} des wahren Zielvektors o errechnet:

$$d(i) = \hat{o}. \quad (6.4)$$

Diese Entscheidungsfunktion d ist tatsächlich ein Vektor von k Funktionen, eine Funktion für jede Klasse. Für einen gegebenen Eingangsvektor entsteht also ein Vektor, dessen Werte d_k jeweils die Wahrscheinlichkeit angeben, dass der Eingangsvektor zur Klasse k gehört. Das Muster wird dann normalerweise in die Klasse eingeordnet, für die der maximale Wert errechnet wurde.

Die Schätzung \hat{o} des Zielvektors wird unweigerlich vom wahren Wert o abweichen. Die Aufgabe bei der Konstruktion eines Klassifikators besteht folglich darin, diese Abweichung zu minimieren. Dazu allerdings muss man sie zunächst einmal messen. Was man mithilfe einer Kostenfunktion macht. Je nach Wahl der Kosten-

¹ Bei kontinuierlichen Variablen müsste man eigentlich von Wahrscheinlichkeitsdichten sprechen, es ändert aber nichts an der grundsätzlichen Argumentation.

funktion entstehen ganz verschiedene Klassifikatoren, von denen hier nur einer erwähnt werden soll, der eine wichtige theoretische Bedeutung hat.

Der Bayes-Klassifikator: Eine sehr naheliegende Kostenfunktion bewertet sämtliche Fehlklassifikationen mit einem Kostenwert c_f und alle Rückweisungen, d. h. Fälle, in denen keine Klassifikation möglich ist, mit einem Wert c_r . Eine korrekte Klassifikation erzeugt dagegen keine Kosten. Der entstehende Klassifikator wird als *Bayes-Klassifikator* bezeichnet und ist der Klassifikator mit *minimaler Fehlerwahrscheinlichkeit*. Dieser positiven Eigenschaft steht das Problem gegenüber, dass er in der Regel nicht konstruiert werden kann, da er die vollständige Kenntnis der statistischen Eigenschaften des Gesamtproblems voraussetzt, d. h. die im vorigen Abschnitt angesprochenen Wahrscheinlichkeitsverteilungen müssen bekannt sein.

In der Praxis hat man niemals den Luxus derart weit reichender Informationen über den Prozess, daher muss man zu Näherungsverfahren greifen. Ein möglicher Weg besteht darin, bestimmte Wahrscheinlichkeitsverteilungen anzunehmen. Man kann z. B. die Annahme treffen, Merkmalsvektoren gehorchten innerhalb jeder Klasse einer N -dimensionalen Normalverteilung (mit N als der Anzahl der Merkmale, d. h. der Dimension des Merkmalsraums). Unglücklicherweise ist dieser Klassifikator bei den hochdimensionalen Merkmalsräumen, mit denen man in der Bildverarbeitung gewöhnlich zu tun hat, kaum zu handhaben und auch die zugrunde liegende Annahme ist praktisch nicht nachprüfbar.

Praktisches Vorgehen: Zur Konstruktion eines praktisch verwendbaren Klassifikators kann man nun zwei Wegen einschlagen, die auf zwei verschiedene Grundtypen von Klassifikatoren führen:

1. Man speichert eine Anzahl typischer Muster, sog. *Prototypen* für die einzelnen Klassen und vergleicht die beobachteten Merkmalsvektoren mit diesen Mustern. Die Klasse des Merkmalsvektors bestimmt sich aus seiner Ähnlichkeit zu den einzelnen Prototypen. Diese Klassifikatoren unterscheiden sich danach, wie die Prototypen bestimmt werden und wie das Ähnlichkeitsmaß berechnet wird. Derartige Klassifikatoren werden als *Multireferenzenklassifikatoren* bezeichnet (siehe Abschnitt 6.3).
2. Man macht einen Funktionenansatz für die Abbildungsfunktion zwischen Merkmals- und Klassenraum und stellt die Parameter dieses Ansatzes so, dass ein Optimalitätskriterium für alle Muster aus der Lernstichprobe minimiert wird. Üblicherweise verwendet man dazu die mittlere quadratische Abweichung zwischen dem Ergebnisvektor \mathbf{o} des Klassifikators und Zielvektor t :

$$E(|\mathbf{t} - \mathbf{o}|^2) = \min . \quad (6.5)$$

Klassifikatoren, die mit einem Funktionenansatz arbeiten, unterscheiden sich nach der Art der verwendeten Funktionen, deren Verknüpfung und der Art und Weise, in der die Optimierung durchgeführt wird. Ein Beispiel für diesen Typ

sind Polynomklassifikatoren, ein anderes die künstlichen neuronalen Netze, die im Abschnitt 6.4.2 ausführlich behandelt werden.

6.3 Multireferenzenklassifikatoren

Multireferenzenklassifikatoren verwenden eine Anzahl von Prototypmustern für die einzelnen Klassen und ordnen ein beobachtetes Eingangsmuster derjenigen Klasse zu, deren Prototypmustern es am ähnlichsten ist. Damit nehmen sie den Ansatz, aus Beispielen zu lernen, ganz wörtlich: Der Klassifikator wird durch eine Reihe von Beispielen definiert. Die verschiedenen Typen von Multireferenzenklassifikatoren unterscheiden sich danach, wie die Prototypmuster bestimmt werden und wie die Ähnlichkeit zwischen Prototyp und Eingangsmuster berechnet wird.

6.3.1 Nächster-Nachbar-Klassifikator

Der einfachste Multireferenzenklassifikator ist der Nächster-Nachbar-Klassifikator. Er benutzt einfach die gesamte Lernstichprobe als Referenz. Jedes Muster der Lernstichprobe wird also zu einem Prototypmuster, das wir hier mit p_i bezeichnen. Als Abstandsmaß wird der euklidische Abstand verwendet. Der Abstand d_j zwischen einem Eingangsvektor i und einem Prototypmuster p_j ist dann:

$$d_j = \|i - p_j\| = \sqrt{\sum_{k=0}^N (i_k - p_{jk})^2}. \quad (6.6)$$

Darin sind die i_k die einzelnen Merkmalswerte des Eingangsvektors i , die p_{jk} die Merkmale des Prototypmusters p_j . Vom Standpunkt einer Funktionsapproximation aus betrachtet, wird die Klassifikationsfunktion durch alle jene Punkte angenähert, in denen sie genau bekannt ist, nämlich den Prototypmustern mit ihrer bekannten Klassenzugehörigkeit.

Jedes Eingangsmuster wird mit allen Prototypmustern auf diese Weise verglichen. Es wird dann der Klasse des Prototypen zugeordnet, dem es am nächsten ist, für den der euklidische Abstand nach Gleichung (6.6) minimal wird. Genau hier liegt aber das Problem des Nächster-Nachbar-Klassifikators: Jeder Klassifikator kann mit zunehmendem Umfang der Lernstichprobe die Klassifikationsaufgabe besser kennen lernen. Beim Nächster-Nachbar-Klassifikator steigt dabei jedoch nicht nur der Lernaufwand, sondern in gleichem Maße auch der Rechenaufwand bei der Reklassifikation, da das Eingangsmuster mit allen vorhandenen Prototypen verglichen werden muss.

Man kann zeigen, dass der Nächster-Nachbar-Klassifikator bei unbegrenzter Lernstichprobe eine Fehlerrate erreicht, die weniger als doppelt so groß ist wie die des Bayer-Klassifikators als des theoretisch bestmöglichen Klassifikators. Natürlich sind unbegrenzte Lernstichproben prinzipiell unmöglich, aber ein Einsatz des Nächster-Nachbar-Klassifikators wird schon wesentlich früher aufgrund des Rechenaufwandes für den Vergleich mit allen Prototypen unpraktikabel. Vor allem die Annäherung komplizierter, ineinander verschachtelter Klassengrenzen mithilfe einzelner Prototypen ist sehr schwierig, wie Abbildung 6.2 für eine Klassifikation mit zwei Merkmalen zeigt. Die Prototypen der Klasse eins sind als schwarze, die der Klasse zwei als weiße Kreise dargestellt. Das schwarze Kreuz repräsentiert ein Eingangsmuster der Klasse eins. Es liegt dicht an der Klassengrenze und, wie der schwarze Kreis um das Muster zeigt, ist der nächste Prototyp ein Mitglied der Klasse zwei. Man sieht also, dass die Prototypen an komplizierten Klassengrenzen sehr dicht liegen müssen, um dort eine einwandfreie Klassifikation zu erreichen.

Merkmal 2

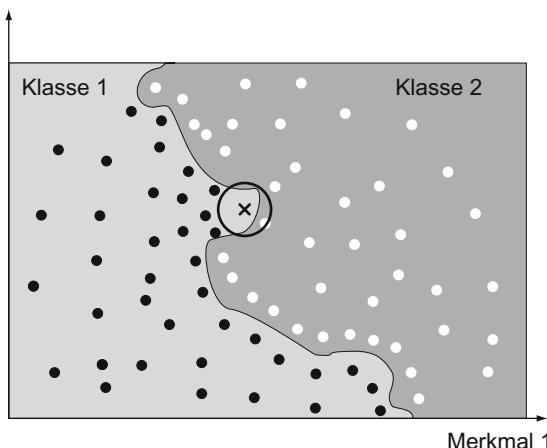


Abbildung 6.2: Annäherung komplexer Klassengrenzen durch einen Nächster-Nachbar-Klassifikator

Randprobleme: In einem Nächster-Nachbar-Klassifikator steckt jeder Prototyp einen gewissen Bereich des Merkmalsraums ab. Die Größe dieses Bereiches hängt von den Entfernungen des Prototypen zu seinen Nachbarprototypen ab, die Begrenzung ist jeweils die Mittelsenkrechte auf diesen Abstand. Alles was innerhalb dieses Bereiches um einen Prototypen liegt, wird seiner eigenen Klasse zugeordnet. Ein Problem entsteht am Rand des Merkmalsraums: Die Bereiche dehnen sich dort einfach bis ins Unendliche aus, erfassen also auch Merkmalskombinationen, die mit Sicherheit kein sinnvolles Muster mehr ergeben. Abbildung 6.3 veranschaulicht dies anhand eines Ausschnitts aus Abbildung 6.2: Dort, wo noch benachbarte Prototypen vorhanden sind, haben die Bereiche sinnvolle Grenzen, am oberen Rand hingegen dehnen sie sich ins Unendliche aus. Das kann zu recht

willkürlichen Klassifikationsergebnissen bei Eingangsmustern führen, die außerhalb des von Prototypen abgedeckten Bereichs liegen.

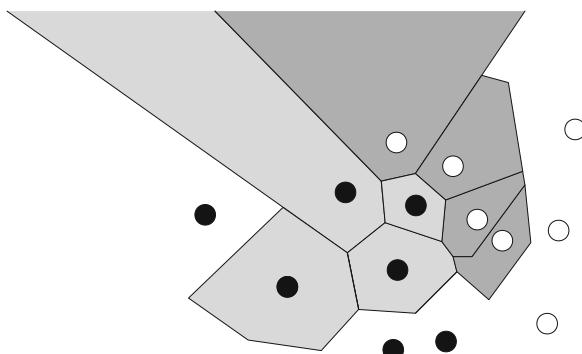


Abbildung 6.3: Randproblematik bei Nächster-Nachbar-Klassifikatoren

6.3.2 RCE-Netze

Die Randproblematik des Nächster-Nachbar-Klassifikators, die in Abbildung 6.3 dargestellt ist, lässt sich vermeiden, wenn man jedem Prototypen eine Einfluss- sphäre zuordnet. Ein Eingangsmuster wird in diesem Fall nur dann einer Klasse zugeordnet, wenn es von dem ihm nächsten Prototypen p_i höchstens einen Abstand R_i hat. Wie Abbildung 6.4 zeigt, entspricht dies einer Annäherung der Klassenverteilung durch überlappende Kreise, bei Verwendung von mehr als zwei Merkmalen entsprechend Kugeln bzw. Hyperkugeln. Dabei ergibt sich zugleich der Nebeneffekt, dass die Zahl der Prototypen gegenüber dem Nächster-Nachbar-Klassifikator deutlich reduziert werden kann, es sind nämlich nur noch die durch Kreuze gekennzeichneten Mittelpunkte der Kreise als Prototypen erforderlich.

Ein Eingangsmuster i wird in diesem Modell derjenigen Klasse zugewiesen, zu der der nächstliegende Prototyp p_j gehört, vorausgesetzt es gilt:

$$d_j = \|i - p_j\| = \sqrt{\sum_{k=0}^N (i_k - p_{jk})^2} \leq R_j . \quad (6.7)$$

Man kann diese Vorgehensweise als *Nächster-Nachbar-Klassifikator mit Rückweisungsschwelle* bezeichnen: Ein Eingangsmuster, das keinem Prototypen ausreichend nahe kommt, wird nicht klassifiziert, sondern zurückgewiesen. Bekannter ist sie unter dem Namen eines Typs neuronaler Netzwerke, der nach diesem Prinzip arbeitet. In Analogie zur Kraftwirkung zwischen geladenen Teilchen, die mit dem Quadrat der Entfernung abnimmt, werden diese Netze als Restricted-

Coulomb-Energy (abgekürzt RCE) *Netzwerk*, (Reilly et al. 1982), bezeichnet. Die Randproblematik des Nächster-Nachbar-Klassifikators tritt bei diesen Klassifikatoren nicht auf, da ein Eingangsmuster, das weit außerhalb des durch Prototypen abgedeckten Bereichs liegt, keinem Prototypen nahe genug ist. Man sieht allerdings auch, dass die Abbildung komplexer Klassengrenzen nach wie vor eine hohe Prototypendichte verlangt.

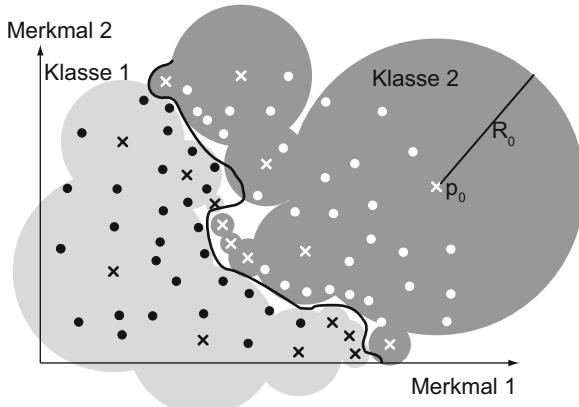


Abbildung 6.4: Nächster-Nachbar-Klassifikator mit Rückweisungsschwelle / RCE-Netzwerk

6.3.3 Vektorquantisierung

Im Zusammenhang mit dem RCE-Netz haben wir gesehen, dass sich die Anzahl der erforderlichen Prototypen gegenüber dem Nächster-Nachbar-Klassifikator unter Umständen drastisch reduzieren lässt. Beim RCE-Netz wurden vorhandene Prototypmuster als Zentren von Kreisen (bzw. Hyperkugeln bei höherdimensionalen Problemen) benutzt, deren Radien dann so eingestellt wurden, dass sie den Bereich einer Klasse möglichst gut abdeckten. Man kann stattdessen auch die Lage der Zentren verändern, die dann nicht mehr vorhandenen Prototypmustern entsprechen, sondern eine Art von „Durchschnittsmuster“ darstellen, das die jeweiligen Klassen – oder einen Ausschnitt davon – möglichst gut repräsentiert.

Diese Aufgabe lässt sich folgendermaßen formulieren: Finde einen Satz von Merkmalsvektoren q_j so, dass der Fehler möglichst klein wird, der entsteht, wenn jeder in der Lernstichprobe vorhandene Vektor i_k durch den nächstliegenden Vektor q_j ersetzt wird. Je nach Verfahren kann dabei die Anzahl der entstehenden Repräsentanten vorgegeben sein oder dynamisch bestimmt werden. Man bezeichnet derartige Methoden als *Vektorquantisierung*, da der gesamte Raum aller möglichen Merkmalsvektoren durch eine diskrete, also quantisierte Menge von Zentrumsvektoren angenähert wird. Ausführliche Informationen hierzu sind beispielsweise in (Kohonen 1989) zu finden.

6.3.4 Template Matching

Gewöhnlich wird Template Matching nicht im Zusammenhang mit Klassifikation betrachtet, sondern mit Segmentierung, wie wir das auch im Abschnitt 4.5 getan haben. Der Mechanismus ist jedoch genau derselbe wie der eines Nächster-Nachbar-Klassifikators mit einer zusätzlichen Rückweisungsschwelle: Jeder Bildausschnitt wird mit allen vorhandenen Templates verglichen und der Klasse des ähnlichsten Templates zugeordnet. Es gibt dabei zwei Unterschiede zum Nächster-Nachbar-Klassifikator:

Abstandsmaß: An die Stelle des euklidischen Abstandes nach Gleichung (6.6) tritt die Korrelation zwischen Template und Bildausschnitt nach Gleichung (4.1).

Rückweisungsschwelle: Gewöhnlich verlangt man bei der Segmentierung eine Mindestähnlichkeit zwischen Bildausschnitt und Template. Wird dieser Korrelationswert nicht erreicht, so wird überhaupt keine Klassenzuweisung vorgenommen.

Man sieht also, dass das Template-Matching-Verfahren genauso wie der Nächster-Nachbar-Klassifikator eingesetzt werden kann. Das bedeutet auch, dass sich das Verhalten bezüglich der Rechenzeit bei vielen Prototypmustern und an komplexen Klassengrenzen ähnelt. Wir werden auch in Abschnitt 9.3 noch sehen, dass die Trennschärfe des Template Matching aufgrund der Eigenschaften der Korrelation nicht besonders hoch ist. Das Verfahren eignet sich daher nur für Muster hoher Qualität, die sich gut unterscheiden lassen.

6.3.5 Bemerkungen zu Multireferenzenklassifikatoren

Wir haben in diesem Abschnitt verschiedene Typen von Klassifikatoren vorgestellt, die alle mit Prototypmustern arbeiten. Zu jedem Prototypmuster gehört dabei ein durch ein Abstandsmaß und ggf. eine Rückweisungsschwelle definierter Bereich des Merkmalsraums. Man könnte diesen Klassifikatortyp daher auch als *integrierend* bezeichnen, da jeder Prototyp jeweils einen gewissen Bereich des Merkmalsraums in seine Klasse einbezieht.

Weiterhin haben die verschiedenen Multireferenzenklassifikatoren gemeinsam, dass sie in der Regel sehr schnell zu konstruieren und zu trainieren sind. Am deutlichsten ist dies beim Nächster-Nachbar-Klassifikator und dem Template Matching, die abgesehen vom Abstandsmaß auf die gleiche Weise funktionieren.

Klassifikatoren dieses Typs sind für hochdimensionale und stark nichtlineare Probleme wenig geeignet, da sie für komplexe Klassengrenzen sehr viele Prototypen und einen entsprechend hohen Rechenaufwand benötigen. Ihr Hauptnachteil liegt aber darin, dass sie keine Gewichtung der Merkmale vornehmen. Wenn beispielsweise ein Nächster-Nachbar-Klassifikator als Merkmalsvektor ein Rasterbild verwendet, so ist für ihn jeder Punkt gleich wichtig, da er den gleichen Beitrag zum euklidischen Abstand zwischen Muster und Prototyp leistet. Er wird also Abweichungen in unbedeutenden Musterbereichen ebenso bewerten wie signifi-

kante Abweichungen. Leichtes Rauschen im Hintergrund kann sich zu denselben Abweichungen summieren wie das Fehlen eines ganzen Striches in einem Zeichen. Kleine Änderungen können daher zu großen Verschiebungen des Klassifikationsergebnisses führen.

Natürlich kann man dagegen prinzipiell Abhilfe schaffen, indem man entsprechende Gewichtungen und Merkmalsbewertungen zusätzlich einführt. Der Aufwand dafür ist jedoch sehr hoch, denn man muss zunächst einmal analysieren, welche Merkmale im Kontext der Anwendung und des jeweiligen Trainingsdatensatzes überhaupt signifikant sind. Die Funktionalklassifikatoren, die wir im nächsten Abschnitt betrachten werden, erweisen sich demgegenüber als wesentlich robuster, vor allem durch die für das Training verwendeten Optimierungsansätze.

6.4 Funktionalklassifikatoren

Der zweite Grundtyp von Klassifikatoren bildet aus den Trainingsmustern keine Prototypen, sondern versucht, die Entscheidungsfunktionen direkt zu approximieren. Die verschiedenen Funktionalklassifikatoren unterscheiden sich nach der für die Approximation verwendeten Funktionenklasse. Häufig verwendet werden Polynome, woraus sich der *Polynomklassifikator* ergibt, und Exponentialfunktionen, die typisch für viele *neuronale Netze* sind.

6.4.1 Polynomklassifikator

Der Polynomklassifikator, der ausführlich in (Schürmann 1977) behandelt wird, modelliert die einzelnen Funktionen d_k des Entscheidungsvektor \mathbf{d} durch Polynome der Elemente des Eingangsvektors i :

$$d_k = a_{0,k} + a_{1,k}i_1 + a_{2,k}i_2 + \dots + a_{N,k}i_N + a_{N+1,k}i_1^2 + a_{N+2,k}i_1i_2 + \dots \quad (6.8)$$

Wenn man alle Polynomterme $i_1^a i_2^b \dots i_N^c$ in einem Vektor \mathbf{x} zusammenfasst, kann man den vollständigen Ansatz für den Vektor der Entscheidungsfunktionen sehr kompakt als Vektorgleichung schreiben:

$$\mathbf{d} = \mathbf{Ax} . \quad (6.9)$$

Üblicherweise nimmt man jetzt noch gewisse Einschränkungen bezüglich der verwendeten Polynomterme vor. Ein einfacher, regelmäßiger Ansatz verwendet beispielsweise sämtliche Polynomterme bis zu einem Grad G . Die Koeffizientenmatrix A kann dann über einen Quadratmittelansatz optimiert werden, d. h. man fordert dass die Summe der quadratischen Abweichungen von den wahren Zielvektoren minimal werden soll:

$$\sum |o - Ax(i)|^2 = \min. \quad (6.10)$$

Die für ein Minimum notwendige Forderung, dass die partiellen Ableitungen dieser Fehlersumme nach allen Polynomkoeffizienten in der Matrix A verschwinden müssen, führt auf ein lineares Gleichungssystem für die Koeffizienten von A , das mit den üblichen Verfahren für lineare Gleichungssysteme gelöst werden kann, worauf wir hier nicht näher eingehen wollen.

Approximationseigenschaften: Nach dem Approximationssatz von Weierstraß, (Bronstein et. al. 2005), gilt, dass jede stetige Funktion mit einem Polynom beliebig genau approximiert werden kann, wenn der Grad des Polynoms hoch genug gewählt wird. Der Polynomklassifikator ist folglich ein universeller Approximator. Die Anzahl der Terme wächst jedoch bei vollständigen Ansätzen so schnell, dass eine Optimierung kaum noch möglich ist. Abhilfe für dieses Problem kann einmal durch die Selektion von Polynomtermen höheren Grades geschaffen werden, wobei die möglichen Selektionsregeln anwendungsabhängig sind. Außerdem lässt sich der Polynomklassifikator zum Funktionalklassifikator erweitern, (Pao 1989), indem man nicht nur Polynomterme, sondern beliebige Funktionen der Eingangsvektoren zulässt. Das stellt keine grundsätzliche Änderung dar, da sich diese Funktionen auch wieder durch Polynome schreiben lassen, kann aber die Anzahl der freien Parameter deutlich verringern.

6.4.2 Neuronale Netze vom Typ Multilayer-Perzeptron

Der Begriff der neuronalen Netze entstand aus dem Bestreben, bestimmte Verarbeitungsleistungen, speziell im Bereich der Mustererkennung, in Analogie zu biologischen Gehirnen durch Zusammenschaltung einer Vielzahl einfacher Verarbeitungseinheiten in einem komplexen Netzwerk zu erreichen. Dies hat lange Zeit zu heftigen Auseinandersetzungen zwischen den Vertretern „klassischer“ statistischer Verfahren und denen der Neuroinformatik geführt. Wenn man einmal von den eher biologisch-psychologisch orientierten Forschungszweigen absieht, die sich tatsächlich um eine Modellierung bestimmter Vorgänge im Nervensystem bemühen, sieht man neuronale Netze mittlerweile eher als eine neue Betrachtungsweise im Rahmen der statistischen Modellierung an, also als eine bestimmte Art, statistische Verfahren darzustellen und zu implementieren. Tatsächlich kann man die bisher dargestellten Klassifikatoren in Form neuronaler Netze implementieren, was bei den RCE-Netzen bereits anklang.

Die neue Sichtweise, die mit den neuronalen Netzen entstand, hat zu einer Reihe interessanter Ergebnisse geführt. Ein wesentlicher Unterschied zwischen den klassischen Verfahren und den neuronalen liegt darin, dass letztere im Allgemeinen mehr Eingangsvariablen verwenden und mit nichtlinearen Funktionenansätzen arbeiten.

Wir können und wollen hier keinen umfassenden Überblick über neuronale Netze im Allgemeinen geben. Stattdessen wollen wir kurz zeigen, wie der in technischen Anwendungen gängigste Typ neuronaler Netze, das Multilayer-Perzeptron, arbeitet.

Aufbau und Verarbeitungseinheiten: Oft werden die Elemente künstlicher neuronaler Netze einfach als *Neuronen* bezeichnet. Da sie sich aber von biologischen Neuronen in ihren Fähigkeiten doch stark unterscheiden, soll hier der Begriff *Verarbeitungseinheiten* oder einfach *Einheiten* benutzt werden, in Analogie zu der in der englischen Literatur häufigen Bezeichnung *unit*.

Der typische Aufbau eines Multilayer-Perzeptrons ist in Abbildung 6.5 gezeigt. Es besteht meist aus drei Schichten von Verarbeitungseinheiten, einer Eingangsschicht, einer so genannten verborgenen Schicht und einer Ausgabeschicht. Dazwischen befinden sich zwei Lagen von Verbindungen. Die Einheiten der Eingangsschicht sind hier mit i für *input* bezeichnet, die der verborgenen Schicht mit h für *hidden* und die der Ausgabeschicht mit o für *Output*. Es gibt viele Abwandlungen von dieser Struktur, beispielsweise kann es Direktverbindungen zwischen der Eingangsschicht und der Ausgabeschicht geben, so genannte *Shortcut-Verbindungen*. Es sind auch mehr als drei Schichten möglich. Der hier gezeigte Aufbau ist aber sehr häufig: dreischichtig² und streng hierarchisch, d. h. Verbindungen überspringen keine Schichten.

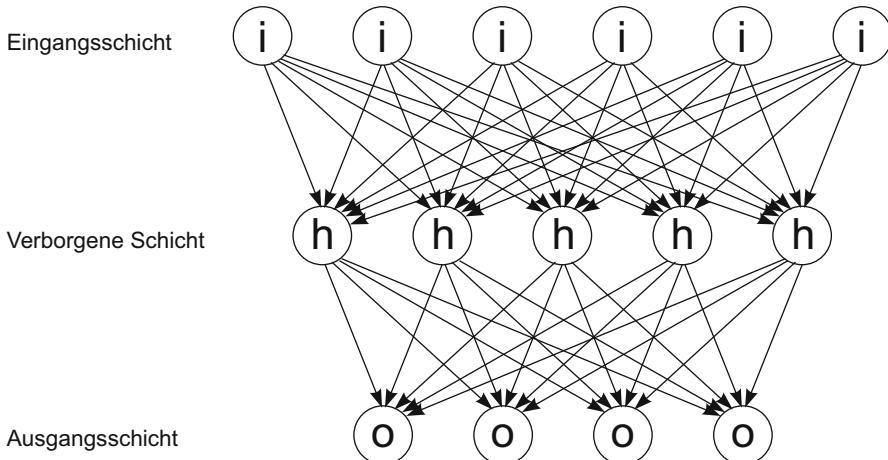


Abbildung 6.5: Typischer Aufbau eines neuronalen Netzes

Abbildung 6.6 zeigt das Blockschaltbild einer Verarbeitungseinheit. Jede Verarbeitungseinheit besitzt einen oder mehrere Eingänge, die mit Gewichtsfaktoren w_k

² Die Bezeichnungsweise in der Literatur ist uneinheitlich; manche Autoren sprechen hier von dreischichtigen Netzen, da drei Schichten von Verarbeitungseinheiten vorhanden sind, andere nur von zweischichtigen, da die Eingangsschicht nur Werte entgegennimmt und keine eigenen Berechnungen durchführt.

versehen sind. Aus der gewichteten Summe der Eingangssignale wird der Ausgangswert berechnet.

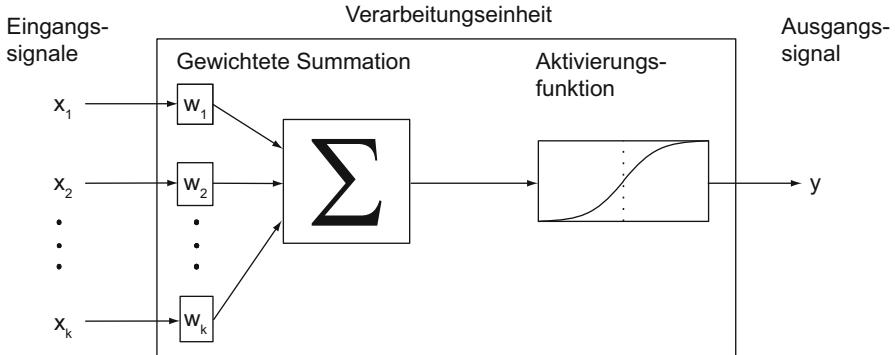


Abbildung 6.6: Blockschaltbild einer Verarbeitungseinheit in einem Multilayer-Perzeptron

Der Aktivierungszustand einer Einheit j ist demnach eine Funktion der gewichteten Summe aller Ausgänge o_k der Einheiten k , die über gewichtete Verbindungen w_{kj} ³ mit der Einheit j verbunden sind:

$$a_j = f_a \left(\sum_k w_{kj} o_k - \theta_j \right). \quad (6.11)$$

Als Aktivierungsfunktionen werden im Multilayer-Perzeptron praktisch immer sigmoidförmige Funktionen verwendet, wofür Abbildung 6.7 zwei typische Beispiele zeigt.

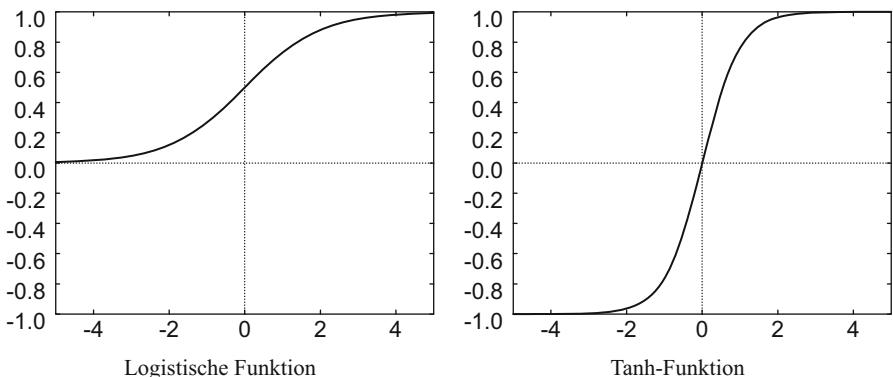


Abbildung 6.7: Sigmoid Aktivierungsfunktionen

³ Der Gebrauch der Indices ist in der Literatur uneinheitlich; hier wird stets der Index der Einheit, von der das Gewicht kommt, zuerst genannt, der, wohin das Gewicht geht, zuletzt.

Meistens verwendet man die *logistische Funktion*:

$$f_a(\text{net}_j) = \frac{1}{1 + e^{-(\text{net}_j - \theta)}}. \quad (6.12)$$

Von besonderem Vorteil ist, dass die logistische Funktion durch ihr asymptotisches Verhalten den Aktivierungszustand automatisch limitiert, unabhängig von der Größe der Eingangsaktivität. In manchen Fällen ist eine Bipolarität des Aktivierungszustandes vorteilhaft, so dass man die tanh-Funktion verwendet, die der Sigmoid-Funktion ähnlich sieht, jedoch Werte zwischen minus eins und eins annimmt statt zwischen null und eins.

Kontrollstrategie und Adaption: Weitere Kennzeichen eines neuronalen Netzes sind die *Kontrollstrategie* und die *Adoptionsregel*. Die Kontrollstrategie bestimmt, die Art und Weise, in der Aktivierungen sich durch das Netz fortpflanzen. Multilayer-Perzeptron-Netze arbeiten nach dem Prinzip der *Vorwärtsvermittlung*, meist mit dem englischen Begriff *feed-forward strategy* bezeichnet, bei der die Aktivierung schichtweise von der Eingabe- zur Ausgabeschicht durchgeführt wird und keine Rückkopplungen bestehen.

Die Adoptionsregel beschreibt, wie Informationen im Netz verankert werden. Bei Multilayer-Perzeptron-Netzen werden dazu die Gewichtsfaktoren zwischen den Verarbeitungseinheiten in einem überwachten Lernvorgang (*supervised learning*) verändert. Dabei werden in jedem Lernzyklus alle Muster des Trainingsdatensatzes vom Netz klassifiziert. Die Ergebnisse des Netzes werden mit den Vorgabewerten des Trainingsdatensatzes verglichen und die Gewichtsfaktoren so angepasst, dass nach einer Reihe von Zyklen der Trainingsdatensatz mit einer vorgegebenen Genauigkeit reproduziert werden kann. Man bezeichnet dies als überwachtes Lernen, da ein Soll-/Ist-Vergleich stattfindet, d. h. das momentane Ergebnis des Netzes wird kontrolliert.

Die meisten überwachten Lernverfahren arbeiten genauso wie die Optimierung der Koeffizientenmatrix eines Polynomklassifikators nach dem Prinzip, die Summe der quadratischen Fehler am Ausgang des Netzes zu minimieren. Das bekannteste Trainingsverfahren ist die *Fehler-Rückvermittlung*, besser bekannt als *Backpropagation-Training*, bei dem in jedem Lernschritt die Gewichte in Richtung abnehmenden Fehlers verändert werden. Das Verfahren ist ausführlicher in Abschnitt A.1 beschrieben.

Unbeschadet mancher Nachteile haben sich mit Backpropagation trainierte Multilayer-Perzeptron-Netze für eine große Klasse von Problemen als sehr leistungsfähige und robuste Klassifikatoren erwiesen. Es sei noch erwähnt, dass Exponentialfunktionen ebenso einen universellen Approximator ergeben wie Polynome, d. h. mit einem neuronalen Netz hinreichender Größe lässt sich jede beliebige Funktion beliebig genau darstellen.

6.5 Bemerkungen zum Einsatz neuronaler Netze

Dieser Abschnitt könnte auch *Regeln für den Einsatz statistischer Klassifikationsverfahren* überschrieben sein, denn bei allen statistischen Modellierungs- und Klassifikationsverfahren müssen gewisse Grundregeln beachtet werden. Einige für die Praxis besonders wichtige wollen wir in diesem Abschnitt anführen.

6.5.1 Zusammensetzung der Lernstichprobe

Die Lernstichprobe muss das Problem hinreichend genau und umfassend repräsentieren. Sie muss daher immer auch „Grenzmuster“ beinhalten, Muster, die gerade noch als gut, oder lesbar, einzustufen sind. Wichtig ist auch, dass alle Klassen in etwa gleichstark repräsentiert sein sollten. Eine unterrepräsentierte Klasse trägt weniger zu der Fehlerfunktion bei, die in der Trainingsphase optimiert wird, und wird daher auch schlechter gelernt. Bis zu einem gewissen Grade kann man dem durch mehrfache Vorlage von Mustern einer unterrepräsentierten Klasse begegnen, wobei diese ein wenig durch Rauschen verfälscht werden sollten. Man muss sich jedoch darüber klar sein, dass dadurch keine neue Information eingeführt wird. Echte Trainingsmuster sind immer einer künstlichen Vervielfältigung vorzuziehen.

6.5.2 Merkmalsskalierung

Bei allen statistischen Analyseverfahren ist es von großer Wichtigkeit, auf vergleichbare Skalen der einzelnen Merkmale zu achten, da sonst bestimmte Merkmale andere dominieren können. Weitere Schwierigkeiten, die durch fehlerhafte Skalierung hervorgerufen werden können, sind mangelnde Selektivität des Klassifikators und Probleme beim Training. Bei neuronalen Netzen kann man beispielsweise beobachten, dass ab einem Betrag des Eingangswertes von etwa fünf die logistische Aktivierungsfunktion nahezu den Wert eins erreicht und nicht mehr verlässt. Dieser Sättigungseffekt hat folgende Wirkungen:

- Das Netz kann zwischen verschiedenen hohen Eingangswerten kaum unterscheiden, es verliert seine Selektivität.
- Das Training wird enorm langsam, da es sich auf die Ableitung der Aktivierungsfunktion stützt, die in diesen Bereichen nahezu Null ist.

Häufig wird eine Normierung der Eingangswerte auf ein Intervall $[-0,5; 0,5]$ vorgenommen. Das hat den Vorteil, dass man sich im steilsten Bereich der Aktivierungsfunktion befindet. Für Eingangssignale, deren Wertebereich vorher genau bekannt ist, ist dies sehr einfach und führt auch zu guten Ergebnissen. Ein Beispiel hierfür sind die einzelnen Bildpunkte eines Rasterbildes.

Bei anderen Merkmalen, wie beispielsweise Objektdimensionen oder Flächen, können die Grenzen nicht vorher bestimmt werden. Hier können auch gelegentlich sehr große Werte auftreten. Wenn man dann eine solche Skalierung auf ein festes Maximum und Minimum durchführt, läuft man Gefahr, kleinere Werte so stark zusammenzupressen, dass sie kaum noch unterscheidbar sind. In diesem Fall kann man sich die Erfahrungen aus der multivariaten Statistik zunutze machen und eine Standardisierung über den gesamten Trainingsdatensatz durchführen. Das bedeutet, dass Mittelwert und Standardabweichung des jeweiligen Merkmals für alle Muster des Trainingsdatensatzes berechnet werden. Danach werden die Werte so skaliert, dass der Mittelwert null ist und die Standardabweichung eins. Damit liegt die Mehrzahl der Werte dicht um null verteilt, aber die Variation der Werte ist immer noch erkennbar.

6.5.3 Rückweisung

Multilayer-Perzeptron-Netze und Polynomklassifikatoren leiden unter genau dem gleichen Problem wie der Nächster-Nachbar-Klassifikator: Für Eingangsmuster, die außerhalb ihres „Erfahrungsbereiches“ liegen, d. h. weitab von allen Mustern des Trainingsdatensatzes, können sie willkürliche Ergebnisse liefern. Das hängt damit zusammen, dass diese Klassifikatoren Trennflächen in den Merkmalsraum legen. Man kann sie daher auch als *separierende* Klassifikatoren bezeichnen, im Gegensatz zu den *integrierenden* Multireferenzenklassifikatoren, die ähnliche Muster unter einem Prototypen zusammenfassen. Diese Trennebenen kommen zwar auf völlig andere Weise zustande als die des Nächster-Nachbar-Klassifikators, der Effekt ist aber der gleiche: Alles was sich auf einer Seite einer Trennebene befindet, wird einer bestimmten Klasse zugewiesen. Da die Trennflächen sich bis ins Unendliche erstrecken, werden auch unsinnige Muster ohne weiteres klassifiziert.

Dies ist übrigens ein generelles Problem statistischer Klassifikationsverfahren. Die approximierten Klassenwahrscheinlichkeiten sind normalerweise gar nicht im gesamten Merkmalsraum definiert, so dass es in großen Bereichen gar keine Trainingsmuster geben kann. Die Schätzfunktionen sind dagegen prinzipiell im gesamten Merkmalsraum definiert.

Es gibt umfangreiche Forschungen zu dem Problem, eine Rückweisungsregel für Perzeptron-Netze zu entwerfen. Sie reichen von einfachen Plausibilitätsprüfungen bis hin zu einer statistischen Modellierung des gesamten Raums der Trainingsmuster. Bei einem gut trainierten Klassifikator ist es erfahrungsgemäß eine brauchbare Rückweisungsregel, den Kontrast zwischen der wahrscheinlichsten und der zweitwahrscheinlichsten Klasse zu berechnen.

Die Klassenzuordnung ergibt sich daraus, welche Ausgabeeinheit die höchste Aktivierung aufweist. Wenn o_{max} dieser höchste Aktivierungswert ist, o_{max2} der zweithöchste, der in der Ausgabeschicht vorkam, so berechnet sich diese *Klassifikationsgüte* zu:

$$g = \frac{o_{max} - o_{max2}}{o_{max} + o_{max2}}. \quad (6.13)$$

Man erkennt, dass dieser Wert sich immer stärker der eins nähert, je weiter die beiden Aktivierungen auseinander liegen. Er wird null, wenn die Aktivierungen gleich sind, sich das Netz also völlig unsicher ist, welche Klasse die richtige ist. Man beachte aber, dass dieses Gütemaß keine sinnvolle Aussage mehr ergibt, wenn der Klassifikator mit Dingen konfrontiert wird, die mit der Aufgabe, auf die er trainiert wurde, nichts zu tun haben. Das gilt vor allem, wenn die Eingangsmuster Klassen angehören, die dem Klassifikator unbekannt sind, für die er also überhaupt keine Entscheidungsfunktionen besitzt. Er muss sich prinzipbedingt für eine der bekannten Klassen entscheiden. In bestimmten Fällen wird er dabei auch keine Unsicherheit zeigen, z. B. bei einem Zwei-Klassen-Klassifikator, der den gesamten Merkmalsraum mithilfe einer einzigen Entscheidungsfunktion in zwei klar getrennte Bereiche eingeteilt hat. In typischen technischen Anwendungen, bei denen ein Klassifikator mit einer größeren Zahl vorher bekannter Klassen trainiert wurde, kommen solche unbemerkt Fehlentscheidungen erfahrungsgemäß nicht vor.

Für weitere Ausführungen zu diesem Thema verweisen wir auf die umfangreiche Literatur zum Gebiet der neuronalen Netzen, z. B. (Zell 1994), (Bishop 1995), (Smith 1993), (Brause 1991).

6.5.4 Abgrenzung zu anderen Klassifikatoren

Zum Schluss dieses Kapitels wollen wir noch ein paar Worte zu einem Vorwurf sagen, der neuronalen Netzen häufig gemacht wird. Man hört oder liest öfter, die Ergebnisse neuronaler Netze seien unberechenbar, zufällig oder subjektiv gefärbt. Dies wird besonders gern im Gegensatz zu klassischen, einfachen Verfahren, wie etwa der Korrelation, behauptet und ist natürlich für jemanden, der ein System zur Qualitätssicherung zu entwerfen hat, von einiger Bedeutung.

Der Vorwurf der Unberechenbarkeit ist für den Klassifikationsvorgang selbst offensichtlich unbegründet, schließlich wird das Ergebnis ja berechnet, und es wird jedes mal auf exakt dieselbe Weise berechnet. Daher kommt auch bei gleichen Eingangssignalen immer genau das Gleiche heraus. Mehr kann man von keinem Verfahren behaupten. Hier, wie auch in anderen Bereichen, unterscheiden sich neuronale Netze überhaupt nicht von „klassischen“ Verfahren, so dass es einfach sinnlos ist, einen Gegensatz konstruieren zu wollen. Zutreffend ist allerdings, dass es keine Stelle gibt, an der das Wissen des Netzes über ein bestimmtes Eingangsmuster gespeichert ist, im Gegensatz zu einem Multireferenzenklassifikator, bei dem sich feststellen lässt, welcher Prototyp zu einer bestimmten Aussage geführt hat. Aus diesem Grund erscheinen Multireferenzenklassifikatoren leichter durchschaubar.

Der Vorwurf der Zufälligkeit kann sich nach den obigen Ausführungen also nur auf das Training beziehen. Neuronale Netze werden in der Regel mit Zufallszah-

len initialisiert, daher muss beim Training mit derselben Stichprobe nicht exakt dieselbe Klassifikator entstehen. Bei austrainierten Klassifikatoren, bei denen sich der Fehlerwert kaum noch ändert und hinreichend niedrig liegt, spielt das jedoch für das Verhalten kaum eine Rolle.

Eng hiermit verknüpft ist der Vorwurf der Subjektivität. Er bezieht sich darauf, dass es sich um ein lernendes Verfahren handelt, das aus einer Stichprobe lernt, die von Menschen zusammengestellt und klassifiziert wurde. Auch hierin unterscheiden sie sich allerdings nicht von anderen statistischen Verfahren, die bei verschiedenen Lernstichproben ebenfalls zu verschiedenen Ergebnissen kommen werden. Das gilt selbst für ein so einfaches Verfahren wie die Korrelation, die sich ja genau wie der Nächster-Nachbar-Klassifikator auf einen Vergleich des aktuellen Musters mit einer Reihe gespeicherter Muster stützt. Diese Muster müssen aber auch zu irgendeinem Zeitpunkt ausgewählt und mit Klassenzuordnungen versehen worden sein. Auch hier besteht also kein Unterschied.

Einer der Hauptgründe für diese Vorwürfe dürfte wohl in der grundsätzlich unterschiedlichen Vorgehensweise der beiden Klassifikatorgrundtypen liegen. Wir erwähnten im Zusammenhang mit den Multireferenzenklassifikatoren, dass ihr Hauptnachteil in der fehlenden Bewertung der Merkmale liegt. Dadurch sind sie sehr störempfindlich. Neuronale Netze zeigen ein wesentlich robusteres Verhalten, allerdings ist die Art und Weise, auf die sie während des Trainings lernen, die Eingangsdaten zu bewerten, oftmals schwer nachzuvollziehen.

6.6 Zusammenfassung

Klassifikation: Unter Klassifikation im technischen Bereich verstehen wir die Zuordnung eines Objekts zu einer von mehreren Klassen aufgrund eines Satzes von Merkmalen, die dieses Objekt beschreiben.

Klassifikation als Abbildung: Man kann Klassifikation als Abbildung aus einem Merkmalsraum in einen Klassenraum auffassen. Bei typischen Klassifikationsanwendungen, wie der Schrifterkennung, ist die Abbildungsvorschrift nicht formal bekannt.

Muster: Im Zusammenhang mit der Klassifikation ist ein Muster ein durch einen Merkmalssatz beschriebenes Objekt. Gleichbedeutung mit Merkmalsvektor.

Maschinelles Lernen: Hierunter versteht man die Ableitung einer Klassifikationsfunktion aus Beispielen einer (z. B. vom Menschen) vorgegebenen Zuordnung zwischen Mustern und Klassen.

Merksmal: Einzelner Wert, der einen Aspekt eines Objekts beschreibt, z. B. eine Größenangabe oder die Helligkeit eines Punkts in einem Rasterbild.

Merkmalsvektor: Dieser Vektor enthält die ein Objekt beschreibenden Merkmalswerte.

Zielvektor: Der Zielvektor beim maschinellen Lernen enthält die korrekte, vorgegebenen Klassenzuordnung.

Ergebnisvektor: Der Ergebnisvektor enthält die vom Klassifikator ermittelten Wahrscheinlichkeiten für die Klassenzugehörigkeit eines Musters.

Lernstichprobe/Trainingsdatensatz: Sammlung von Mustern mit zugehörigen Klassenzuordnungen, die zur Anpassung eines Klassifikators auf das zu lernende Problem verwendet wird.

Überwachtes Lernen: Hierunter werden alle Lernverfahren zusammengefasst, bei denen eine Überprüfung des Klassifikationsergebnisses nach einem Lernschritt stattfindet und dann eine Korrektur der Parameter des Klassifikators erfolgt. Gegensatz dazu ist das *unüberwachte Lernen*, bei dem der Klassifikator aufgrund von Ähnlichkeiten der Trainingsmuster die Klassenstruktur selbst ermittelt.

Randproblem: Das Randproblem entsteht dadurch, dass die Lernstichprobe nicht den gesamten Merkmalsraum abdecken kann. Wenn ein Klassifikator mit einem Muster konfrontiert ist, das weit außerhalb des durch die Lernstichprobe beschriebenen Bereichs liegt, kann er unter Umständen mit willkürlichen Ergebnissen antworten, da er keine Möglichkeit hat, ein solches Muster vernünftig abzugrenzen.

Bayes-Klassifikator: Dieser Klassifikator hat die theoretisch geringstmögliche Fehlerrate. Da er vollständige Kenntnis der statistischen Eigenschaften des Problems voraussetzt, kann er in der Praxis nicht konstruiert werden.

Multireferenzenklassifikatoren: Dieser Klassifikatortyp bestimmt die Klassenzugehörigkeit eines Musters nach seiner Ähnlichkeit zu gespeicherten Prototypen. Sie sind sehr einfach zu konstruieren und zu trainieren. Ihre Hauptnachteile sind die hohe Anzahl von Prototypen, die für komplizierte Klassengrenzen erforderlich sind, und die dadurch sehr hohe Rechenzeit bei der Klassifikation.

Nächster-Nachbar-Klassifikator: Dieser einfachste Typ des Multireferenzenklassifikators bestimmt die Ähnlichkeit eines Musters zu den Prototypen mittels des euklidischen Abstandes.

RCE-Netze: Diese Abwandlung des Nächster-Nachbar-Klassifikators arbeitet mit einer Rückweisungsschwelle für jeden Prototyp. Muster, die einen größeren Abstand vom nächsten Prototypen aufweisen, als dessen Rückweisungsschwelle angibt, werden nicht klassifiziert. Der Klassenbereich wird durch Kreise (bzw. Kugeln und Hyperkugeln für höherdimensionale Probleme) angenähert. Damit kann die Zahl der Prototypen reduziert und das Randproblem der Nächster-Nachbar-Klassifikatoren vermieden werden.

Vektorquantisierung: Hierunter versteht man die Repräsentation eines Raums von Merkmalsvektoren durch eine gewisse Anzahl von Prototypvektoren. Kann zur Reduzierung der Prototypenzahl von Nächster-Nachbar-Klassifikatoren oder als eigenständiges Klassifikationsverfahren benutzt werden.

Template Matching: Diese Form des Nächster-Nachbar-Klassifikators verwendet statt des euklidischen Abstandes den Korrelationskoeffizienten zur Bewertung der Ähnlichkeit von Mustern und Prototypen.

Funktionalklassifikatoren: Diese Klassifikatoren modellieren die Entscheidungsfunktionen für die Klassenzugehörigkeit direkt durch einen Funktionenansatz. Sie unterscheiden sich nach der Wahl des Funktionensystems. Vorteile liegen in der hohen Verarbeitungsgeschwindigkeit in der Klassifikationsphase und der Fähigkeit, sehr komplexe Probleme abilden zu können. Nachteile sind Randprobleme und teilweise sehr aufwändige Trainingsverfahren.

Polynomklassifikator: Dieser Typ der Funktionalklassifikatoren verwendet Polynome als Ansatz für die Entscheidungsfunktionen. Polynome hinreichenden Grades sind ein universelles Approximationssystem, d. h. sie können jede Funktion mit beliebiger Genauigkeit darstellen. Bei steigendem Polynomgrad wird die Bestimmung der Koeffizienten des Klassifikators allerdings sehr schwierig. Die Koeffizienten werden nach dem Kriterium des minimalen quadratischen Abstandes zwischen den Ergebnisvektoren des Klassifikators und den Zielvektoren des Trainingsdatensatzes bestimmt.

Multilayer-Perzeptron-Netz: Dieses weit verbreitete neuronale Netz ist ein Funktionalklassifikator auf der Basis von Exponentialfunktionen, die wie Polynome ein universelles Approximationssystem darstellen. Sie werden meist mit dem Backpropagation-Verfahren trainiert.

Backpropagation: Die Fehlerrückvermittlung ist das am meisten verbreitete Trainingsverfahren für neuronale Netze. Es beruht auf der schrittweisen Veränderung der internen Gewichtsfaktoren des Netzes in Richtung abnehmenden Fehlers. Das Fehlerkriterium ist der mittlere quadratische Abstand zwischen den Ergebnisvektoren des Netzes und den Zielvektoren des Trainingsdatensatzes.

Merkmalskalierung: Eine Voraussetzung für den erfolgreichen Einsatz jedes statistischen Modellierungs- oder Klassifikationsverfahrens ist eine vergleichbare Skalierung aller Merkmale in einem Bereich, in dem die Klassifikationsfunktionen ausreichende Trennschärfe aufweisen. Wie bei vielen Verfahren aus der multivariaten Statistik ist auch für neuronale Netze eine Standardisierung auf Mittelwert null und Standardabweichung eins vorteilhaft.

Rückweisung: Zur Vermeidung des Randproblems bei neuronalen Netzen und anderen Klassifikatoren kann man Rückweisungsregeln aufstellen. Das Auftreten unerwarteter Ergebnisse bei Eingangsmustern, die außerhalb der spezifizierten und trainierten Problemdefinition liegen, ist jedoch dadurch nicht völlig auszuschließen.

7. Vermessung

Vermessung oder Form- und Maßprüfung gehört zu den anspruchsvollsten Aufgaben der industriellen Bildverarbeitung, sowohl in algorithmischer Hinsicht als auch bezüglich der Systemtechnik, des Anlagenbaus. Es ist durchaus möglich, Messgenauigkeiten bis hinunter zu wenigen Lichtwellenlängen zu erreichen, aber dafür ist ein außerordentlich hoher Aufwand erforderlich. Wie in jeder technischen Disziplin ist es auch hier unmöglich, präzise Ergebnisse ohne entsprechende Sorgfalt zu erreichen. Sorgfalt vor allem bei der Peripherie, der Auswahl der Komponenten, dem mechanischen Aufbau, der Beleuchtung, der Bildaufnahme. Qualität, die in der Sensorkette verloren geht, ist unwiederbringlich verloren. Aus diesem Grund schließt sich auch das Querschnittskapitel über Beleuchtungs- und Aufnahmetechnik an dieses Kapitel an.

Maßprüfungen kommen in vielerlei Form vor, von der relativ groben Ermittlung von Teiledimensionen bis hin zu hochgenauen Vermessungen im Mikrometerbereich. Vermessungen können selbstständige Prüfaufgaben sein, wenn z. B. die Einhaltung von metrisch vorgegebenen Toleranzen zu prüfen ist, sie können aber auch als Hilfsmittel zur Montagekontrolle dienen. Dort spielt dann unter Umständen der exakte Wert eines Maßes eine geringere Rolle als Verhältnisse von Maßen zueinander, anhand derer sich die korrekte Lage einer Komponente in Bezug auf andere Bauteile feststellen lässt.

Entsprechend vielfältig sind die Verfahren und Systeme, die für die optische Vermessung entwickelt wurden. Sie reichen von der Feststellung relativer Maße auf Pixelebene über subpixelgenaue Vermessung im Rahmen von Standardsystemen bis hin zu hoch spezialisierten Koordinatenmeßsystemen, komplett mit Mechanik, Optik und Beleuchtung. Derartige Systeme erreichen Genauigkeiten bis zu einem Mikrometer – bei integralen Maßen, wie z. B. Schwerpunkten von Objekten, auch genauer –, eignen sich aber aufgrund ihres Aufbaus nur für Inspektionsaufgaben innerhalb automatisierter Fertigungsanlagen in kontrollierten Umgebungen (wie z. B. Reinraumbereichen) oder für den Laboreinsatz.

Wir wollen in diesem Kapitel versuchen, anhand einiger praktischer Beispiele ein Gefühl für die Möglichkeiten und Voraussetzungen dieser Technologie zu geben. Zuvor aber müssen einige wichtige Begriffe und Techniken eingeführt werden.

7.1 Vermessungsaufgaben

Vermessung kann sehr verschiedenen Zwecken dienen. Entsprechend unterschiedlich sind die Anforderungen an Sensorik und Algorithmik. Typische Anwendungen mit sehr unterschiedlichen Anforderungen sind:

Montagekontrolle: Hier werden Methoden aus der Vermessung zur Überprüfung der Anwesenheit von Bauteilen oder anderer Charakteristika (wie z. B. Bohrlöchern) eingesetzt. Vermessung wird somit als Hilfsmittel zur Anwesenheitskontrolle eingesetzt. Derartige Aufgaben erfordern häufig keine besonders hohe Genauigkeit, so dass man sich oft auf einfache Pixelvermessung beschränken kann.

Formkontrolle: In diesem Bereich geht es um die Feststellung von Abweichungen gegenüber einer Idealform, also beispielsweise um die Prüfung, ob eine Werkstückkante hinreichend glatt ist, alle Kontakte eines ICs exakt auf einer Geraden liegen oder eine Gruppe von Bohrungen nicht nur vorhanden ist, sondern auch genau in einem Kreis angeordnet ist. Oft wird bei derartigen Anwendungen zwar eine hohe relative Genauigkeit verlangt, jedoch keine so hohe absolute Genauigkeit. Das bedeutet, die Messergebnisse müssen präzise wiederholbar sein und innerhalb enger Toleranzen liegen, eine hochgenaue Kalibrierung zur Umrechnung von Bildpunkten in metrische Maße ist jedoch nicht erforderlich.

Maßprüfung: Die anspruchsvollsten Aufgaben liegen im Bereich der eigentlichen Maßprüfung, bei der bestimmte Maße mit hoher absoluter Genauigkeit zu ermitteln sind. Das stellt nicht nur hohe Anforderungen an die Bestimmung der Maße selbst, die hier fast immer den Einsatz von Subpixelverfahren erforderlich macht, sondern auch an die Sensorik, die absolut stabil wiederholbare Bildverhältnisse gewährleisten muss. Nur so ist eine Kalibrierung möglich, die eine exakte Umrechnung von Bildmaßen in reale Maße erlaubt.

In den folgenden Abschnitten werden wir diese Aufgabentypen an Beispielen vorstellen und dabei jeweils die benötigten Verfahren und Algorithmen einführen. So werden wir der Reihe nach die einfache Pixelvermessung, die Vermessung mit Bezug auf Idealformen und die subpixelgenaue Vermessung besprechen. Kalibrierung und Sensorik werden in einem eigenen Abschnitt am Ende dieses Kapitels behandelt.

7.2 Einfache Vermessung

Wir beginnen diese Einführung in das Gebiet der Vermessung mit einem sehr einfachen Beispiel, um verschiedene Aspekte einer Messaufgabe vorstellen zu können, ohne dabei auf besondere Schwierigkeiten und Eigenschaften des Prüfteils oder der Aufgabe eingehen zu müssen. Im Interesse einer knappen und geschlossenen Darstellung werden wir dabei einige Vereinfachungen in Kauf nehmen, auf

die wir dann bei der eingehenderen Behandlung einzelner Maße und Messverfahren noch zurückkommen.

Das Prüfteil für dieses Einführungsbeispiel sehen Sie in Abbildung 7.1. Es handelt sich um das gestanzte Seitenblech eines Autoradios, an dem wir im Folgenden verschiedene Maße feststellen werden.

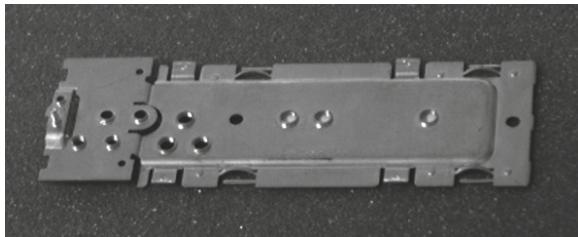


Abbildung 7.1: Seitenblech eines Autoradios

Abbildung 7.2 zeigt den vorderen Bereich des Bleches im Durchlicht, d. h. vor einer Leuchtplatte. Hintergrundbeleuchtungen dieser Art sind für Vermessungsaufgaben typisch wegen des hervorragenden Kontrastes, den sie erzeugen, allerdings gibt es da auch einige interessante Fallen, auf die wir im Abschnitt 7.7 eingehen werden.

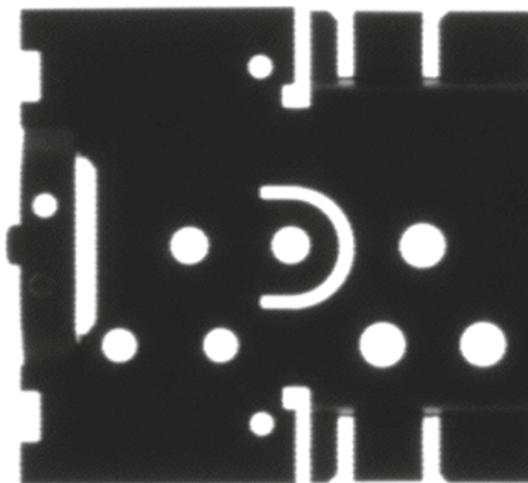


Abbildung 7.2: Ausschnitt aus gestanztem Blechteil im Durchlicht

7.2.1 Schwerpunktsabstände

Es gibt viele Möglichkeiten, in der industriellen Bildverarbeitung Abstände zu messen, sowohl von der Definition als auch von der Methodik her. Abstände können zwischen den Schwerpunkten von Objekten gemessen werden, zwischen ihren Konturen oder Kanten, senkrecht auf Kanten usw.

Eine der einfachsten Aufgaben ist die Feststellung von Schwerpunktsabständen. Sobald die relevanten Objekte gefunden sind, kann durch eine Mittelung über die Koordinaten aller Objektpunkte der Schwerpunkt jedes Objekts bestimmt werden (mehr hierüber im entsprechenden Abschnitt des Kapitels über Objektmerkmale, 10.1.3). Wenn die Schwerpunkte (x_1, y_1) und (x_2, y_2) der beiden fraglichen Objekte bekannt sind, kann der Pixelabstand einfach nach Pythagoras berechnet werden:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} . \quad (7.1)$$

Damit ist der Abstand in Bildpunkten bekannt. Wenn man etwas über das reale Objekt herausfinden will, muss man eine Kalibrierung durchführen, anhand derer sich dann die Bildpunkte in metrische Maße umrechnen lassen. Diesem keineswegs trivialen Aspekt der Vermessung ist mit 7.7 ein eigener Abschnitt gewidmet, daher beschränken wir uns hier auf das Nötigste.

Aufgrund von perspektivischen Effekten ist der Abbildungsmaßstab nicht überall im Bild genau gleich, weshalb man für hochgenaue Vermessung besondere Maßnahmen ergreifen muss, aber für den Augenblick nehmen wir einmal einen einheitlichen Abbildungsmaßstab an. Wir setzen weiter voraus, dass die Bildpunkte wie heute üblich quadratisch sind, d. h. also dass eine horizontale Strecke von einem Millimeter Länge in ebenso vielen Bildpunkten abgebildet wird wie eine vertikale Strecke derselben Länge. Unter dieser Voraussetzung kann der metrische Abstand der Objektschwerpunkte einfach durch Multiplikation mit einem Kalibrierfaktor c aus dem Pixelabstand der Gleichung (7.1) errechnet werden:

$$d_m = cd = c\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} . \quad (7.2)$$

Es erscheint visuell selbstverständlich, dass Bildpunkte quadratisch sind, aber das liegt daran, dass sie auf Bildschirmen so dargestellt werden. In den Anfängen der industriellen Bildverarbeitung waren nichtquadratische Pixel tatsächlich üblich. Wenn wir also diese Voraussetzung fallenlassen, wird die Umrechnung schon etwas komplizierter. Man benötigt dann nämlich unterschiedliche Kalibrierfaktoren in x- und y-Richtung, die entsprechend einzeln anzuwenden sind:

$$d_m = \sqrt{(c_x(x_1 - x_2))^2 + (c_y(y_1 - y_2))^2} . \quad (7.3)$$

Wir werden in allen folgenden Beispielen im Interesse einer übersichtlicheren Darstellung auf eine Beschreibung der Kalibrierung verzichten. In Abschnitt 7.7 werden wir dann einige Bemerkungen zur Kalibrierung machen. Außerdem werden wir stets von quadratischen Pixeln ausgehen. Ein Grund dafür ist die Vereinfachung der Algorithmen, und dies betrifft nicht nur die leichtere Umrechnung eines fertigen Maßes in metrische Werte, sondern auch die Ermittlung bestimmter Maße, wie z. B. von Konturabständen, die bei nicht-quadratischen Pixeln, wenn überhaupt, nur mit großem Aufwand möglich ist. Der bei weitem wichtigere Grund ist aber, dass man beim heutigen Stand der Technik nichtquadratische Pixel als sensorischen Defekt betrachten muss und nicht einzusehen ist, warum sensorische Defekte durch algorithmische Klimmzüge ausgeglichen werden sollten. Von allen Anwendungsbereichen der industriellen Bildverarbeitung stellt die Vermessung sicherlich die höchsten Anforderungen an die Qualität der Sensorik, von der Beleuchtung über die Kameratechnik bis hin zur Übertragung zum Bildverarbeitungsrechner, und man kann einfach keine genauen Ergebnisse erwarten, wenn in der Sensorkette bereits Fehler auftreten.

Kehren wir nach diesem Exkurs in das Gebiet der Kalibrierung zu den Schwerpunktsabständen zurück. Zu vermessen ist hier der Abstand der kleinen Bohrungen oben und unten am Stanzblech. Es handelt sich dabei um Schraublöcher, die auf die entsprechenden Gewindelöcher im Gehäuse passen müssen. In Abbildung 7.3 ist dieses Maß eingezeichnet.

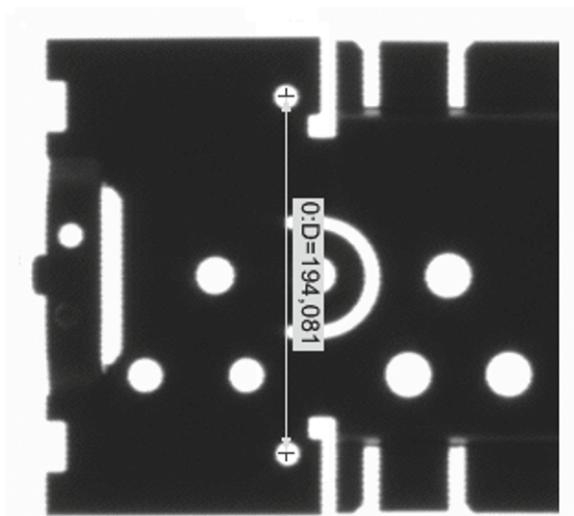


Abbildung 7.3: Schwerpunktsabstand zweier Schraublöcher

Wenn Sie das in Tabelle 7.1 beschriebene Beispiel ausprobieren, werden Sie feststellen, dass für den Abstand der Bohrungen ein Wert mit Nachkommastellen herauskommt. Dasselbe Phänomen werden Sie feststellen, wenn Sie zusätzlich zum

Abstand der Bohrungen auch die Koordinaten der Schwerpunkte berechnen lassen. Auch diese Messungen erzeugen Werte mit Nachkommastellen. Es stellt sich die Frage, wie eine Messung auf einem ortsdiskreten Pixelbild einen Dezimalwert erzeugen kann?

Die Antwort auf diese Frage liegt in der Art und Weise begründet, wie der Schwerpunkt eines Objekts ermittelt wird. Es ist völlig richtig, dass die einzelnen Punkte des Objekts, auch alle Punkte seiner Kontur, exakte, diskrete Pixelkoordinaten besitzen. Der Schwerpunkt eines Objekts wird aber durch Mittelung über alle diese Koordinaten berechnet, wodurch er genauer ist als jeder Objektpunkt einzeln. Betrachten wir als Beispiel eine Linie, die im 15° -Winkel durch das Bild läuft, wie es Abbildung 7.4 zeigt. Anhand des eingezeichneten Punkteraster kann man erkennen, dass die Linie durch folgende Punkte verläuft¹:

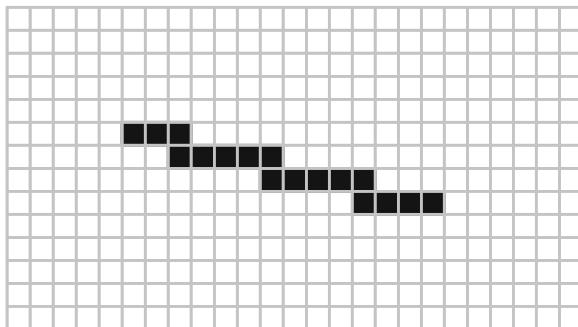


Abbildung 7.4: Diskretisierte Linie

(5; 5)	(6; 5)	(7; 5)	(7; 6)	(8; 6)	(9; 6)	(10; 6)	(11; 6)
(11; 7)							
(12; 7)	(13; 7)	(14; 7)	(15; 7)	(15; 8)	(16; 8)	(17; 8)	(18; 8)

Der zentrale Punkt dieser diskreten, digitalen Linie ist der in der Tabelle hervorgehobene Punkt (11, 7). Der wahre Mittelpunkt der diskreten Linie liegt jedoch bei (11,5; 6,5) – wenn man annimmt, dass die Endpunkte exakt sind – und tatsächlich kommt bei einer Schwerpunktsberechnung durch Mittelung über alle Koordinaten der Wert (11,4; 6,59) heraus, der bis auf 0,1 Bildpunkte am wahren Schwerpunkt liegt.

Die Mittelung über das gesamte Objekt hat noch einen weiteren angenehmen Effekt. Schwerpunktsmaße sind sehr stabil gegenüber Beleuchtungsveränderungen und eignen sich daher gut als Kalibriermaße. Abbildung 7.5 zeigt den Effekt verschiedener Belichtungen. Die Radien der beiden Bohrungen variieren je nach Belichtung zwischen 5,5 und 7,5 Bildpunkten, während die Schwerpunktsabstände in einem Toleranzbereich von weniger als 0,2 Pixel bleiben.

¹ Nach Programmierergewohnheit ab (0, 0) und von links oben nach rechts unten gezählt.



Abbildung 7.5: Veränderung der scheinbaren Öffnungs durchmesser durch unterschiedliche Beleuchtung

Beispielbeschreibung 7.1. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 7.2.1 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Vermessung
Beleuchtung	Diffuse Hintergrundbeleuchtung
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, gesamtes Bild 3. <i>Binärschwelle erzeugen</i> 4. <i>Arbeitsbereiche binär erzeugen</i>, gesamtes Blech 5. <i>Arbeitsbereiche binär erzeugen</i>, alle Öffnungen innerhalb des Bleches 6. <i>Maße erzeugen</i>, Abstand der beiden Bohrungen

7.2.2 Konturabstände

Oftmals sind nicht nur die Abstände zwischen den Schwerpunkten von Objekten interessant, sondern auch die zwischen ihren Außenkonturen. Dabei stellt sich allerdings die Frage, wie dieser Abstand überhaupt zu definieren ist. Letztlich sind Abstände nur zwischen Punkten zu messen, und je nachdem welche Punkte der beteiligten Konturen man betrachtet, kommt man zu ganz unterschiedlichen Abständen. Man muss also definieren, welche Punkte der Konturen benutzt werden sollen.

Eine naheliegende Spezifikation ist die Suche nach dem minimalen oder maximalen Abstand. Interessante Anwendungen für den minimalen Abstand zwischen Konturen sind z. B. die Breite der Isolation zwischen Leiterbahnen auf einer Platine oder der Kerndurchmesser eines Gewindes, gemessen als der minimale Abstand zwischen den Randkonturen des Gewindes, wie im Einführungsbeispiel in Abschnitt 1.6. Der maximale Abstand zweier Konturen kann entsprechend zur Bestimmung des Außendurchmessers eines Gewindes eingesetzt werden. Auf un-

serem Stanzblech ist z. B. die Frage interessant, wie dicht der senkrechte Schlitz links und der Kreis an seiner rechten unteren Ecke beieinander liegen, denn wenn der Abstand zwischen zwei Öffnungen zu gering wird, kann das eine Instabilität zur Folge haben. Abbildung 7.6 zeigt vergrößert die resultierende Maßlinie.

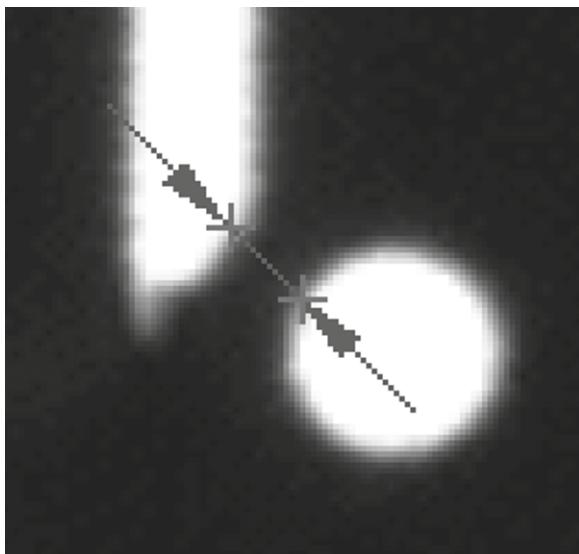


Abbildung 7.6: Minimaler Abstand zwischen zwei Konturpunkten

Die Bestimmung eines solchen Konturabstandes ist erheblich aufwändiger als die Bestimmung eines Schwerpunktsabstandes: Zwischen jedem Punkt der einen Kontur und jedem Punkt der anderen Kontur muss der Abstand berechnet, dann das Minimum oder Maximum gesucht werden. Die Berechnung des in Abbildung 7.6 dargestellten Maßes dauert daher ziemlich genau fünfmal so lange wie die des Schwerpunktsabstandes in Abbildung 7.3. Vereinfachungen sind praktisch nur möglich, wenn man Vorwissen über die Form der Objekte benutzt. Wenn man voraussetzen kann, dass die Objekte immer konvex² sind, braucht man für die Bestimmung des maximalen Abstandes beispielsweise nur den Teil der Kontur heranzuziehen, der auf der dem anderen Objekt abgewandten Seite liegt, wenn die Objekte aber von komplizierter Form sind und sich gegenseitig umschließen können, ist eine solche Vereinfachung nicht möglich, wie Abbildung 7.7 zeigt.

Die hellen Linien und Kreuze in Abbildung 7.7 bezeichnen die am weitesten auseinander liegenden Punkte der jeweiligen Konturen und die zugehörige Maßlinie. Man sieht, dass bei den beiden konvexen Objekten links die Punkte maximalen Abstandes jeweils auf der Seite des Objekts liegen, die bezogen auf die Objektschwerpunkte dem anderen Objekt abgewandt ist. Bei der mit null

² Ein Objekt wird als konvex bezeichnet, wenn keine zwei Punkte auf seiner Kontur existieren, deren Verbindungsline auch nur teilweise außerhalb der Objektkontur verläuft.

bezeichneten Maßlinie rechts ist dies nicht der Fall, denn dort liegt der Punkt auf der Kontur des Kreises zwischen den Schwerpunkten der Objekte.

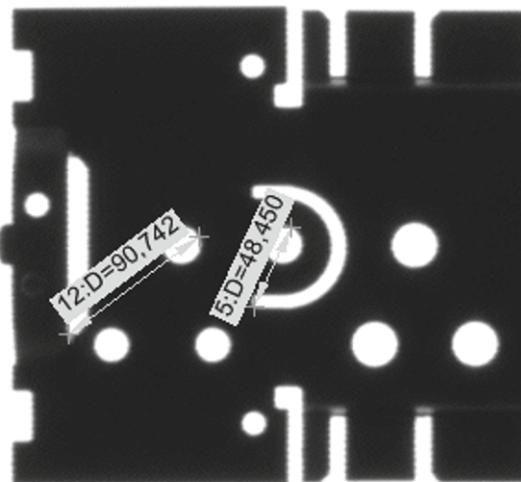


Abbildung 7.7: Maximaler Konturabstand konvexer und ineinander liegender Objekte

Kontaktabstand eines Relais: Abbildung 7.8 zeigt eine Nahaufnahme des Schaltkontakte in einem Relais. Hier muss vor dem Einbau sichergestellt werden, dass der Abstand weder so groß ist, dass das Relais nicht anzieht, noch so klein dass es zu Funkenschlag kommen kann. Diese Prüfung kann sehr gut mithilfe des minimalen Konturabstandes durchgeführt werden.

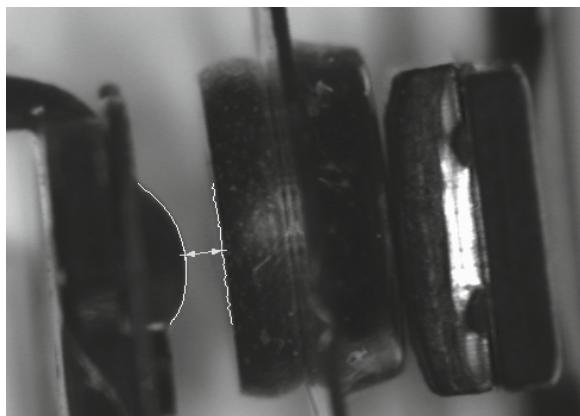


Abbildung 7.8: Prüfung des Kontaktabstandes in einem Relais

Sonstige Abstandsmaße: Man kann Abstandsmaße auf sehr viele verschiedene Arten definieren. In manchen Fällen ist man nicht am direkten Abstand interessiert, sondern am Abstand in einer ganz bestimmten Richtung. Ein Beispiel hierfür ist die Messung des maximalen Gewindedurchmessers an der Zündkerze aus dem Einführungsbeispiel in Abschnitt 1.6. Abbildung 7.9 zeigt den Unterschied zwischen dem einfachen und dem gerichteten maximalen Abstand der beiden Gewindekonturen. Der einfache maximale Abstand erstreckt sich diagonal von einem Ende der unteren Kontur zum gegenüberliegenden Ende der oberen, was nichts mit dem gewünschten Maß zu tun hat, dem Außendurchmesser des Gewindes. Dieses erhält man durch eine in 90° -Richtung orientierte Abstandsmessung. Zur Ermittlung dieses Maßes legt man zwei Geraden in das Bild, die im rechten Winkel zur gewünschten Richtung ausgerichtet sind, und verschiebt jede davon so lange, bis sie genau einen Punkt einer der beiden Konturen berührt. Der gesuchte Außendurchmesser entspricht dann dem lotrechten Abstand der beiden Geraden.

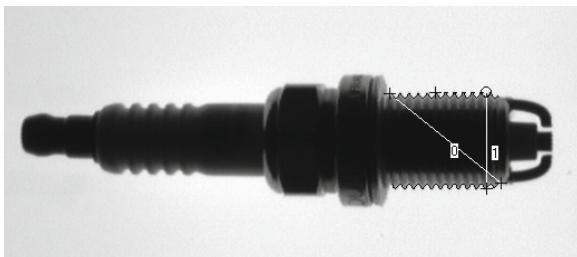


Abbildung 7.9: Unterschied zwischen gerichtetem und einfacherem maximalen Konturabstand

Bei der Berechnung gerichteter Abstände ist es wichtig, darauf zu achten, ob das Objekt gedreht wurde. Abbildung 7.10 zeigt den Effekt einer Messung des maximalen Abstands in 90° -Richtung, bei der nicht auf die Rotation des Messobjekts geachtet wurde.

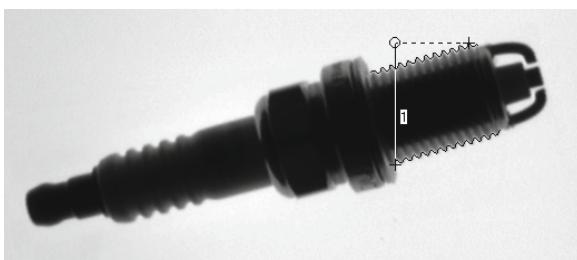


Abbildung 7.10: Gerichtete Abstandsmessung ohne Rotationskorrektur

Die Veränderung des Abstandmaßes bei einer Rotation des Objekts kann je nach Anwendung auch ein gewünschter Effekt sein, es gibt eben sehr viele verschiedene Möglichkeiten, Abstände zwischen Konturen zu definieren. Eine davon wollen wir noch kurz erwähnen, bevor wir uns anderen Maßen zuwenden. Abbil-

dung 7.11 zeigt einen Ausschnitt aus einem Elektronikstanzteil, mit dem wir uns in Abschnitt 7.3 noch eingehender befassen werden. Die dicke schwarze Linie zwischen den weiß hervorgehobenen Kontakten bezeichnet den zu messenden Abstand. Man könnte jetzt versuchen, die eben beschriebene Methode für eine Richtung von null Grad anzuwenden, aber dabei entsteht ein Problem. Da sich die Konturen überlappen, gibt es nie einen definierten nächsten oder fernsten Punkt, an dem die Verschiebung der parallelen Geraden endet. In diesem bei der Prüfung elektronischer Bauelemente, seien es Schalter oder Leiterplatten, sehr häufigen Fall muss man statt der lotrechten Messung zwischen parallelen Geraden eine Art von „Antastmessung“ durchführen. Dabei wird von jedem Punkt der einen Kontur in der definierten Richtung eine Linie gezogen. Wenn diese die andere Kontur schneidet, wird ein Maß abgenommen, wenn nicht, wird diese Linie ignoriert. Abbildung 7.11 zeigt das Prinzip und das Ergebnis dieser Messung.

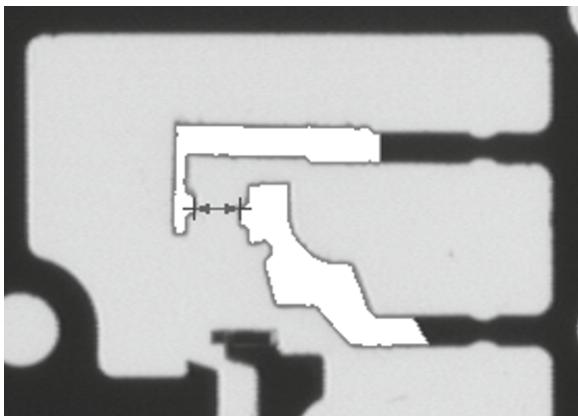


Abbildung 7.11: Prinzip und Ergebnis einer Antastmessung zwischen Konturen

Programmhinweis: In **NeuroCheck** sind für die Antastmessung die Messregeln „Minimaler/maximaler Abstand gegenüber“ definiert, für die Messungen zwischen parallelen Loten die Messregeln „Minimaler/maximaler Abstand in Suchrichtung“.

7.2.3 Winkelmessungen

Winkelmaße können unter anderem dafür verwendet werden, die relative Lage von Objekten zueinander zu prüfen. Die einfachste Methode einen Winkel in einem Bild zu messen, beruht darauf, dass man die Richtung einer Geraden als Winkel angeben kann und dass eine Gerade durch zwei Punkte in einer Ebene exakt definiert ist. Sobald man also zwei Objektschwerpunkte hat, kann man durch diese eine Gerade legen und deren Winkel nach Gleichung (7.4) berechnen. Darin sind Δx und Δy die Unterschiede der x- bzw. y-Koordinaten der beiden Objektschwer-

punkte. Da der Tangens π -periodisch ist, muss man dabei einige Fallunterscheidungen bezüglich der Vorzeichen treffen und außerdem darauf achten, dass für $\Delta x = 0$ der Winkel $\pi/2$ wird, wenn $\Delta y \geq 0$ ist, sonst $-\pi/2$, aber davon abgesehen ist dies eine elementare Messung.

$$\alpha = \arctan\left(\frac{\Delta y}{\Delta x}\right). \quad (7.4)$$

Abbildung 7.12 zeigt zwei Beispiele für diese Methode. Im ersten Beispiel wurden die Schwerpunkte der beiden Stanzlöcher links mit einer Geraden verbunden, dann deren Richtung als Winkel zur Horizontalen berechnet. Im zweiten Beispiel wurden jeweils die Schwerpunkte der unteren beiden Löcher der Dreiergruppe rechts mit dem des oberen verbunden und dann der Öffnungswinkel zwischen diesen beiden Strahlen bestimmt.

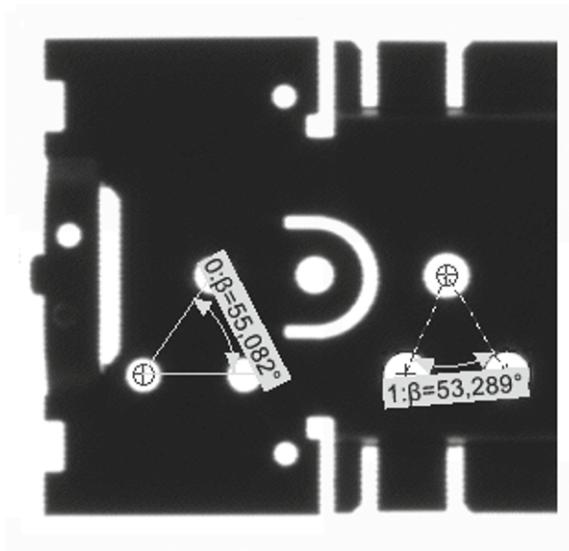


Abbildung 7.12: Winkelmessungen an gestanztem Blech

Der schwierigere Typ der Winkelmessung, die Bestimmung des Winkels einer Objektkontur, wird uns in den folgenden beiden Abschnitten noch beschäftigen.

7.3 Formkontrolle an Elektronikstanzteilen

7.3.1 Prüfaufgabe

Abbildung 7.13 zeigt einen Ausschnitt aus einem Elektronikstanzteil. Eine der Prüfaufgaben an diesem Teil bestand darin, eine Verbiegung der Kontakte festzustellen, die zu Funktionsstörungen des fertigen Produktes führen würde.

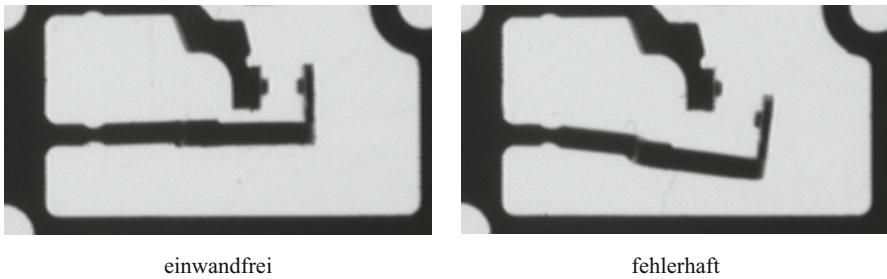


Abbildung 7.13: Elektronikstanzteile im Durchlicht

Zur Kontrolle werden die Teile über eine Leuchtplatte geführt; da die Teile seitlich geführt werden und die Fertigung getaktet erfolgt, ist es sehr einfach ein gut positioniertes Standbild von jedem einzelnen Element zu erhalten: Das Bild wird in dem Moment aufgenommen, in dem die Teile für einen Arbeitsgang angehalten werden. Das Signal für die Positionierung eines neuen Elementes unter der Kamera geht einfach an das Auswerteprogramm, das daraufhin das Kamerabild aufnimmt und verarbeitet. Eine Positions korrektur ist nicht erforderlich, die Suchbereiche können gemäß dem linken Bild der Abbildung 7.14 fest positioniert werden.

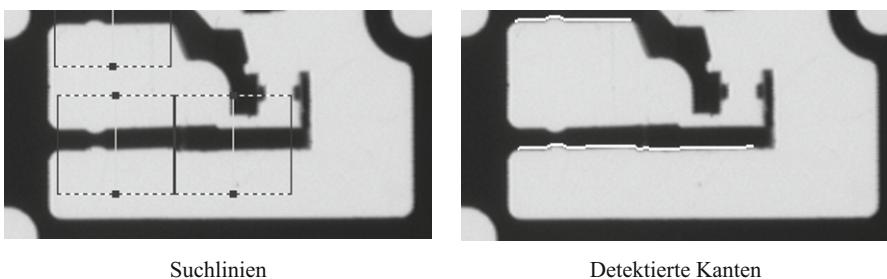


Abbildung 7.14: Suchlinien für Stanzteilkontrolle und detektierte Kanten der Kontakte

Auf diesen Suchlinien werden dann Kanten erzeugt. Dazu wird der erste Hell-Dunkel-Übergang gesucht, dann bis zu den Rändern der im linken Bild der Abbildung 7.14 angedeuteten Umgebungen die Kontur verfolgt. Im rechten Bild der Abbildung 7.14 sind die detektierten Kanten weiß hervorgehoben.

Wenn man noch einmal das fehlerhafte Element in Abbildung 7.13 betrachtet, ist klar, worin die Prüfaufgabe besteht: Das entscheidende Kriterium ist der Winkel des Kontaktes zur Horizontalen. Dabei besteht ein wesentlicher Unterschied zu den Winkelmessungen in Abschnitt 7.2.3: Dort waren die Richtungen jeweils in Form einer Geraden durch zwei Punkte definiert, hier geht es um den Winkel einer Kontur, die bei genauerem Hinsehen an mehreren Punkten ihre Richtung ändert. Im folgenden Abschnitt werden wir uns daher mit der Frage beschäftigen, wie sich der Winkel einer Kontur bestimmen lässt.

7.3.2 Berechnung von Ausgleichsgeraden

Die Winkelmessung in Abschnitt 7.2.3 war deshalb so einfach, weil dort durch jeweils zwei Objektschwerpunkte unmittelbar eine Gerade gegeben war, also ein Gebilde mit einer klar definierten mathematischen Beschreibung. In dieser Anwendung haben wir es hingegen mit einer Kontur zu tun, die ja nichts anderes ist als eine Ansammlung einzelner Punkte. Wenn man an die detektierte Kontur in Abbildung 7.14 an verschiedenen Stellen eine Tangente legte, erhielte man immer wieder andere Winkel, da die Kontur mehrfach ihre Richtung ändert.

Es stellt sich mithin die Frage, wie die Richtung einer Kante eigentlich zu definieren ist. Intuitiv ist die Antwort wiederum klar. Ein Mensch würde jetzt – bildlich gesprochen – ein Lineal möglichst gut passend an den Kontakt anlegen und dessen Winkel messen. Das kann man auch so ausdrücken, dass er versucht, die unregelmäßige Kante durch eine Gerade anzunähern. Und genau das kann man auch digital im Bild tun. Zu diesem Zweck verwenden wir den Begriff der *Ausgleichsgeraden*. Diese ist in der mathematischen Statistik (Bronstein et al. 2005) als diejenige Gerade definiert, für die die Summe der vertikalen Abstandsquadrate aller gegebenen Punkte von der Geraden minimal ist. Wir stellen diese Methode jetzt so vor, wie es in der Statistik üblich ist, da so die Gleichungen einfacher werden. Danach gehen wir kurz auf die Abwandlungen ein, die für die Anwendung in der praktischen Bildverarbeitung notwendig sind.

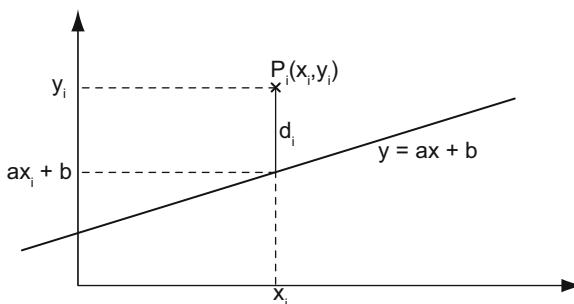


Abbildung 7.15: Ansatz zur Berechnung einer Regressionsgeraden

Abbildung 7.15 veranschaulicht diese Definition anhand eines von n Punkten, die durch eine Gerade angenähert werden sollen. Für diese Gerade macht man den allgemeinen Ansatz

$$y = ax + b. \quad (7.5)$$

Für jeden Punkt $P_i = (x_i, y_i)$ ist der vertikale Abstand von dieser Geraden:

$$d = y_i - ax_i - b. \quad (7.6)$$

Die Summe der vertikalen Abstandsquadrate aller n Punkte von der Geraden ist demnach:

$$D = \sum_{i=1}^n (y_i - ax_i - b)^2. \quad (7.7)$$

Die Parameter a und b sind nun so zu bestimmen, dass diese Summe minimal wird. Die notwendige Bedingung hierfür ist, dass die partiellen Ableitungen der Summe nach beiden Parametern verschwinden. Die partiellen Ableitungen können wegen der Linearität der Summation unter dem Summenzeichen vorgenommen werden, womit die Bedingungen lauten:

$$\frac{\partial D}{\partial a} = -2 \sum (y_i - ax_i - b) x_i = 0, \quad (7.8)$$

$$\frac{\partial D}{\partial b} = -2 \sum (y_i - ax_i - b) = 0, \quad (7.9)$$

woraus die aus der Statistik bekannten Gleichungen für die lineare Regression folgen:

$$a = \bar{y} - b \bar{x}, \quad (7.10)$$

$$b = \frac{\sum x_i y_i - n \bar{x} \bar{y}}{\sum x_i^2 - n \bar{x}^2}. \quad (7.11)$$

mit \bar{x} und \bar{y} als den Mittelwerten der x- bzw. y-Koordinaten der Punkte.

Wir sprachen oben schon an, dass man in der praktischen Bildverarbeitung ein wenig anders verfahren muss. An der Methodik ändert sich prinzipiell nichts, man

minimiert weiterhin die Summe der Abstandsquadrate, aber der Abstand muss aus zwei Gründen anders berechnet werden:

1. Vertikale Regressionsgeraden lassen sich mit der obigen Methode nicht berechnen.
2. Man erhält für die Zwecke der Bildverarbeitung bessere Ergebnisse, wenn man statt des vertikalen Abstandes den lotrechten Abstand zwischen Punkt und Gerade benutzt.

Iterative Berechnung: Gewöhnlich berechnet man zunächst eine Regressionsgerade in der oben beschriebenen Weise und transformiert dann das Koordinatensystem so, dass diese erste Regressionsgerade zur x-Achse wird. In diesem Koordinatensystem sind die vertikalen Abstände der einzelnen Punkte nahezu lotrecht zur Achse. Durch erneute Regression und anschließende Rücktransformation kann so die Näherungsgerade verbessert werden. Damit ist auch das Problem vertikaler Regressionsgeraden gelöst, da nach der Koordinatentransformation eine annähernd horizontale Regressionsgerade berechnet wird und die endgültige Vertikale erst bei der Rücktransformation entsteht.

Diese iterative Berechnung eröffnet außerdem die Möglichkeit, „Ausreißer“ zu unterdrücken. Eine kleine Störung an einer Objektkontur kann die Regressionsgerade erheblich beeinflussen, je nachdem, an welcher Stelle sie auftritt. Solche Störungen können z. B. durch Oberflächenstrukturen erzeugt werden. Die Unterdrückung dieser Störreinflüsse kann man erreichen, in dem in jedem Iterationsschritt die Punkte verworfen werden, die von der aktuellen Regressionsgeraden am weitesten entfernt liegen. Je kleiner der noch zulässige Abstand der Punkte von der aktuellen Regressionsgeraden gewählt wird, desto schneller stabilisiert sich die Iteration auf eine bestimmte Richtung. Dadurch lässt sich im Allgemeinen eine sehr gute Annäherung an die vom Menschen wahrgenommene Achse eines Objekts erreichen. Wenn der Abstand jedoch zu klein gewählt wird, kann es sein, dass eine schlechte Anfangsnäherung zum Verwerfen eigentlich relevanter Punkte führt, die nur von dieser ersten Geraden zu weit entfernt sind.

7.3.3 Messung des Konturwinkels

Kehren wir nach diesem Ausflug in die Berechnung von Ausgleichsgeraden zurück zur Prüfung der Stanzteilkontakte. Wenn für jedes der erzeugten Kantenstücke nach dem soeben beschriebenen Verfahren die Ausgleichsgeraden berechnet wurden, dann liegt mit der Steigung dieser Geraden der Tangens des gesuchten Winkels vor. Dieser muss dann zur Lösung der Prüfaufgabe nur noch mit den zulässigen Toleranzen verglichen werden. Abbildung 7.16 zeigt die Maßlinien für einen einwandfreien und einen defekten Schalter. Tabelle 7.2 fasst das Beispiel zusammen.

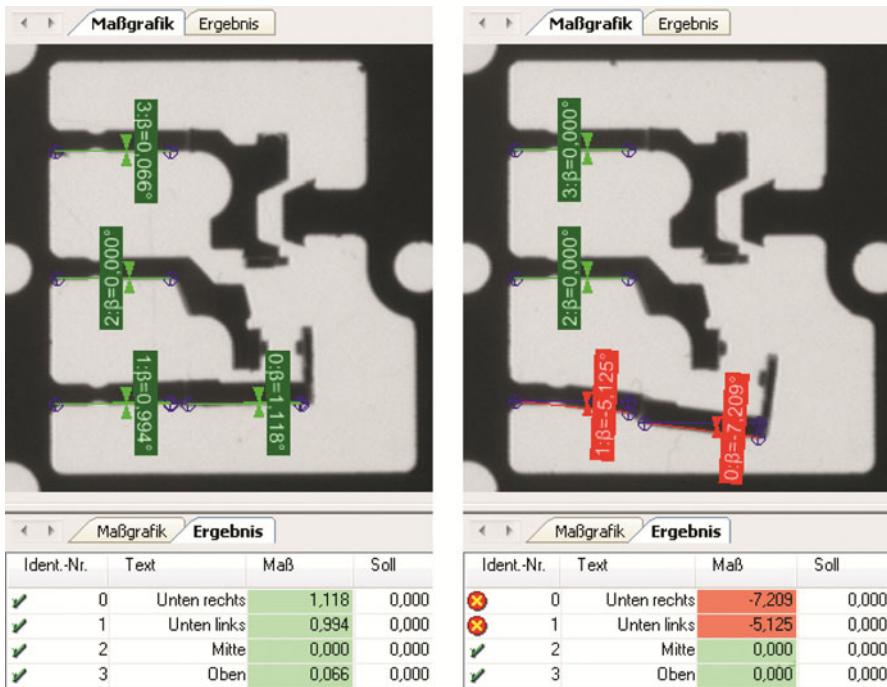


Abbildung 7.16: Messergebnisse an Stanzteilen; (zulässige Maße grün, unzulässige Maße rot)

Beispielbeschreibung 7.2. Die folgende Tabelle fasst das Beispiel aus Abschnitt 7.3 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Maßprüfung
Beleuchtung	Diffuse Hintergrundbeleuchtung mit Leuchtplatte
Prüfschritte	Das Beispiel verwendet folgenden Prüfschrittablauf: <ol style="list-style-type: none"> <i>Bildaufnahme</i> <i>Arbeitsbereiche definieren</i>, setzt Suchlinien für die Antastung der einzelnen Kontaktkanten. <i>Binärschwelle erzeugen</i>, optimiert die Binärschwelle anhand einer Histogrammanalyse. <i>Arbeitsbereiche binär erzeugen</i>, detektiert und verfolgt die Konturen der Kontakte. <i>Ausgleichsgeometrien erzeugen</i>, berechnet für jedes erzeugte Konturstück eine Regressionsgerade. <i>Maße erzeugen</i>, berechnet aus den Parametern der Regressionsgeraden die Winkel der einzelnen Konturstücke. <i>Maße Toleranzen überprüfen</i>, vergleicht die ermittelten Winkel mit den Vorgabewerten.

7.4 Winkelmessung an einem Zahnriemen

Abbildung 7.17 zeigt einen handelsüblichen Zahnriemen. Die Prüfaufgabe besteht darin, den Winkel zwischen den Zähnen und dem Rand des Zahnriemens festzustellen, bevor der Riemen geschnitten und zu dem abgebildeten Ring geformt wird. Abweichungen zwischen dem durch das Zahnrad vorgegebenen Sollwinkel und dem tatsächlichen Winkel der Zähne des Riemens würden zu unruhigem Lauf und verstärkter Abnutzung führen. Dabei ist eine Messgenauigkeit von $0,5^\circ$ gefordert.

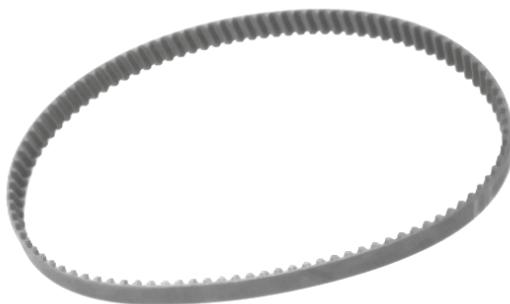


Abbildung 7.17: Zahnriemen (stark überbelichtet, um die Zahnung innen zu zeigen)

7.4.1 Beleuchtungsaufbau

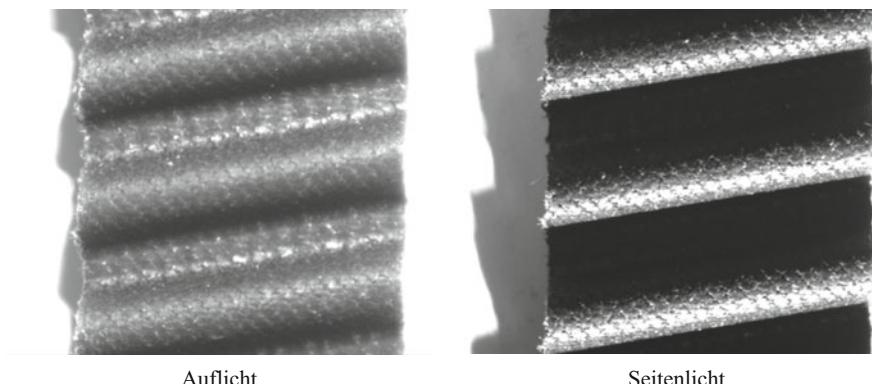


Abbildung 7.18: Zahnriemen unter normaler Auflichtbeleuchtung und gerichteter Beleuchtung

Die Hauptschwierigkeit liegt bei dieser Anwendung darin, eine klare Kante zur Bestimmung des Winkels zu erhalten. Das linke Bild in Abbildung 7.18 zeigt,

dass eine gewöhnliche Auflichtbeleuchtung hier nicht zum Ziel führt. Aufgrund der starken Texturierung des Materials ist es nicht möglich, eine exakte Kantensegmentierung durchzuführen. Man muss den Zahnriemen stattdessen sehr flach beleuchten, so dass jeder Zahn einen Schatten auf den dahinter liegenden projiziert. Das Ergebnis ist rechts in Abbildung 7.18 zu sehen.

Abbildung 7.19 zeigt das Prinzip dieser flachen gerichteten Beleuchtung in einer Seitenansicht. Man sieht, dass die Höhe, in der der Schatten projiziert wird, von folgenden Größen abhängt:

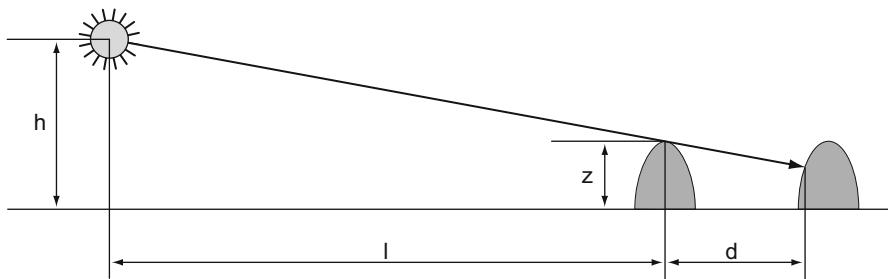


Abbildung 7.19: Prinzip der Beleuchtung des Zahnriemens

- Höhe h der Lichtquelle über der Ebene des Zahnriemens
- Abstand l zwischen Lichtquelle und beleuchtetem Zahn
- Höhe z eines Zahns
- Abstand d zwischen den Zähnen

Wenn die Lichtstrahlen parallel zum Zahnriemen einfallen, ist der Abstand zwischen Lichtquelle und beleuchtetem Zahn an der linken und rechten Kante des Zahnriemens aufgrund der Schräglage der Zähne unterschiedlich. Dadurch wäre auch die Projektionshöhe des Schattens unterschiedlich, der Schatten verliefe nicht parallel zur Oberkante der Zähne. Der Beleuchtungsaufbau muss daher so erfolgen, dass das Licht senkrecht zur Orientierung der Zähne einfällt. Bei einem korrekt orientierten Zahn ist der projizierte Schatten dann parallel zur Oberkante des Zahnes. Wenn einer der Zähne von der Sollorientierung abweicht, äußert sich dies entsprechend in einer Abweichung der projizierten Schattenlinie.

Man muss noch beachten, dass sich dabei zwei Fehlerquellen überlagern können: Der Schatten entsteht an einem Zahn, der möglicherweise fehlerhaft ist, und wird auf einen zweiten Zahn projiziert, der andere Fehler aufweisen kann. In dem vorliegenden Beispiel erwies sich die Genauigkeit im Rahmen der Spezifikationen als völlig ausreichend. Für noch höhere Genauigkeit kann man die erste Fehlerquelle ausschalten, indem man die Schattenlinie durch ein genau definiertes Hindernis im Strahlengang erzeugt, beispielsweise eine präzise gefertigte und ausgerichtete Metallkante. In diesem Fall wird die Schattenlinie nur noch die Fehler des Zahnes zeigen, auf den sie projiziert wird.

7.4.2 Kantenerzeugung

Zur Messung des Winkels benötigen wir wiederum eine Gerade, die prinzipiell genauso erzeugt werden könnte wie in Abschnitt 7.3, d. h. durch Berechnung einer Regressionsgeraden aus allen Punkten einer binär gefundenen Kante. Jedoch zeigt das Material des Zahnrückens auch unter der optimierten Beleuchtung noch eine deutliche Strukturierung, die eine binäre Kantensegmentierung erschwert. Es ist daher von Vorteil, die in Abschnitt 4.4 beschriebene Methode der Kantendetektion zu verwenden. Bei dieser Methode wird ein Bildpunkt als Bestandteil einer Kante betrachtet, wenn in einer Umgebung dieses Punkts ein bestimmter Grauwertgradient erreicht wird. Die Umgebung kann danach noch mit der ebenfalls in Abschnitt 4.4 eingeführten Interpolationsfunktion untersucht und der Ort des Kantenpunkts subpixelgenau bestimmt werden. Dank der hohen Genauigkeit, die sich auf diese Weise erreichen lässt, kann man sich zur Reduzierung des Rechenaufwandes darauf beschränken, einzelne Suchlinien zu bearbeiten anstelle der gesamten Kante.

Die stabilste Kantendetektion erhält man in diesem Fall bei der Suche nach der ersten ansteigenden Kante aus dem Schattenbereich heraus. Dadurch kann exakt die Projektionslinie des Schattens auf dem dahinter liegenden Zahn bestimmt werden. Aus jeder der im linken Bild von Abbildung 7.20 hell eingezeichneten Suchlinien entsteht dann genau ein subpixelgenauer Kantenpunkt. Aus allen diesen Punkten kann auf dieselbe Weise wie in Abschnitt 7.3.2 eine Ausgleichsgerade berechnet werden, deren Winkel dann gemessen werden kann, wie es im rechten Bild von Abbildung 7.20 zu sehen ist.

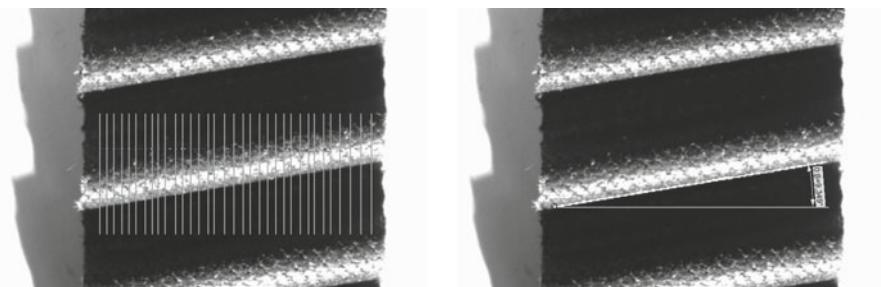


Abbildung 7.20: Suchlinien zur Kantenerzeugung am Zahnrücken und Ergebnis der Winkelmessung

Tabelle 7.3 fasst die Anwendung in der beschriebenen Form zusammen. Es sei noch angemerkt, dass die Anwendung in dieser Form positionskritisch ist. Wenn die Bildaufnahme nicht genau zum richtigen Zeitpunkt stattfindet, können sich die Suchlinien relativ zum Bild so verschieben, dass noch Kanten innerhalb des stark texturierten Bereiches eines Zahns gefunden werden. Abbildung 7.21 zeigt eine solche Verschiebung der Suchlinien. In diesem Fall muss eine Positionskorrektur vorgenommen werden. Hierfür ist eine binär detektierte Kante stabil genug. Wir

werden auf die Positions korrektur hier nicht näher eingehen, da sie demselben Schema folgt wie beispielsweise in den Anwendungen von Kapitel 5.

Beispielbeschreibung 7.3. Die folgende Tabelle fasst das Beispiel aus Abschnitt 7.4 in der Fassung ohne Positions korrektur zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Maßprüfung
Beleuchtung	sehr flach einfallende Auflichtbeleuchtung angepasst an die Schrägstellung der Zähne des Zahnriemens
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt Suchlinien aus dem Schattenbereich über die nächste Projektionskante. Bei nicht ausreichend exakter Triggerung müssen diese Suchlinien aufgrund der Position einer binär detektierten Kante nachgeführt werden. 3. <i>Kanten erzeugen</i>, bestimmt subpixelgenau die Übergänge aus dem Schattenbereich über die Projektionslinie in den beleuchteten Bereich des nächsten Zahns. 4. <i>Ausgleichsgeometrien erzeugen</i>, berechnet aus den detektierten Kantenpunkten eine Regressionsgerade. Eine geringe Anzahl fehldetektierter Kantenpunkte kann durch Iteration eliminiert werden. 5. <i>Maße erzeugen</i>, berechnet den Winkel der Regressionsgeraden.

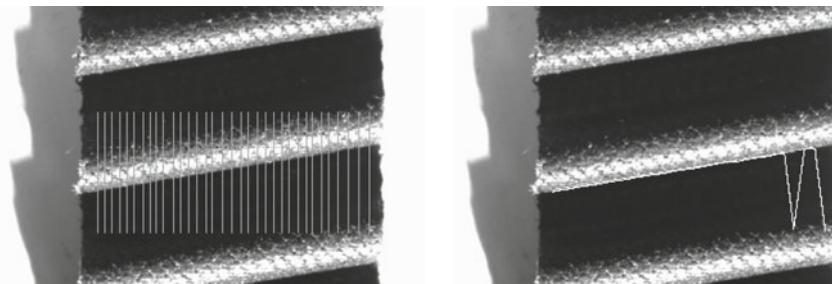


Abbildung 7.21: Fehldetektion von Kantenpunkten ohne Positions korrektur

7.5 Formkontrolle an Spritzgussteil

Objektkonturen können außer durch Geraden, wie in Abschnitt 7.3, auch durch andere Ideal geometrien angenähert werden, beispielsweise Kreise. Auf diese Weise lassen sich Radien messen und Deformationen feststellen. Abbildung 7.22 zeigt ein Kunststoffteil, an dem die Radien der abgerundeten Ecken und die Anordnung der Löcher zu prüfen sind, um Deformationen festzustellen.



Abbildung 7.22: Spritzgussteil im Durchlicht

7.5.1 Bestimmung von Radien

Um festzustellen, ob sich das Prüfteil insgesamt verzogen hat, kann man die Relationen der drei Öffnungen zueinander vermessen. Diese sind in der kontrastreichen Durchlichtaufnahme durch eine Binärsegmentierung sehr leicht zu finden. Nach Berechnung ihrer Schwerpunkte kann man die gegenseitige Lage mit folgenden Maßen beschreiben:

- Abstände der Schwerpunkte, wie in Abschnitt 7.2.1
- Winkel zwischen den Verbindungslien, entsprechend Abschnitt 7.2.3
- Radius eines Kreises durch die Schwerpunkte

Die ersten beiden Maße bringen nichts Neues gegenüber Abschnitt 7.2; das dritte Maß ist ebenfalls leicht zu berechnen, da ein Kreis durch drei Punkte genau bestimmt ist. Interessanter ist die Bestimmung der Radien an den abgerundeten Ecken. Da diese nur Kreisausschnitte darstellen, müssen zunächst vollständige Kreislinien an diese Konturen angepasst werden.

Zu diesem Zweck erzeugt man Teilstücke der gerundeten Konturen. Die dazu verwendeten Suchlinien, von denen eine in Abbildung 7.23 gezeigt sind, sollten durch die Scheitel der Rundungen verlaufen und keine zu breiten Kantenstücke ausschneiden, da sonst Teile der nahezu geraden Seitenabschnitte in die Kreisberechnung mit einbezogen würden. Aus diesem Grund ist eine vorherige Positions-korrektur vorteilhaft. Ähnlich wie in Abschnitt 3.3.4 ist auch hier die stabilste Po-

sitionsreferenz eine Objektgruppe, die in diesem Fall von den drei Öffnungen gebildet wird.

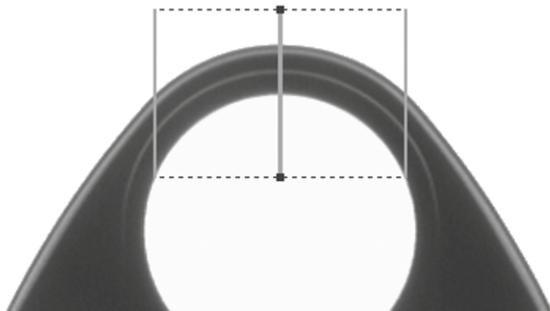


Abbildung 7.23: Eine der Suchlinien zur Erzeugung von Konturstücken an den abgerundeten Ecken

An die erzeugten Konturstücke können dann vollständige Kreise angepasst werden. Abgesehen davon, dass hier drei Parameter (x- und y-Koordinate des Mittelpunkts und Radius) bestimmt werden müssen, ist das Prinzip das gleiche wie bei der Bestimmung einer Ausgleichsgeraden in Abschnitt 7.3.2, so dass wir hier nicht näher auf den Algorithmus eingehen. Abbildung 7.24 zeigt die resultierenden Kreise für die drei Rundungen und dazu den Kreis durch die Schwerpunkte der drei Öffnungen. Man sieht, dass die Ausgleichskreise sich genau der Form der Rundungen anpassen und sich am Übergang zu den nahezu geraden Seitenabschnitten von der Objektkontur entfernen.

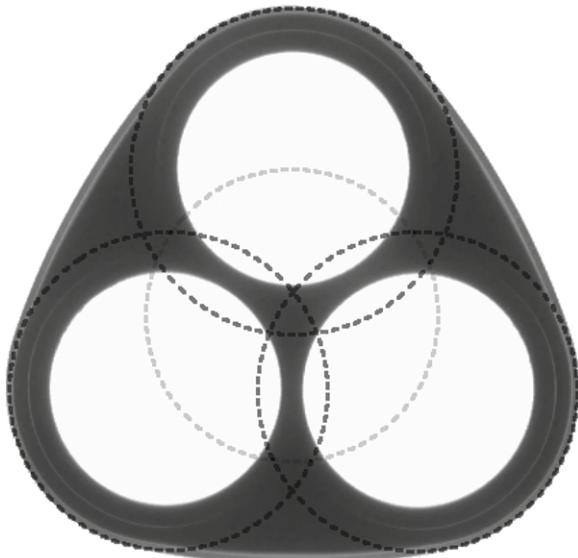


Abbildung 7.24: Ausgleichskreise für die abgerundeten Ecken und Kreis durch die Schwerpunkte der Öffnungen (Kreisliniен verdickt)

Mit der Bestimmung der Kreisparameter ist der gesuchte Eckenradius als einer der Kreisparameter bereits bekannt. Darüber hinaus können noch minimale und maximale Abstände der tatsächlichen Kontur von dem berechneten Kreiszentrums bestimmt werden. Die Differenz dieser beiden Größen wird auch als *Rundheit* bezeichnet und kann als Maß für die Kreisähnlichkeit dienen. Ein idealer Kreis hat dabei eine Rundheit von null. Dieses Maß kann Auskunft über Deformationen der Außenkontur geben. Deformationen des gesamten Prüfteils schlagen sich auch im Radius des Kreises durch die Schwerpunkte der Öffnungen nieder. Dort ist allerdings die Rundheit immer exakt null, da ein Kreis durch drei Punkte exakt bestimmt ist. Eine Ausgleichsrechnung ist in diesem Fall nicht erforderlich. Der Prüfablauf ist in Tabelle 7.4 zusammengefasst.

Beispielbeschreibung 7.4. Die folgende Tabelle fasst das Beispiel aus Abschnitt 7.5 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Maßprüfung
Beleuchtung	Diffuse Hintergrundbeleuchtung.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, einfacher Suchbereich zur Detektierung des gesamten Teils 3. <i>Binärschwelle erzeugen</i>, kann hier einfach manuell gesetzt werden, da die Hintergrundbeleuchtung eine stabile Helligkeitswert erzeugt. 4. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert das gesamte Prüfteil als Objekt. 5. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert innerhalb des Prüfteils die drei Öffnungen. 6. <i>Arbeitsbereiche Position bestimmen</i>, berechnet aktuelle Position und Ausrichtung des Prüfteils anhand der Gruppe von Öffnungen. 7. <i>Arbeitsbereiche definieren</i>, setzt Suchlinien auf die Scheitel der abgerundeten Ecken. 8. <i>Arbeitsbereiche positionieren</i>, verschiebt diese Suchlinien entsprechend der festgestellten Referenzposition. 9. <i>Arbeitsbereiche binär erzeugen</i>, erzeugt an jeder der abgerundeten Ecken ein Stück aus der Kontur des Prüfteils als Objekt. 10. <i>Arbeitsbereiche kombinieren</i>, fasst die segmentierten Öffnungen und die erzeugten Konturstücke zur gemeinsamen Vermessung zusammen. 11. <i>Ausgleichsgeometrien erzeugen</i>, berechnet den exakten Kreis durch die Schwerpunkte der drei Öffnungen und die Bestkreise für die Konturstücke an den drei abgerundeten Ecken. 12. <i>Maße erzeugen</i>, berechnet die Radien aller dieser Kreise und außerdem die minimalen und maximalen Abstände der Konturstücke von den Mittelpunkten der zugehörigen Bestkreise, um Deformationen festzustellen.

7.5.2 Bemerkungen zur Ausgleichskreisberechnung

Wie Ausgleichsgeraden können auch Ausgleichskreise iterativ berechnet werden, wobei in jeder Iteration die Punkte mit dem größten Abstand zum Kreis verworfen werden. Dadurch lässt sich in dieser Anwendung bis zu einem gewissen Grad der Einfluss von Konturpunkten eliminieren, die zu den nahezu geraden Seitenabschnitten des Prüfteils gehören.

Ausgleichskreise können außer aus Konturen auch aus Einzelpunkten berechnet werden, wie die Ausgleichsgerade in Abschnitt 7.4. Die iterative Berechnung eines mittleren Ausgleichskreises aus einer Reihe subpixelgenauer Konturpunkte ist die genaueste Methode zur Bestimmung des Durchmessers eines kreisförmigen Werkstücks oder einer Bohrung.

7.6 Hochgenaue Vermessung eines Gewindeflansches

In dieser Anwendung geht es um einen Gewindeflansch, der auf einer NC-Drehmaschine vollautomatisch gefertigt und dann ebenfalls automatisch gereinigt und geprüft wird. Die Prüfaufgabe bestand in einer hochgenauen Vermessung der Gewindetiefe, um frühzeitig den Verschleiß des Schneidwerkzeugs feststellen zu können, da ansonsten unbrauchbare Flansche entstehen. Im Extremfall kann sogar das Werkzeug brechen und weitere Schäden verursachen. Abbildung 7.25 zeigt das Schneidergebnis bei verschiedenen Zuständen des Werkzeuges. Es ist deutlich zu sehen, wie die ursprüngliche klare Form des Gewindes bei zunehmendem Werkzeugverschleiß abgeschliffen erscheint, bis zum Schluss nur ein völlig zerstörter Rohling übrig bleibt.



Abbildung 7.25: Gewindeflansche, mit unterschiedlich abgenutzten Werkzeugen geschnitten

Bei einer geforderten Messgenauigkeit von zwei Mikrometer befindet man sich schon nahe am Bereich der Lichtwellenlängen. Berücksichtigt man außerdem, dass die nominelle Gewindetiefe einen Millimeter beträgt und man etwas Spielraum für Positionsschwankungen und die Antastung der Gewindekontur benötigt,

so erkennt man schnell, wo die Herausforderungen bei dieser Aufgabe liegen. Einmal stellt die für solche Genauigkeiten notwendige Bildqualität höchste Anforderungen an die Beleuchtung und Sensorik. Zum anderen ist auf der algorithmischen Seite bei Einsatz einer Kamera mit 2448×2050 Bildpunkten eine algorithmische Genauigkeit von $1/3$ Bildpunkt erforderlich, da man von einem Bildfeld von etwa zehn Millimeter ausgehen muss.

7.6.1 Beleuchtung und Bildaufnahme

Wie bei vielen Vermessungsaufgaben stellen Beleuchtung und Bildaufnahmetechnik auch hier die entscheidenden Komponenten dar. Dabei hat man überraschend wenig Gestaltungsfreiheit, da die Anforderungen und die physikalischen Randbedingungen praktisch vorschreiben, welcher Art die verwendeten Geräte sein müssen:

Beleuchtung: Die erforderlichen präzisen Konturen lassen sich nur mit einer Hintergrundbeleuchtung erreichen. Häufig wird man dafür eine LED-Beleuchtung wählen, die stromstabilisiert angesteuert werden muss, um Helligkeitsschwankungen zu vermeiden. Ansonsten käme es zu Schwebungen zwischen der Beleuchtung und der Kamera, die eine stabile und wiederholbare Subpixelberechnung unmöglich machen würden.

Ein weiteres Problem sind Verfälschungen der Kontur durch schräg einfallendes Seitenlicht bei üblichen Hintergrundbeleuchtungen. Dieser Effekt kann bei leichter Überbelichtung sogar schon bei dem flach auf der Beleuchtung aufliegenden Stanzteil aus Abschnitt 7.2 beobachtet werden. Da die Hintergrundbeleuchtung hier mindestens um den Radius des Flansches von der eigentlichen Messebene entfernt sein muss, wie die schematische Ansicht in Abbildung 7.26 zeigt, wäre der Effekt noch deutlich stärker. Daher verwendet man in einem solchen Fall eine telezentrische Hintergrundbeleuchtung, die einen exakt parallelen Strahlengang aufweist.

Objektiv: Das Objektiv muss in dieser Anwendung exakt auf die Messebene fokussiert sein und darf bei kleinen Abstandsschwankungen keine Maßstabsveränderung zeigen. Aus diesem Grund kommt nur ein telezentrisches Objektiv in Frage.

Kamera: Je näher die Auflösung des Bildes der geforderten Messauflösung kommt, desto stabiler wird die Messung, da mehr „echte“ Information – im Gegensatz zu rekonstruierter Information – zur Verfügung steht. Daher muss die Kamera möglichst hochauflösend sein.

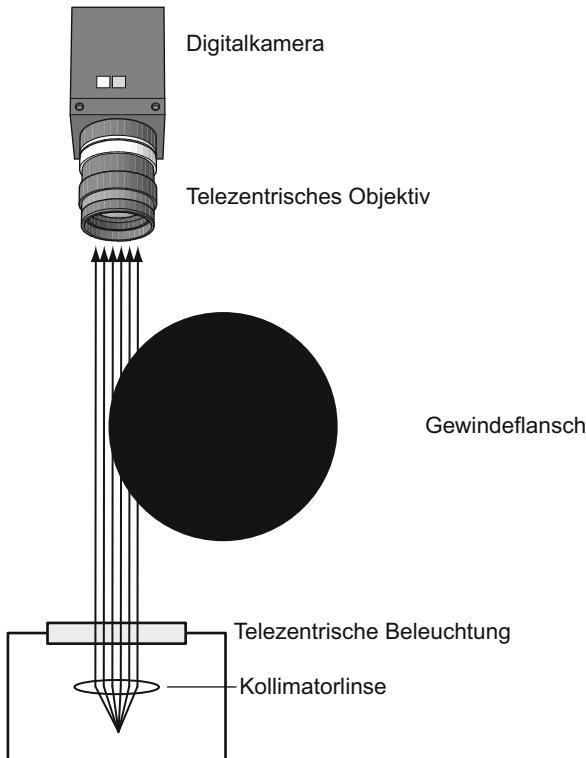


Abbildung 7.26: Schematische Ansicht auf den Messaufbau

7.6.2 Subpixelgenaue Vermessung der Gewindetiefe

Aus der Beschreibung der Beleuchtungs- und Aufnahmesituation im vorigen Abschnitt geht bereits hervor, dass man bei einer so genauen Vermessung, wie sie in dieser Anwendung gefordert ist, eine entsprechende mechanische Präzision verlangen muss. Nachträgliche Positions korrekturen im Bildverarbeitungssystem sind hier kaum möglich, da bei einer falschen Ausrichtung des Prüfteils relativ zu Beleuchtung und Kamera bereits Verfälschungen durch veränderten Strahlengang und Schärfenebene auftreten. Man muss daher davon ausgehen, dass die Prüfteile absolut präzise im Aufnahmefeld positioniert sind. Dies betrifft vor allem die vertikale Ausrichtung.

Unter dieser Voraussetzung kann man bei der Messung folgendermaßen vorgehen: Man erzeugt eine künstliche, exakt senkrechte Referenzgerade und misst von dieser den minimalen und maximalen Abstand zur Gewindekontur in horizontaler Richtung. Die höchste Genauigkeit lässt sich erreichen, indem die gesamte Kontur subpixelgenau interpoliert wird, aber bei der durch den telezentrischen Strahlengang, die konstante Beleuchtung und digitale Bildübertragung erreichten Präzisi-

on, genügt es vollauf, diejenigen Punkte der Kontur pixelgenau festzustellen, die der Referenzgeraden am nächsten und am fernsten liegen. In einer Umgebung dieser Punkte führt man dann die Subpixelinterpolation durch und bestimmt den exakten Abstand. Die Abbildung 7.27 zeigt die Maßlinien für die Bestimmung der Gewindetiefe. Die Anwendung ist in Tabelle 7.5 zusammengefasst.

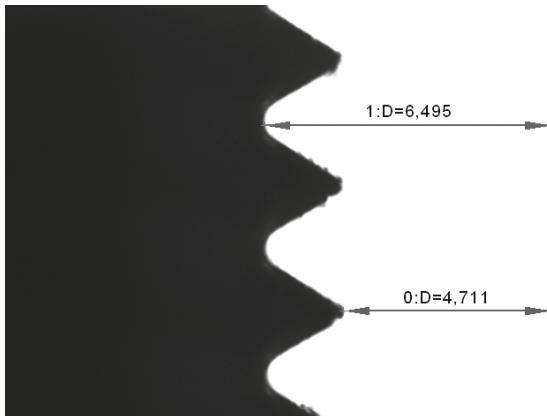


Abbildung 7.27: Maßlinien für Gewindetiefe

Beispielbeschreibung 7.5. Die folgende Tabelle fasst das Beispiel aus Abschnitt 7.6 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Maßprüfung
Beleuchtung	telezentrische Hintergrundbeleuchtung.
Bildaufnahme	telezentrisches Objektiv, exakt auf die Mittelebene des Flansches fokussiert; hochauflösende Digitalkamera.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt die Suchlinie für die Gewindekontur und die künstliche senkrechte Referenzlinie. 3. <i>Binärschwelle erzeugen</i>, optimiert die Binärschwelle durch eine Histogrammanalyse. 4. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert die Konturlinie des Gewindes zunächst pixelgenau als Ausgangspunkt der Messung. 5. <i>Arbeitsbereiche kombinieren</i>, fasst die senkrechte Referenzlinie und die Gewindekontur zur gemeinsamen Vermessung zusammen. 6. <i>Maße erzeugen</i>, berechnet den minimalen und maximalen Abstand in horizontaler Richtung zwischen der Referenzlinie und der Gewindekontur. Dabei werden zuerst die minimal und maximal weit entfernten Pixel gesucht, dann in deren Umgebung eine Subpixelinterpolation durchgeführt. 7. <i>Maße verknüpfen</i>, berechnet die Gewindetiefe als Differenz des maximalen und minimalen Abstandes.

7.7 Kalibrierung

Wir haben das für die Vermessung sehr wichtige Thema der Kalibrierung nur kurz im Abschnitt 7.2 angeschnitten. In den nachfolgenden Beispielen sind wir im Interesse einer knapperen, einfacheren Darstellung davon ausgegangen, dass:

1. durch die verwendete Sensorik (Optik, Kamera) ein stabiler Abbildungsmaßstab gewährleistet wird.
2. die Abbildung mit korrektem Seitenverhältnis erfolgt, d. h. die Kamera wie heute üblich quadratische Bildelemente aufweist.
3. der Umrechnungsfaktor zwischen Bildpunktgröße und Dimension in der realen Welt an einem geeigneten Referenzteil bestimmt wurde.

Auf die ersten beiden Punkte geht Kapitel 8 ausführlich ein, mit dem letzten wollen wir uns in diesem Abschnitt beschäftigen. Dabei ist der Ausdruck „geeignetes Referenzteil“ von besonderer Bedeutung, denn wenn man ein in der realen Welt genau bekanntes Maß mit der größtmöglichen Genauigkeit optisch gemessen hat, ist die Bestimmung des Umrechnungsfaktors mathematisch trivial. Die Frage ist also, wie man an ein geeignetes Maß kommt, was sich als Referenzteil eignet und wie man es aufnehmen muss.

7.7.1 Kalibriermodus

In vielen Bildverarbeitungssystemen gibt es einen gesonderten Kalibriermodus, in dem anhand eines Kalibriernormals, d. h. eines Musterteils mit bestimmten festgelegten Charakteristika, die Abbildungseigenschaften des Systems bestimmt werden. Solche Kalibriernormale können beispielsweise sein:

- Glasplatten mit geätzten Markierungen
- Hochgenau gefräste und geschliffene Metallteile mit festgelegten Maßen
- Metallplatten mit präzisen Bohrungen

Die mechanische Genauigkeit dieser Kalibriernormale steht außer Frage. Dennoch sehen wir in diesem Ansatz verschiedene Probleme. Das wichtigste Problem ist, dass eine Kalibrierung immer mit demselben Aufbau durchgeführt werden sollte wie die endgültige Prüfung. Die Aufnahmesituation sollte sich zwischen Kalibrierung und Prüfung keinesfalls unterscheiden. Unterschiedliche Beleuchtungen oder Blendeneinstellungen führen unweigerlich zu einem geänderten Abbildungsverhalten und damit unterschiedlichen Kalibriereigenschaften. Wenn also das Kalibriernormal eine andere Größe hat als das Prüfteil, ein anderes Bildfeld benötigt, einen Wechsel von Beleuchtung oder Objektiv erfordert, so sind die Ergebnisse praktisch nicht mehr auf die Prüfsituation übertragbar. Das bedeutet aber, dass jede Prüfsituation unter Umständen ein eigenes Kalibriernormal erfordern kann.

Selbst wenn das Kalibriernormal im gleichen Abstand, mit dem gleichen Objektiv, unter derselben Beleuchtung aufgenommen werden kann wie das Prüfteil,

kann es sich so stark vom Prüfteil unterscheiden, dass die Abbildungseigenschaften sich verändern. Ein wichtiger Unterschied ist beispielsweise die Frage, ob Außenkanten vermessen werden sollen, z. B. die Breite eines Prüfteils, oder Innenkanten, wie etwa der Durchmesser einer Bohrung. Je nach Beleuchtung und sonstigem Aufbau können sich die für die Kalibrierung wesentlichen Bildeigenschaften dabei verändern. Möglicherweise können die bekannten Referenzmaße des Kalibrierteils nicht mit dem gleichen Segmentierungsverfahren bestimmt werden, das in der späteren Prüfung verwendet wird, oder die Parameter müssen unterschiedlich sein. Die Schärfenebene, in der die wesentlichen Eigenschaften erkennbar sind, kann sich von der des Prüfteils unterscheiden, wodurch sich wiederum die Abbildungseigenschaften ändern ...

7.7.2 Prüfungsbezogene Kalibrierungen

Die oben dargestellten Überlegungen führen zu dem Schluss, dass die Kalibrierung nicht nur in derselben Aufnahmesituation vorgenommen werden sollte wie die eigentliche Prüfung, sondern auch direkt am endgültigen Prüfteil. Man benötigt dazu also ein, besser mehrere, „Meisterteile“, Prüfteile, deren zu messende Eigenschaften mit einem anderen Verfahren so genau wie möglich bestimmt wurden – beispielsweise einem mechanischen Verfahren, das für die automatisierte Fertigung nicht geeignet ist. Anhand dieser Teile kann dann das Bildverarbeitungssystem *für genau diese Aufgabe* kalibriert werden. Diese Kalibrierung muss mit demselben Systemaufbau, den gleichen Segmentierungs- und Messverfahren und identischen Parametereinstellungen vorgenommen werden wie die endgültige Prüfung. Nur so kann man sicherstellen, dass der ermittelte Umrechnungsfaktor für die Prüfaufgabe auch tatsächlich geeignet ist.

Wenn es tatsächlich nicht möglich ist, die Referenzmaße direkt am Prüfteil abzunehmen, ist es vorteilhaft, ein Kalibrierteil anzufertigen, das in seinen Eigenschaften dem Prüfteil möglichst nahe kommt. Wenn es um die Vermessung von Außenkanten eines Werkstücks geht, eignet sich gut ein hochgenau gefertigtes Teil prinzipiell ähnlicher, aber vereinfachter Form; bei einem flachen, geraden Teil ein ebenfalls flaches Kalibrierstück mit gerade Kanten, bei einem runden Prüfteil eher ein Zylinder, da sich das Verhalten von Kantenabbildungen zwischen runden und scharfkantigen Teilen deutlich unterscheidet. Bei der Gewindeflanschvermessung in Abschnitt 7.6 hatten wir schon gesehen, dass man bei einem runden Teil sehr präzise auf die Mittelebene fokussieren muss und meist einen sehr kleinen Schärfentiefenbereich verwendet.

Für die Vermessung von Abständen ist eine Metallplatte mit mehreren Bohrungen ein brauchbares Referenzteil, das mit einer modernen NC-Maschine mit sehr hoher Präzision und in einer an das Prüfteil angepassten Größe hergestellt werden kann. Im Durchlicht ist der Zentrumsabstand der Bohrungen in weiten Bereichen unabhängig von Belichtung und Segmentierungsparametern, da eine Aufweitung oder Verengung der Bohrungskonturen durch geänderte Belichtung symmetrisch erfolgt und die Schwerpunktskoordinaten nicht verändert.

7.8 Zusammenfassung

Vermessungsaufgaben: Man kann zwischen drei Typen von Vermessungsaufgaben unterscheiden:

- Montagekontrolle mit relativ niedrigen Genauigkeitsanforderungen
- Formkontrolle, bei der es vor allem auf Beziehungen zwischen Objekten und relative Maße ankommt
- Maßkontrolle, die höchste Genauigkeit und entsprechend präzise Sensorik verlangt

Ablauf: Vermessungsaufgaben sind sehr häufig kantenorientiert. Daher folgen sie im Allgemeinen einem typischen Ablauf:

1. Setzen von Suchlinien, ggf. nach vorheriger Positionsbestimmung
2. Antasten von Kanten, entweder direkt aufgrund eines Kantenmodells oder zunächst binär mit nachfolgender Subpixelinterpolation
3. Berechnung von Ausgleichsgeometrien zur Bestimmung von Winkeln und Radien
4. Maßberechnung
5. Prüfungsbezogene Kalibrierung
6. Überprüfung von Toleranzen

Schwerpunktsabstände: Diese gehören zu den am einfachsten zu ermittelnden und robustesten Maßen; sie sind prinzipiell subpixelgenau, da der Schwerpunkt durch Mittelung über das gesamte Objekt berechnet wird.

Konturabstände: Abstände zwischen Konturen erfordern eine Spezifikation, zwischen welchen Punkten und ggf. in welcher Richtung zu messen ist. Man kann z. B. minimale und maximale Abstände in festgelegten Richtungen und nur für überlappende Konturbereiche messen.

Winkelmessung an Objektschwerpunkten: Bei der einfachsten Form der Winkelmessung werden Objektschwerpunkte mit Geraden verbunden, für die dann aus der Geradengleichung direkt der Winkel hervorgeht.

Winkelmessung an Konturen: Die Messung von Konturwinkeln erfordert die Approximation der Kontur durch eine Regressionsgerade; iterative Berechnungen können die Anpassung durch Vernachlässigung von Ausreißern verbessern.

Ausgleichsgeometrien: Konturen können durch verschiedene Geometrien, wie Geraden, Kreise und Ellipsen angenähert werden, um den Grad der Abweichung von diesen Idealformen zu bestimmen. Dazu können auch Einzelpunkte einer Kante benutzt werden.

Kantendetektion: Vermessungsaufgaben sind häufig an Kanten orientiert, daher empfiehlt sich oft eine direkte Auswertung von Kanteneigenschaften anstelle der

Binärsegmentierung mit anschließender Subpixelberechnung, da sich abhängig von der Binärschwelle der Startpunkt der Subpixelberechnung verändern kann.

Subpixelgenauigkeit: Messauflösungen besser als ein Bildpunkt lassen sich einerseits, wie bei der Schwerpunktsberechnung, durch Mittelung über eine Vielzahl von Bildpunkten erreichen, andererseits durch direkte Interpolation von Grauwertverläufen.

Beleuchtung: Hochgenaue Vermessung erfordert entsprechend stabile Beleuchtung. Hintergrundbeleuchtung erzeugt gute Kontraste und präzise Kanten, perspektivische Effekte lassen sich durch telezentrische Beleuchtungen ausschalten.

Bildaufnahme: Für Vermessungsaufgaben sollten quadratische Pixel und Digitalkameras verwendet werden, um Verzerrungen auszuschließen. Für hohe Genauigkeiten sollten telezentrische Objektive verwendet werden.

Kalibrierung: Die Bestimmung der Beziehung zwischen Maßen im Bild und in der realen Szene sollte unter denselben Bedingungen erfolgen, wie bei der endgültigen Anwendung, und möglichst direkt am Prüfteil, um dessen Einfluss auf die Bildeigenschaften zu erfassen. Wenn das nicht möglich ist, sind Referenzteile, die den Eigenschaften des Prüfteils nahe kommen, standardisierten Kalibriernormalen vorzuziehen.

8. Querschnitt: Bildaufnahme und Beleuchtung

In den vorangegangenen Kapiteln haben wir eine Reihe von Möglichkeiten kennengelernt, digitale Bilder zu verarbeiten, Objekte darin zu erkennen und zu bewerten. Bislang haben wir jedoch noch nichts darüber gesagt, wie man zu diesem digitalen Bild kommt, dessen Eigenschaften und Qualität von entscheidender Bedeutung für die Lösung einer Bildverarbeitungsaufgabe sind.

Digitale Bilder können von ganz verschiedenen Sensoren erzeugt werden. Der bei weitem häufigste Fall ist jedoch die Abbildung sichtbaren Lichtes durch ein Objektiv auf einen Kamerasensor, dessen Signal dann an das Bildverarbeitungssystem übertragen wird.

Alle Elemente dieser Kette, vom Objektiv über die eigentlichen lichtempfindlichen Elemente bis hin zur Übertragung in den Rechner, beeinflussen die Eigenschaften des digitalen Bildes und sind damit auch entscheidend für die Qualität und Reproduzierbarkeit der Ergebnisse, die sich erzielen lassen. Verluste, die hier entstehen, lassen sich softwaretechnisch kaum oder gar nicht ausgleichen. In allen Anwendungsbereichen der industriellen Bildverarbeitung ist man auf hochwertige Bilder angewiesen, am meisten aber bei der optischen Vermessung, da hier auch die höchsten Genauigkeitsanforderungen gestellt werden.

Wie die Beispiele der bisherigen Kapitel gezeigt haben, ist die Beleuchtung für die erfolgreiche Lösung einer Bildverarbeitungsaufgabe mindestens ebenso wichtig wie Optik und Aufnahmetechnik. Es ist naturgemäß sehr schwierig, Beleuchtungsempfehlungen zu geben. Vieles muss man hier in Experimenten mit den tatsächlichen Prüfteilen herausfinden, vieles ist auch abhängig von den Bedingungen, die am Ort der Installation herrschen: Platzverhältnisse, Teilebewegung, Fremdlichteinflüsse usw. Erfolgreiches Experimentieren setzt aber grundlegende Kenntnisse der Möglichkeiten und Techniken voraus, weshalb wir in Abschnitt 8.6 einen Überblick über typische Beleuchtungsmethoden und ihre Anwendung geben werden, um ein Gefühl dafür zu vermitteln, wie man an eine konkrete Beleuchtungsproblematik herangehen könnte.

8.1 Halbleitersensoren für die Bildverarbeitung

Wir beginnen unsere Besprechung der Aufnahmetechnik mit den heute überwiegend verwendeten Halbleitersensoren, gehen also zu Beginn dieses Kapitels davon aus, dass die optischen Elemente für die Entstehung eines einwandfreien Bildes auf dem Sensor gesorgt haben. Das ermöglicht uns, in der Besprechung der opti-

schen Grundlagen in 8.5 einige Gesetzmäßigkeiten direkt im Hinblick auf die Eigenschaften der verwendeten Sensoren zu untersuchen.

Einen kurzen Überblick über die verschiedenen Sensorarten und die allen Sensoren gleichen Eigenschaften werden im ersten Unterabschnitt behandelt. Darauf aufbauend werden in den folgenden Abschnitten die CCD- und die CMOS-Sensoren eingehender behandelt. Spezialsensoren bzw. -technologien sollen kurz im vierten Abschnitt vorgestellt werden. Bezogen sich alle vorangehenden Ausführungen auf monochrome Sensoren, soll den Farbsensoren und ihren Eigenschaften ein eigener Abschnitt gewidmet werden. Abschließend soll noch auf die allgemeinen elektro-optischen Eigenschaften, die für alle Sensoren gelten, eingegangen werden. Auf die physikalischen Zusammenhänge werden wir hier nicht eingehen, diese sind z. B. in (Theuwissen 1995) dargestellt.

8.1.1 Einführung

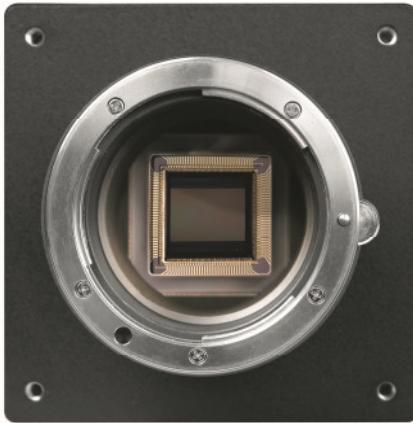
Halbleitersensoren prägen als die elektronischen Augen unsere heutigen Kameras. Sie wandeln die Intensität des einfallenden Lichtes in eine Spannung bzw. in einen Digitalwert, der je nach Anwendung weiterverarbeitet werden kann. Grundsätzlich unterscheidet man die Bildsensoren hinsichtlich

- der geometrischen Anordnung der einzelnen lichtempfindlichen Elemente (Pixel) in Zeilen- oder Matrixsensoren
- der Funktionsweise und Fertigungstechnologie in Charge Coupled Device (abgekürzt CCD) und Complementary Metal Oxide Semiconductor (abgekürzt CMOS) Sensoren

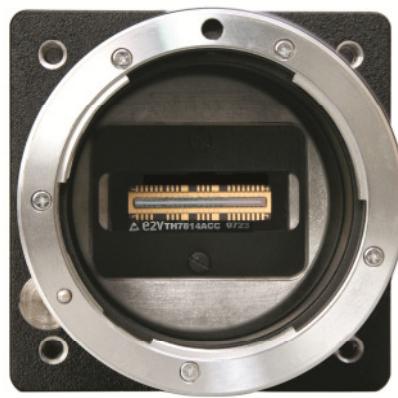
Matrixsensoren: Bei Matrixsensoren wird, wie bei einer handelsüblichen Digitalkamera für den Consumer-Markt, ein Bild aus mehreren Zeilen und Spalten während einer einzigen Belichtungszeit aufgenommen. Dies ist die mit Abstand am häufigsten in der industriellen Bildverarbeitung eingesetzte Ausführung. Abbildung 8.1 zeigt auf der linken Seite eine Kamera mit Matrixsensor.

Zeilensensoren: Zeilensensoren nehmen immer nur eine einzige Zeile auf. In vielen Zeilenkameraanwendungen werden mehrere nacheinander aufgenommene Zeilen zu einem rechteckigen Bild zusammengesetzt. Abbildung 8.1 zeigt auf der rechten Seite eine Kamera mit Zeilensor. Auf die Besonderheiten beim Einsatz von Zeilenkameras gehen wir in Abschnitt 8.4 ein.

CCD-Sensoren: In der industriellen Bildverarbeitung dominieren heute nach wie vor CCD-Sensoren, deren erste Exemplare Anfang der siebziger Jahre aus analogen Laufzeitgliedern, sogenannten Eimerketten-Schaltungen, entstanden sind und schnell die Bildaufnahmeröhren verdrängten. Mit dem Einzug in den Consumer-Markt durch die Verwendung in Camcordern hatten die CCD-Sensoren ein attraktives Preisniveau erreicht, das ihren Einsatz in der industriellen Bildverarbeitung beschleunigte.



Matrixsensor



Zeilensensor

Abbildung 8.1: Matrix- und Zeilensensoren (mit freundlicher Genehmigung der Firma Basler AG (Ahrensburg))

CMOS-Sensoren: Die Forschung auf dem Gebiet der CMOS-Sensoren erhielt Anfang der neunziger Jahre richtig Schwung. 1993 stellte das Jet Propulsion Laboratory (Pasadena, CA, USA) der US-amerikanischen Weltraumbehörde NASA einen Sensor mit einer Auflösung von 128×128 Pixeln vor, dem bereits drei Jahre später der erste Ein-Megapixel-Sensor folgte. Die Herstellung von CMOS-Chips und -Sensoren basiert im Gegensatz zu der von CCD-Sensoren auf einer verbreiteten Standardtechnologie. Damit ist die Produktion von CMOS-Sensoren prinzipiell preisgünstiger. Für die Eigenschaften der Sensoren wesentlich bedeutsamer sind aber die Unterschiede, wie die Bildinformationen aus ihnen ausgelesen werden.

Im Grunde arbeiten alle Halbleitersensoren nach einem gleichen Funktionsprinzip:

1. Das einfallende Licht in Form eines Photonenstromes generiert in dem lichtempfindlichen Teil des Halbleiterkristalls freie Elektronen.
2. Die generierten Elektronen werden über eine bestimmte Zeit, der sogenannten Belichtungs- bzw. Integrationszeit gesammelt.
3. Nach Ablauf der Belichtungszeit wird bei CCD-Sensoren die gesammelte Ladungsmenge Pixel für Pixel zur Wandlungsstufe transportiert.
4. Abschließend wird die gesammelte Ladungsmenge in eine proportionale Spannung gewandelt, die von einem Analog-Digital-Wandler in einen numerischen Wert umgesetzt werden kann.

CCD- und CMOS-Sensoren wandeln in Photodioden die eintreffenden Photonen in Ladungen um und sammeln diese in einem sogenannten Potentialtopf. Dort werden die Ladungen für die Dauer der Belichtungszeit gespeichert. Die Speicherkapazität hängt in erster Linie von der geometrischen Pixelgröße und dem Aufbau der Halbleiterstrukturen ab. Sie wird als *Full Well Capacity* des Pixels in

Elektronen (e^-) angegeben. Wichtig ist die für alle Sensoren geltende Daumenregel: je größer das Pixel, desto größer die Full Well Capacity, und damit der erreichbare Dynamik- bzw. Kontrastumfang.

Ein grundlegender Unterschied zwischen CCD- und CMOS-Sensoren besteht im Auslesen der Ladungen der einzelnen Pixel. Dies hat erhebliche Auswirkungen auf die Eigenschaften der Sensoren zur Folge und soll deshalb in den folgenden Abschnitten dargestellt werden.

8.1.2 CCD-Sensoren

Ein Halbleitersensor besteht aus lichtempfindlichen Fotodioden, die in einer Matrix oder einer Linie im Fall eines Zeilensensors angeordneten sind. Sobald Licht auf sie fällt, erzeugen sie Ladungsträger. Da Digitalkameras meistens keine mechanische Blende haben, sind die Fotodioden permanent Licht ausgesetzt. Um brauchbare Bilder zu erhalten, muss die sogenannte Belichtungszeit sowohl einen definierten Anfang als auch ein definiertes Ende haben. Es sind also Mechanismen nötig um die Belichtungszeit, also die Zeit in der das eintreffende Licht zum aufgenommen Bild beiträgt, zu begrenzen. Da das während der Belichtungszeit auf den Sensor auftreffende Licht gesammelt wird, spricht man auch von Integrationszeit.

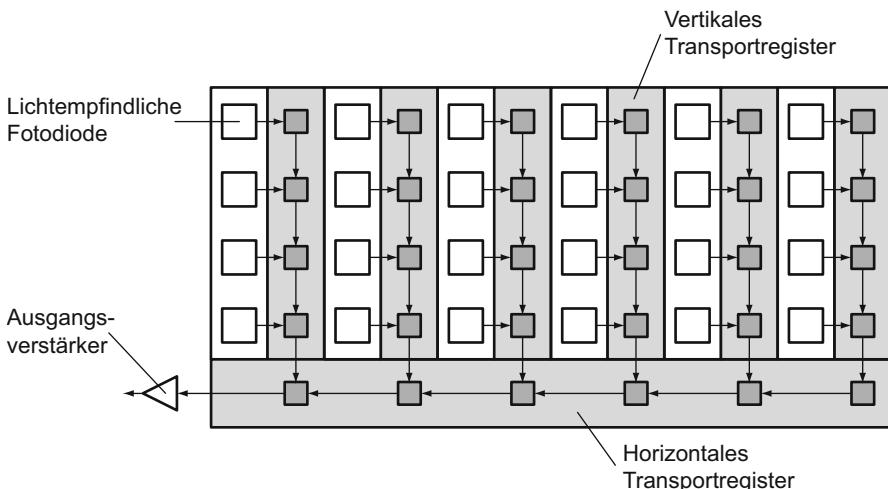


Abbildung 8.2: Aufbau eines Interline-CCD-Sensors

Die Belichtungszeit beginnt mit dem gleichzeitigen Löschen der Ladungen in den Fotodioden aller Pixel. Ab diesem Zeitpunkt werden alle erzeugten Ladungsträger gesammelt. Zwischen den Fotodiodenspalten befinden sich die sogenannten vertikalen Schiebe- oder Transportregister, wie in Abbildung 8.2 schematisch abgebildet. Aufgrund dieser Register wird diese Bauart als Interline-Transfer-Sensor be-

zeichnet. Am Ende der Belichtungszeit werden die Ladungen, die die Bildinformation repräsentieren, aus den Fotodioden aller Pixel gleichzeitig in die Transportregister verschoben. Da die Transportregister durch eine dünne Metallfolie vor dem Lichteinfall weitgehend geschützt sind, werden in ihnen keine weiteren Ladungsträger mehr erzeugt.

In den Transportregistern können die Ladungen pixelweise verschoben werden. Die dabei verwendete Methode hat dem Sensortyp seinen Namen Charge Coupled Device gegeben. Aus den vertikalen Transportregistern werden die Ladungen Zeile für Zeile in das horizontale Transportregister verschoben, aus dem dann Pixel für Pixel ausgelesen wird. Auf diese Weise wird die im Sensor aufgenommene Bildinformation als eine Folge von Ladungen der einzelnen Pixel ausgelesen.

Am Ausgang des horizontalen Transportregisters wandelt der als Ausgangsverstärker bezeichnete Ladungsverstärker die Ladungspakete der einzelnen Pixel in eine Spannung um, die anschließend digitalisiert wird. Bei der Analog-Digital-Wandlung des Signals nutzt man häufig die Technik des Correlated Double Sampling (abgekürzt CDS). Dabei werden nacheinander der Schwarzwert und danach der Weißwert des Signals abgetastet. Durch Subtraktion beider Werte im Analog-Digital-Wandler (abgekürzt A/D-Wandler) wird ein der Helligkeit proportionaler Wert ausgegeben. Der Vorteil dieses Verfahrens liegt z. B. in der Unterdrückung von niederfrequenten Störungen durch die Betriebsspannung des Ausgangsverstärkers.

Da die Integration und der Transport der Ladungen in getrennten Bereichen erfolgt, ist es möglich, während die Ladungen eines Bilds ausgelesen werden, zeitgleich das nächste Bild zu belichten. Dies bezeichnet man als Overlapped-Modus, mit dem natürlich die Bilder schneller hintereinander aufgenommen werden können, als wenn die Belichtungszeit erst nach dem vollständigen Auslesen des vorherigen Bilds beginnen kann, was in Abbildung 8.3 schematisch dargestellt ist.

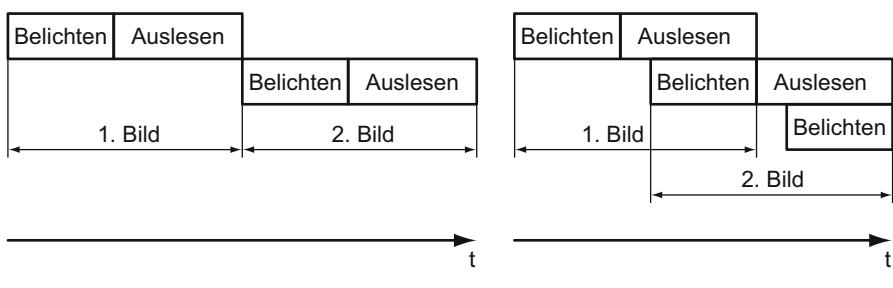


Abbildung 8.3: Betrieb eines Sensors im sequentiellen Modus und im Overlapped-Modus

Neben der Interline-Transfer-Architektur gibt es auch noch andere Typen wie beispielsweise sogenannte Full-Frame-Sensoren und Frame-Transfer-Sensoren. Diese Architekturen verzichten auf die vertikalen Transportregister, weshalb sie eine höhere Empfindlichkeit haben. Allerdings können mit solchen Sensoren ohne me-

chanische Blende nur ruhende Objekte aufgenommen werden, so dass sie nur selten in der industriellen Bildverarbeitung eingesetzt werden.

Bis jetzt sind wir davon ausgegangen, dass das komplette Bild auf einmal ausgelesen wird. Dieses Verfahren wird als Progressive Scan bezeichnet. Daneben existiert das jahrelang sehr verbreitete Verfahren des Interlaced Scanning. Dieses Verfahren kommt aus der analogen Fernsehtechnik bei der man mit sogenannten Halbbildern gearbeitet hat. Dabei werden im Wechsel alle geraden Zeilen und alle ungeraden Zeilen ausgelesen. Ein Nachteil dieses Verfahrens ist die durch die getrennte Belichtung der Halbbilder stark verringerte Bildrate. Ein anderer Nachteil sind die Verzerrungen, die bei bewegten Aufnahmeszenen entstehen. In der industriellen Bildverarbeitung werden interlaced Kameras daher aktuell kaum noch eingesetzt.

Der Geschwindigkeit, mit der die Ladungen eines Sensors verschoben und ausgelesen werden können, sind physikalische Grenzen gesetzt. Dies begrenzt die sogenannte Bildrate, also die Anzahl der Bilder pro Sekunde. Typische Werte für einen Auslesekanal sind 40–60 MHz, also 40–60 Millionen Pixel pro Sekunde. Um höhere Bildraten, insbesondere bei Sensoren mit großer Auflösung, erreichen zu können, wurden Architekturen mit mehreren Ausgängen, sogenannten Taps, entwickelt. Über die einzelnen Ausgangsverstärker dieser als Multi-Tap-Sensoren bezeichneten Sensoren, werden parallel die Ladungen von jeweils einem Teil der Pixel des Sensors ausgelesen. Dadurch kann man beispielsweise einen Sensor mit vier Taps viermal so schnell auslesen, wie einen Sensor mit nur einem Ausgang. Bei dem in Abbildung 8.4 schematisch dargestellten Sensor mit vier Taps werden die Ladungen der Pixel im oberen, linken Viertel des Sensors zum Ausgangsverstärker oben links transportiert, die Ladungen der Pixel oben, rechts in den oberen, rechten Ausgangsverstärker transportiert usw.

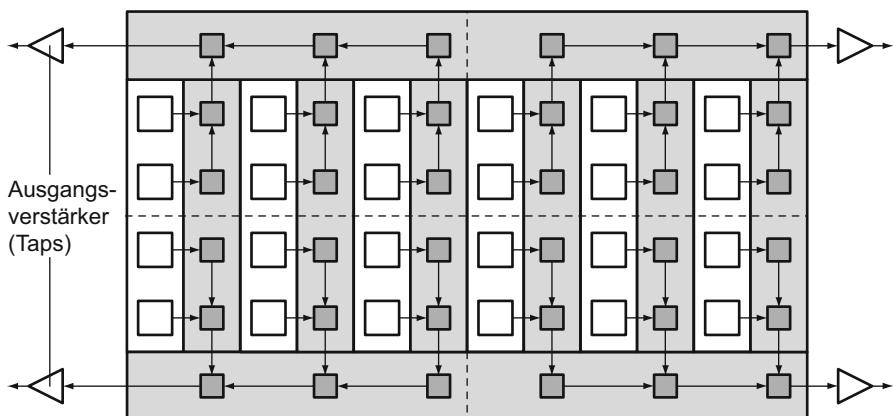


Abbildung 8.4: Prinzip eines 4-Tap-Sensors

8.1.3 CMOS-Sensoren

Der zentrale Unterschied zwischen CMOS- und CCD-Sensoren besteht darin, dass CCD-Sensoren, außer bei Multi-Tap-Sensoren, nur einen Ladungsverstärker für den gesamten Sensor haben, während bei CMOS-Sensoren jedes Pixel einen eigenen Ladungsverstärker hat. Diese wandeln bereits im Pixel die während der Belichtungszeit erzeugten Ladungen in Spannung um. Aufgrund dieser Umwandlung werden CMOS-Sensoren auch als Active Pixel Sensoren bezeichnet werden. Die Helligkeitsinformationen der einzelnen Pixel werden als Spannungen und nicht, wie bei CCD-Sensoren, als Ladungen ausgelesen. Die Schaltung für das Auslesen der einzelnen Pixel-Spannungen ist schematisch in Abbildung 8.5 dargestellt. Von der vertikalen Steuerung werden die sogenannten Auswahltransistoren der Pixel einer Zeile geschaltet, wodurch die Bildzeile zum Auslesen ausgewählt wird. Jeder geschaltete Auswahltransistor legt die Spannung seines Pixels auf die zugehörige sogenannte Column Line. Die horizontale Steuerung schaltet dann eine Column Line nach der anderen auf den Ausgangsverstärker des Sensors, der dadurch die Spannungen der Pixel nacheinander ausliest.

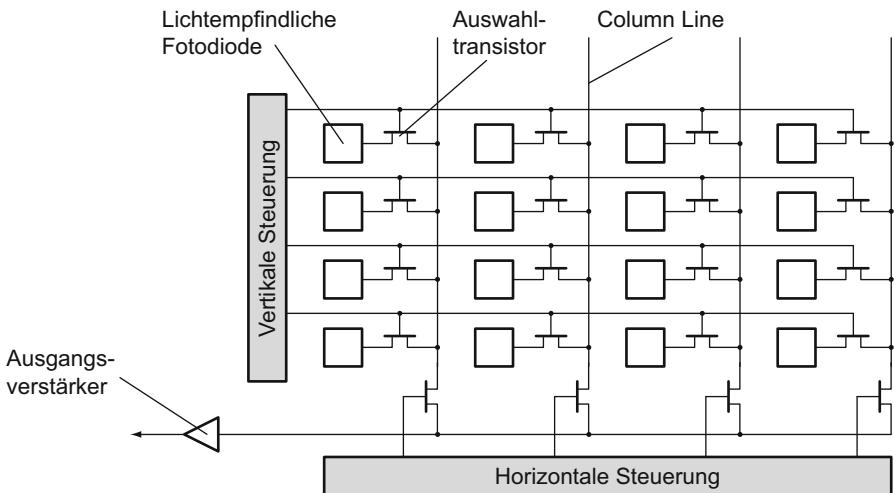


Abbildung 8.5: Aufbau eines CMOS-Sensors

Wie gesagt hat jeder Pixel für die Umwandlung der gesammelten Ladungen in eine Spannung einen eigenen Ladungsverstärker. Diese Ladungsverstärker unterscheiden sich untereinander hinsichtlich ihrer Verstärkung und ihres Offsets, so dass die Ausgangsspannungen der einzelnen Pixeln bei gleicher Ladung häufig variieren. Diese Unterschiede überlagern das aufgenommene Bild als ein ortsfestes Rauschen. Dies wird als Fixed Pattern Noise bezeichnet und ist typisch für CMOS-Sensoren. Durch das bei CCD-Sensoren bereits erwähnte Correlated Double Sampling lässt sich der Fixed Pattern Noise reduzieren. Dies muss allerdings in jedem Pixel realisiert werden, so dass der Umfang der Schaltung auf Kos-

ten der lichtempfindlichen Fläche anwächst. Dies hat gemäß oben beschriebener Regel eine geringe Empfindlichkeit des CMOS-Sensors zur Folge.

Außer dem Correlated Double Sampling lässt sich die Schaltung in den einzelnen Pixeln noch um andere Funktionen erweitern. Die Analog-Digital-Wandlung wird z. B. häufig bereits auf dem Chip durchgeführt, entweder mit einem Analog-Digital-Wandler für den gesamten Sensor oder mit einem für jede Spalte. Daher haben CMOS-Sensoren oft ein ausschließlich digitales Interface. Dies erlaubt nachfolgenden Komponenten die Ausgangsdaten direkt auszuwerten. Weitere typische Funktionen sind die integrierte Belichtungszeitsteuerung sowie das Auslesen von Teilbereichen, die wie die Suchbereiche in einem Bild als Region of Interest bezeichnet werden. Wenn der Sensor zudem über Funktionen zur Vorverarbeitung oder Farbumwandlung verfügt, spricht man von einem System-on-Chip.

Belichtungssteuerung: Wie bei CCD-Sensoren werden auch bei CMOS-Sensoren Ladungen erzeugt sobald Licht auf die lichtempfindlichen Flächen der Pixel trifft. Von daher ist auch bei CMOS-Sensoren eine Belichtungssteuerung erforderlich, die den Beginn und das Ende der Belichtungszeit definiert. Dabei sind zwei verschiedene Verfahren gebräuchlich:

Rolling Shutter: Das häufigste Verfahren ist der sogenannte Rolling Shutter. Wie bei CCD-Sensoren beginnt die Belichtungszeit damit, dass die bereits erzeugten Ladungen gelöscht werden. Beim Rolling Shutter erfolgt dies nicht gleichzeitig für den gesamten Sensor, sondern für eine Zeile nach der anderen. Parallel dazu werden die Zeilen nacheinander ausgelesen, womit die Belichtungszeit endet. Der zeitliche Versatz, der zwischen dem Beginn der Belichtungszeit zweier benachbarter Zeilen liegt, entspricht genau der Zeit, die für das Auslesen einer Zeile erforderlich ist.

Dieses Verfahren erlaubt es einfache und günstige Sensoren mit einheitlicher Belichtungszeit für alle Pixel herzustellen. Deshalb wird diese Art von Sensoren häufig in Multimedia-Anwendungen eingesetzt. Durch die versetzten Belichtungszeiten der einzelnen Zeilen ergeben sich aber trapezförmige Verzerrungen bei bewegten Objekten. Daher sind diese Sensoren in der industriellen Bildverarbeitung nur für die Aufnahme von ruhenden Objekten geeignet.

Global Shutter: Im Gegensatz zu Rolling-Shutter-Sensoren verfügen Sensoren mit Global Shutter über eine Ladungsspeicherzelle im Pixel. Damit können die während der Belichtungszeit erzeugten Ladungen bis zum Auslesen zwischengespeichert werden. Dadurch ist es möglich, die Belichtungszeit für alle Pixel des Sensors gleichzeitig beginnen und enden zu lassen. Der Preis dafür ist ein höherer Schaltungsaufwand, denn der Zwischenspeicher erfordert bis zu acht zusätzliche Transistoren pro Pixel, was zu einer kleineren lichtempfindlichen Fläche und damit geringerer Empfindlichkeit führt.

Zur Zeit gibt es nur wenige CMOS-Sensoren mit Global Shutter auf dem Markt. Meistens handelt es sich dabei um Sensoren die speziell für die industrielle Bildverarbeitung gefertigt werden. Aufgrund der geringen Stückzahlen sind sie

erheblich teurer als CMOS-Sensoren mit Rolling Shutter, die in großen Stückzahlen für Multimedia-Anwendungen hergestellt werden. Gegenüber CCD-Sensoren haben CMOS-Sensoren mit Global Shutter deshalb keinen Preisvorteil, aber den Nachteil, dass die Inhomogenitäten in den mit ihnen aufgenommenen Bildern aufwändig korrigiert werden müssen.

Bei CMOS-Sensoren ist die Auslesegeschwindigkeit von der Anzahl der Auslesekanäle abhängig. Genau wie man bei CCD-Sensoren durch die Multi-Tap-Architektur die Auslesegeschwindigkeit steigern kann, kann man bei CMOS-Sensoren das Bildfeld in mehrere Bereiche aufteilen, die jeweils über einen eigenen Auslesekanal verfügen. Die Teilung in Bereiche erfolgt bei CMOS-Sensoren immer so, dass mehrere Spalten zusammengefasst werden. Für diese Architektur ist es typisch, dass sich die Auslesegeschwindigkeit steigert, wenn weniger Zeilen ausgelesen werden, die Auslesegeschwindigkeit aber konstant bleibt, wenn weniger Spalten ausgelesen werden.

8.1.4 Sonderbauformen

Wie bereits angesprochen kann man bei CMOS-Sensoren zusätzliche elektronische Funktionalität einfach und kostengünstig in jedem Pixel integrieren, wenn auch auf Kosten der lichtempfindlichen Fläche. Damit ist es möglich, für spezielle Anwendungen Sonderbauformen zu implementieren. Ein Beispiel für eine derartige Sonderbauform sind sogenannte Time-of-Flight-Sensoren (abgekürzt TOF-Sensoren), die neben einem Helligkeitsbild zusätzlich ein Entfernungsbild aufnehmen, wobei die Entfernungsmessung auf der Ausbreitungszeit des Lichtes, eben der „Time of flight“ basiert. Damit sind sie auch ein Beispiel für einen 3D-Sensor.

Für die Messung der Lichtlaufzeit ist eine Lichtquelle erforderlich, die ein mit einer Sinuskurve moduliertes Licht aussendet, das mit dem Sensor synchronisierte ist. Das permanente Zu- und Abnehmen der Helligkeit der Lichtquelle kann man natürlich auch im von einem Gegenstand reflektierten Licht beobachten, das auf den Sensor zurückfällt. Die Laufzeit des Lichtes kann man als Differenz zwischen den Phasen von ausgesendetem und empfangenen Licht beobachten. Als Empfänger arbeitet der Sensor synchron zur Modulation des gesendeten Lichtes. Während einer Viertelperiode werden die Ladungen in einer Speicherzelle gespeichert, danach wird die nächste Speicherzelle gefüllt. Da eine Periode zu kurz ist, um genügend Ladungsträger zu erzeugen, werden in den vier Speicherzellen die Ladungen mehrerer Perioden gesammelt. Nach dem Ende der Integrationszeit lässt sich aus den in den Speicherzellen gesammelten Ladungen Folgendes berechnen:

- Mittelwert, als Maß für die Helligkeit des Objekts, wie bei einem normalen Kamerasensor
- Phasendifferenz, als Maß für die Entfernung zum Objekt
- Amplitude, als Maß für die Genauigkeit der Entfernungsmessung

Aufgrund der umfangreichen zusätzlichen Elektronik nimmt ein Pixel ein Vielfaches der Fläche von Pixeln normaler CMOS-Sensoren ein. Damit sind die Auflösungen von TOF-Sensoren wesentlich geringer als die von normalen Kamerasensoren. Stand der Technik sind heute Auflösungen von 200×200 . Damit sind die Einsatzmöglichkeiten von TOF-Sensoren noch sehr beschränkt, dennoch ist diese Technologie ein interessantes Beispiel für Sonderentwicklungen auf der Basis von CMOS-Sensoren.

Neben den TOF-Sensoren gibt es eine Reihe von Sensorsystemen die mittels Lichtschnitt, Stereoskopie und anderer Verfahren ebenfalls 3D-Informationen ermitteln. Für die meisten von ihnen gilt, dass sie wie TOF-Sensoren bisher nur für sehr begrenzte Aufgabenfelder einsetzbar sind. In diesem Markt ist aber eine große Dynamik zu beobachten, so dass sich dort in den nächsten Jahren einige Technologien in der industriellen Bildverarbeitung etablieren dürften.

8.1.5 Farbsensoren

Während die Helligkeit von Grauwertbildern als physikalische Größe der Intensität beschrieben werden kann und deshalb auch problemlos im Infraroten oder Ultravioletten angewandt werden kann, ist Farbe unlösbar mit der menschlichen Wahrnehmung verbunden. Außerhalb des vom menschlichen Auge wahrnehmbaren Wellenlängenbereich von 380 nm bis 780 nm sind keine Farben definiert. Auch die in der industriellen Bildverarbeitung häufige Darstellung von Farbe mit drei sogenannten Kanälen ist in Anlehnung an das menschliche Auge entstanden. Die farbempfindlichen Photorezeptoren, die sogenannten Zapfen, kommen beim Menschen in drei verschiedenen Arten vor, die jeweils ihre eigene spektrale Empfindlichkeit haben, so dass jede Zapfenart auf Licht gleicher Wellenlänge verschieden stark reagiert. Nach dem Wellenlängenbereich, in dem die verschiedenen Arten besonders kräftig reagieren, bezeichnet man sie als:

- L-Zapfen, für *long*, also langwelliges, rotes Licht
- M-Zapfen, für *middle*, das mittlere Spektrum
- S-Zapfen, für *short*, das kurzwellige, blaue Licht

Wie man in Abbildung 8.6 sieht, ist die Empfindlichkeit der L- und M-Zapfen sehr ähnlich, während sich die der S-Zapfen davon deutlich unterscheidet.

Wie bereits erwähnt werden Farben in der industriellen Bildverarbeitung üblicherweise durch drei Kanäle beschrieben. Dabei repräsentiert jeder Kanal eine Grundfarbe, auch Primärvalenz genannt. Durch die gewichtete Mischung der drei Primärvalenzen können sämtliche durch den Sensor erfassbaren Farben beschrieben werden.

Da ein einzelnes Pixel nur einen Helligkeitswert aufnehmen kann, sind damit Farbmessungen nicht ohne weiteres möglich. Um Farbinformation zu gewinnen, muss das einfallende Licht spektral zerlegt und auf unterschiedliche Sensorelemente geleitet werden. Dafür gibt es eine Reihe von Verfahren:

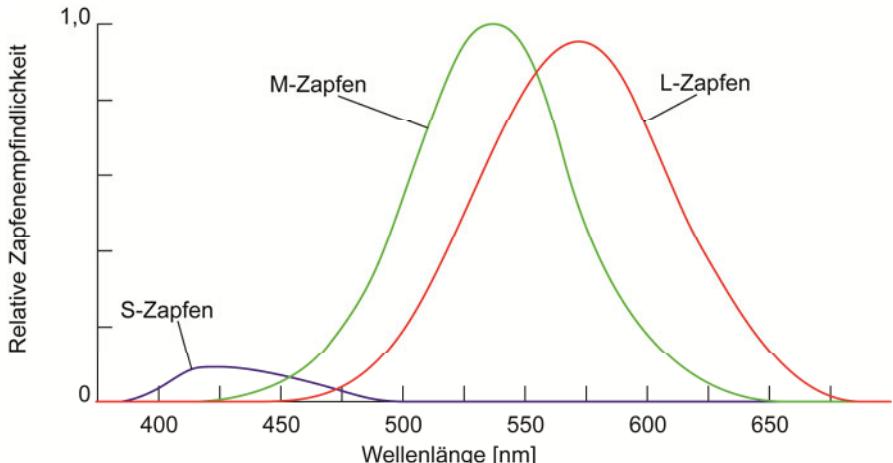


Abbildung 8.6: Spektrale Empfindlichkeit der Zapfen des menschlichen Auges

Drei-Chip-Farbkameras: Bei 3-Chip-Farbkameras wird das Licht auf dem Weg zwischen Objektiv und Kamerasensor durch einen Strahlteiler in eine rote, grüne und blaue Komponente geteilt. Jede Komponente fällt auf einen anderen Sensor-Chip, auf dem dann entsprechend das rote, grüne oder blaue Bild entsteht. Damit stehen für jedes Pixel alle drei Farbwerte zur Verfügung, so dass die Farbauflösung hervorragend ist. Ein gravierender Nachteil dieser Kameras ist, dass sie aufgrund der Strahlteilung und der erforderlichen drei Sensoren mit Ausleseelektronik relativ groß, schwer und teuer sind. Ein weiteres Problem ist es sicherzustellen, dass die Farbkanäle deckungsgleich auf den drei Sensoren liegen. Da der Einfallswinkel des Lichts auf den Strahlteiler die Filtercharakteristik beeinflusst, können derartige Kameras nur für kleine Chips mit geringer Auflösung hergestellt werden.

Ein-Chip-Farbkameras mit Farbfiltermaske: Technisch wesentlich einfacher ist es nur einen Chip für die Farbbildaufnahme zu verwenden. Dabei wird eine sogenannten Farbfiltermaske (englisch Color Filter Array, abgekürzt CFA) direkt auf dem Sensor aufgebracht. Das CFA sorgt dafür, dass ein Pixel nur die Helligkeit eines Farbkanals empfängt. Da die benachbarten Pixel andere Farbkanäle erfassen, kann die vollständige Farbinformation aus den umliegenden Pixeln interpoliert werden. Es gibt eine Reihe verschiedener Farbfiltermasken, von denen Abbildung 8.7 eine Auswahl zeigt. Die mit Abstand häufigste Maske ist das sogenannte Bayer-Pattern¹, das ganz links in Abbildung 8.7 dargestellt ist. Ein-Chip-Farbkameras haben den Nachteil, dass sie eine geringere Ortsauflösung haben, da die Farbwerte aus benachbarten Pixeln interpoliert werden müssen. Ein weiterer Nachteil ist die reduzierte Empfindlichkeit, da die Transmissionsfarbfilter stets auch einen Teil

¹ Benannt nach Bryce E. Bayer der diese Anordnung 1976 bei Eastman Kodak Company (Rochester, NY, USA) entwickelte.

des Lichts aus dem Wellenlängenbereich herausfiltern, für den sie an sich durchlässig sind.

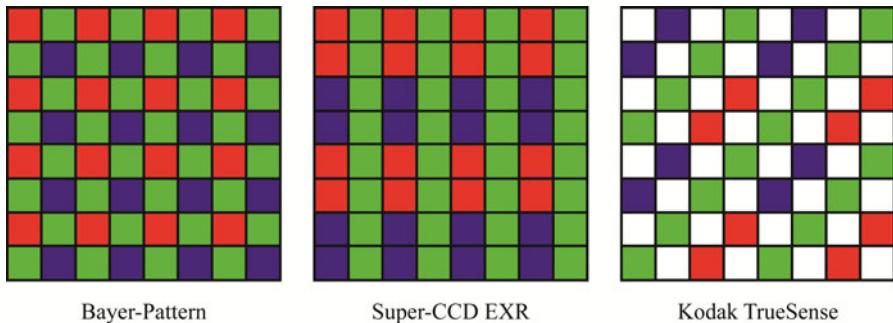


Abbildung 8.7: Farbfiltermasken

Dreilagenfarbsensor: Eine dritte Möglichkeit sind die Dreilagenfarbsensoren, wie sie von der Firma Foveon Inc. (Santa Clara, CA, USA) entwickelt wurden. Bei diesen Sensoren sind drei Pixel übereinander angeordnet, wobei in jeder Pixellage das Licht eines anderen Wellenlängenbereichs absorbiert und in Ladungen umgewandelt wird. Das andersfarbige Licht gelangt in die nächste Pixellage und wird dort absorbiert und in Ladungen umgewandelt oder weitergeleitet. Bisher konnte sich diese Technologie allerdings nicht auf dem Markt durchsetzen.

Gamut: Alle Farben, die ein Farbsensor aufnimmt, werden durch gewichtete Addition der Primärvalenzen beschrieben. Die Farben der Primärvalenzen ergeben sich dabei aus der spektralen Empfindlichkeit des Siliziums und der Farben der Farbfiltermatrix bzw. der Charakteristik des Strahlteilers. Aufgrund dieser technischen Einschränkungen ist es nicht möglich mit Farbsensoren alle Farben zu erfassen, die ein Mensch sehen kann. Den Ausschnitt des Farbraums, den ein Farbsensor erfasst, wird als Gamut bezeichnet. Verschiedene Kameras können unterschiedliche Primärvalenzen haben, so dass sich auch ihr Gamut unterscheidet.

8.1.6 Eigenschaften von Sensoren

Pixelgröße: Je größer die Pixel sind, desto größer ist ihre lichtempfindliche Fläche und damit ihre Empfindlichkeit. Außerdem können größere Pixel mehr Ladungen speichern, was zu einem höheren Dynamikumfang führt. In Datenblättern wird diese wichtige Sensoreigenschaft oft als Pixel Pitch, also als Abstand zwischen benachbarten Pixeln angegeben. Typische Pixelgrößen für den Einsatz in der industriellen Bildverarbeitung sind vier bis sieben Mikrometer, da kleinere Pixelgrößen durch die Beugungsbegrenzung der optischen Abbildung im Allgemeinen nicht sinnvoll sind. Ausnahmen sind hier Farb-CMOS-Sensoren für den Con-

sumer-Bereich mit bis zu zwei Mikrometer kleinen Pixeln. Bei ihnen werden für die Berechnung der RGB-Werte eines Pixels die benachbarten Pixel mit einbezogen, wodurch die Ortsauflösung ebenfalls bei mindestens vier Mikrometer liegt.

Auflösung: Bei Matrixsensoren gibt die Auflösung die Anzahl der Pixel in horizontaler und vertikaler Richtung an oder die Gesamtzahl der Pixel auf dem Chip in Megapixel (1.000.000 Pixel). Bei Flächensensoren beträgt das Seitenverhältnis häufig 4 : 3, allerdings gibt es auch viele andere Seitenverhältnisse bis hin zu quadratischen. Übliche Auflösungen in der industriellen Bildverarbeitung liegen zwischen 640×480 , was 0,3 Megapixel entspricht, und 2448×2050 , was 5 Megapixel entspricht. Für spezielle Anwendungen werden auch Sensoren mit 16 Megapixel und mehr verwendet.

Bei Zeilensensoren gibt die Auflösung die Anzahl der Pixel pro Zeile an. Da Zeilenkameras immer nur eine Zeile auf einmal aufnehmen und diese Zeilen dann zu Bildern zusammengesetzt werden, kann man bei Zeilenkameras keine Zeilenanzahl angeben. Typische Auflösungen von aktuell angebotenen Zeilenkameras sind 1024 bis 8192 Pixel.

Optisches Format: Das optische Format eines Sensors gibt seine Bilddiagonale an. Dieser Wert ist wichtig bei der Auswahl eines geeigneten Objektivs, da der Bildkreis mindestens so groß wie die Bilddiagonale des Sensors sein muss. Das optische Format wird üblicherweise in Inch angegeben, wobei einem Inch dabei 16 mm entsprechen! Ursprünglich bezeichnete dieses Maß den Außendurchmesser des Glaskolbens einer Fernsehaufnahmeröhre. Bei einem Glaskolbendurchmesser von 25,4 mm, also einem Inch, hatte der Aufnahmebereich eine Diagonale von 16 mm. Die Röhren sind vom Markt verschwunden, die 16 mm für einen Inch sind geblieben. Typische optische Formate sind in Tabelle 8.1 zusammengestellt.

Tabelle 8.1 Typische optische Formate von Halbleitersensoren

Optisches Format	Breite [mm]	Höhe [mm]	Diagonale [mm]
1/1"	12,8	9,6	16,0
2/3"	8,8	6,6	11,0
1/2"	6,4	4,8	8,0
1/3"	4,8	3,6	6,0

Quanteneffizienz: Sowohl CCD- als auch CMOS-Sensoren beruhen auf dem Prinzip, dass die auf ein Pixel treffenden Photonen Ladungen erzeugen. Allerdings ist es keineswegs so, dass jedes Photon auch einen Ladungsträger erzeugt. Zum einen treffen einige Photonen auf die lichtunempfindlichen Bereiche des Pixels, des weiteren werden aus physikalischen Gründen nie alle Photonen, die auf den lichtempfindlichen Bereich treffen, in Elektronen umgewandelt. Die sogenannte Quanteneffizienz beschreibt wie gut dieser Umwandlungsprozess insgesamt funktioniert, indem sie das Verhältnis zwischen erzeugten Elektronen und aufgetroffenen Photonen angibt:

$$\mu = \frac{n_{\text{Elektronen}}}{n_{\text{Photonen}}} \quad (8.1)$$

Füllfaktor: Ein Pixel besteht aus einem lichtempfindlichen und einem lichtunempfindlichen Bereich. Das Verhältnis der lichtempfindlichen Fläche zur Gesamtfläche wird als Füllfaktor bezeichnet. Der Füllfaktor geht natürlich auch in die Quanteneffizienz mit ein.

Mikrolinsen: Um bei Halbleitersensoren mit schlechtem Füllfaktor die Quanteneffizienz zu verbessern, setzt man häufig Mikrolinsen ein. Die Mikrolinsen werden als Matrix auf dem Sensor-Chip aufgebracht, so dass zu jedem Pixel eine Mikrolinse gehört, wie in Abbildung 8.8 zu sehen. Diese fokussiert das einfallende Licht auf die lichtempfindliche Fläche des Pixels. Damit kann Licht, das ohne die Linse auf lichtunempfindliche Bereiche gefallen wäre, auf die lichtempfindliche Fläche umgelenkt werden.

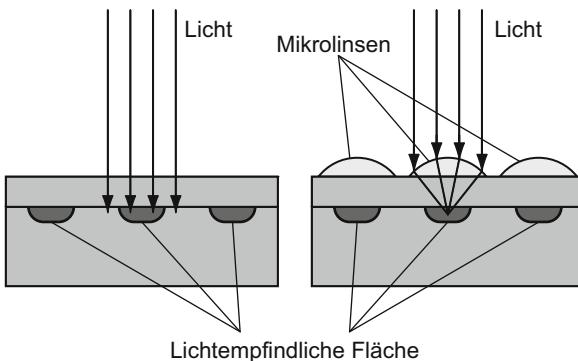


Abbildung 8.8: Mikrolinsen und ihre Wirkung

Spektrale Empfindlichkeit: Die Quanteneffizienz von Halbleitersensoren ist stark von der Wellenlänge des einfallenden Lichtes abhängig, wie in Abbildung 8.9 zu sehen ist. Dabei wird der Bereich, in dem Halbsensoren empfindlich sind, durch zwei Effekte begrenzt:

Langwelliges Licht: Je langwelliger das einfallende Licht ist, desto weiter dringt ein Photon in das Sensormaterial ein, bevor es absorbiert wird und ein Ladungsträger erzeugt wird. Damit sinkt die Wahrscheinlichkeit, dass die Ladung innerhalb der Photodiode entsteht und gespeichert werden kann mit zunehmender Wellenlänge. Bei Silizium, aus dem die meisten Halbleitersensoren bestehen, müssen Photonen eine kleinere Wellenlänge als 1125 nm haben, damit sie Elektronen freisetzen können. Für langwelligeres Licht sind Sensoren auf Silizium-Basis völlig unempfindlich.

Kurzwelliges Licht: Bei kurzwelligem Licht ist es genau umgekehrt. Kurzwelliges Licht wird sehr nahe der Sensoroberfläche absorbiert und erzeugt damit dort Ladungsträger, die deshalb häufig zu weit vom Ladungsspeicher entfernt sind, so dass sie nicht gesammelt werden können. Damit fällt die Empfindlichkeit zum kurzweligen Bereich des Spektrums ebenfalls ab.

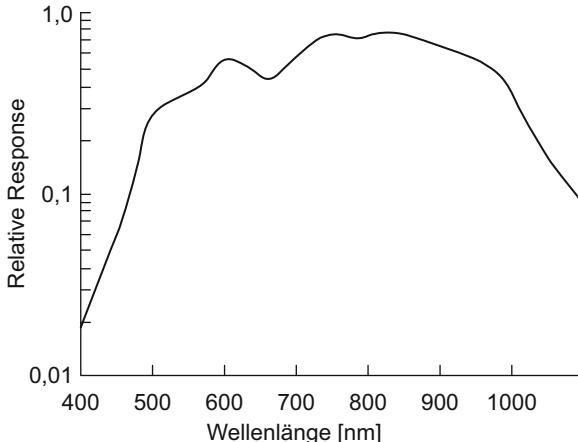


Abbildung 8.9: Spektrale Empfindlichkeit eines CCD-Sensors

Um die spektrale Empfindlichkeit zu beschreiben wird in Datenblättern häufig die Relativ Response (abgekürzt R/R_{\max}) angegeben, die das Verhältnis zwischen der Quanteneffizienz bei einer bestimmten Wellenlänge und der maximalen Quanteneffizienz angibt.

Linearität: Die Ladung eines Pixels steigt linear mit der Anzahl der einfallenden Photonen bis zum Erreichen der Full Well Capacity an. Im Ladungsverstärker kann es aber zu geringen Nicht-Linearitäten kommen, die im Allgemeinen nur bei sehr präzisen Farbmessaufgaben eine Rolle spielen. Die Linearität beschreibt, inwieweit das Ausgangssignal der Kamera linear mit der Zahl der einfallenden Photonen anwächst.

Rauschen: Die Bildinformation wird immer von einem zufälligen Rauschen überlagert, das verschiedene Ursachen hat. Eine Reihe von Rauscharten werden im Bildaufnahmesensor generiert, daneben entsteht aber auch Rauschen in der Kameraelektronik, worauf wir im Abschnitt 8.2.3 eingehen.

Photonenrauschen: Beim Photonenauslesen handelt es sich um einen unvermeidlichen Quanteneffekt, der unabhängig von der Qualität des Sensors auftritt. Wenn die Pixel im zweistelligen Prozentbereich ihrer Full Well Capacity gefüllt sind, ist es die größte Rauschquelle.

Fixed Pattern Noise: Der in Abschnitt 8.1.2 beschriebenen Fixed Pattern Noise tritt vor allem bei CMOS-Sensoren auf.

Bildaufnahmeeffekte: Bei der Bildaufnahme mit Halbleitersensoren treten verschiedene Verfälschungen gegenüber der aufgenommenen Szene auf. Typische Fehler sind:

Smear: Der sogenannte Smear tritt nur bei CCD-Sensoren auf. Da die Transportregister nie ganz gegen Licht abgeschirmt werden können, entstehen in ihnen störende Ladungsträger, die zu Beginn der Belichtungszeit nicht gelöscht werden. Beim Auslesen der Pixel werden diese Ladungsträger aus den Transportregistern zu den Ladungsträgern aus den Pixeln hinzugefügt.

Blooming: Wenn zuviel Licht auf ein Pixel fällt, entstehen mehr Ladungsträger als im Pixel gespeichert werden können. Die Ladungsträger, die nicht mehr gespeichert werden können, fließen in die Nachbarpixel ab, deren Helligkeit damit ansteigt. Dieser Effekt ist als Blooming bekannt. Da die Ladungsträger die Transportregister schwer überqueren können, breitet sich Blooming meistens senkrecht entlang der Spalten aus. Bei CCD-Sensoren wird durch sogenannte Anti-Blooming-Drains versucht die überschüssige Ladung geordnet abzuleiten, um Blooming zu verhindern. CMOS-Sensoren enthalten häufig bereits in den einzelnen Pixeln Anti-Blooming-Funktionen und sind deshalb häufig weniger anfällig für Blooming als CCD-Sensoren.

Als Abschluss dieses Abschnittes über Halbleitersensoren sind die wichtigsten Vor- und Nachteile von CCD- und CMOS-Sensoren zusammengestellt. Zunächst die Vorteile von CCD- gegenüber CMOS-Sensoren:

- Geringes Rauschen, insbesondere niedriger Fixed-Pattern-Noise.
- Niedriger Dunkelstrom
- Global Shutter
- Optimierte, ausgereifte Technologie

Demgegenüber stehen eine Reihe von Vorteilen von CMOS- gegenüber CCD-Sensoren:

- Niedrige Herstellungskosten bei Sensoren für den Massenmarkt
- Hoher Integrationsgrad
- Niedrige Leistungsaufnahme
- Auslesen von Bildausschnitten möglich
- Höhere Auslesegeschwindigkeit

8.2 Digitalkameras

Im Gegensatz zur analogen Datenverarbeitung älterer Kameras werden bei heutigen Digitalkameras, wie sie Abbildung 8.10 zu sehen sind, alle Signale des Bildaufnahmesensors in digitale Werte umgesetzt. Die digitalen Werte werden in der Kamera für die Übertragung über eine standardisierte, digitale Datenschnittstelle

aufbereitet. Damit ist die verlustfreie Übertragung der Bilddaten vom Sensor bis zum Bildverarbeitungsrechner sichergestellt.



Abbildung 8.10: Moderne Digitalkameras

Für die Leistungsfähigkeit einer Digitalkamera sind eine ganze Reihe von Funktionen und Eigenschaften von Bedeutung. Dieser Abschnitt beginnt mit einer Reihe grundlegender Funktionen von allen Kameras und dann speziell von Farbkameras. Danach werden Kennwerte für die Beschreibung von relevanten Kameraeigenschaften vorgestellt. Daran anschließend wollen wir auf die besonderen Bedingungen, denen Kameras in der industriellen Praxis ausgesetzt sind, eingehen.

8.2.1 Steuerung der Bildaufnahme

Digitale Kameras verfügen über einen Prozessor, der häufig ein sogenannter Field Programmable Gate Array (abgekürzt FPGA) ist. Dieser steuert die Bildaufnahme und bereitet die Bilddaten für die Übertragung an den Bildverarbeitungsrechner auf. Die Steuerung der Bildaufnahme ist von großer Bedeutung, um ein ausreichend helles und kontrastreiches Bild aufzunehmen. Die Steuerung der Bildaufnahme wird durch eine Reihe von Parametern beeinflusst:

Belichtungszeit: Der wichtigste Parameter für die Bildaufnahme ist die Belichtungszeit (englisch Exposure), mit der gesteuert wird wie lange Ladungen in den Pixeln gesammelt werden. Mit steigender Belichtungszeit wird das aufgenommene Bild heller, sie sollte also lang genug gewählt werden, um ein ausreichend helles Bild aufzunehmen. Manche Anwendungen lassen es allerdings nicht zu, die Belichtungszeit beliebig lang zu wählen. Bei bewegten Teilen beispielsweise muss man eine kurze Belichtungszeit wählen um die Bewegungsunschärfe zu minimieren.

Signalverstärkung: Wenn es aufgrund der Anwendung nicht möglich ist, die erforderliche Bildhelligkeit durch Verlängern der Belichtungszeit oder Optimieren der Beleuchtung und der Optik zu erreichen, kann das im Sensor erzeugte Signal

künstlich vor der Analog-Digital-Wandlung verstärkt werden. Mit dem Bildsignal wird allerdings auch das Rauschen verstärkt, so dass eine hohe Signalverstärkung (englisch Gain) oft zu einem merklichen Bildrauschen führen. Die Erhöhung der Bildhelligkeit durch höhere Verstärkung sollte deshalb nur angewandt werden, wenn die Möglichkeiten der Beleuchtung, des Objektives und der Belichtungszeit voll ausgeschöpft sind.

Automatische Helligkeitssteuerung: Meistens wird die Belichtungszeit und die Verstärkung der Kamera fest eingestellt. Wenn sich die Beleuchtung des Objektes ändert, wird das Bild entsprechend heller oder dunkler. Um dieser Helligkeitsänderung entgegen zu wirken, verfügen viele Kameras über eine automatische Nachführung der Belichtungszeit (englisch Automatic Exposure Control, abgekürzt AEC) und der Verstärkung (englisch Automatic GControl, abgekürzt AGC). In der industriellen Bildverarbeitung werden diese beiden Funktionen fast nie genutzt, da die Beleuchtungsverhältnisse meistens konstant sind und die Änderungen der Belichtungszeit und der Verstärkung nicht vorhersehbar sind.

Steuerung externer Beleuchtung: Es muss immer wieder betont werden, dass eine gute, an die Umgebung und das Prüfteil angepasste Beleuchtung von zentraler Bedeutung ist, um ein Bild zu erfassen, das man überhaupt auswerten kann. In der industriellen Bildverarbeitung werden häufig Blitzbeleuchtungen verwendet. Digitalkameras verfügen meistens über spezielle digitale Ausgänge, mit denen sie die Beleuchtung während der Bildaufnahmezeit einschalten können.

Synchronisierung der Bildaufnahme: Ein Bild muss immer genau dann aufgenommen werden, wenn sich das Prüfobjekt im Sichtfeld der Kamera befindet. Dazu muss die Bildaufnahme im richtigen Moment ausgelöst werden. Das Signal, mit dem die Bildaufnahme ausgelöst wird, wird als Trigger bezeichnet. Der Trigger kann grundsätzlich aus zwei verschiedenen Quellen stammen, entsprechend wird er als Hardwaretrigger oder als Softwaretrigger bezeichnet.

Hardwaretrigger: Beim Hardwaretrigger ist die Kamera direkt mit der Signalquelle über ein Kabel verbunden. Das Signal wird im Allgemeinen als Änderung eines Spannungspegels übertragen. Typische Signalquellen sind Lichtschranken. Wird die Lichtschranke vom Prüfteil durchbrochen, steigt der Ausgangspegel der Lichtschranke an, worauf die Kamera die Bildaufnahme startet. Hardwaretrigger werden häufig dann eingesetzt, wenn eine sehr genaue Synchronisation der Bildaufnahme nötig ist.

Softwaretrigger: Das Softwaretriggersignal wird von der Bildverarbeitungssoftware auf dem Auswerterechner mit einem Methodenaufruf generiert und über die digitale Schnittstelle zur Kamera übertragen. Aufgrund des längeren Signalwegs und den spezifischen Eigenschaften des Betriebssystems auf dem PC kann es beim Softwaretrigger zu Latenzzeiten kommen.

Bildrate: Die Anzahl der Bilder, die in einer Sekunde aufgenommen werden, wird als Bildrate bezeichnet. Die Bildrate hängt im Wesentlichen von der Belichtungszeit, der Zeit für das Auslesen des Bildes und der Zeit für die Übertragung des

Bildes von der Kamera in den Bildverarbeitungsrechner ab. Alle drei Zeiten sind umso größer, je mehr Pixel belichtet, ausgelesen und übertragen werden. Daher ist es offensichtlich, dass sich die Bildrate steigern lässt, indem man die Anzahl der Pixel reduziert. Zwei Methoden, um die Anzahl der Pixel zu reduzieren, sind Partial Scan und Binning. Neben der Reduktion der Pixelanzahl kann man die Bildrate durch das gleichzeitige Belichten und Auslesen der Bilddaten, wie in 8.1.2 beschrieben, erhöhen.

Partial Scan: Manche Sensoren erlauben es, dass nur eine klar begrenzte Teilmenge der Pixel ausgelesen wird. Diese Funktion wird als Partial Scan bezeichnet und der ausgelesene Teil des Sensors wie ein Suchbereich als Region Of Interest. Partial Scan kann dann zur Steigerung der Bildrate eingesetzt werden, wenn alle für die Auswertung relevanten Informationen in einem begrenzten Bildausschnitt enthalten sind.

Binning: Beim Binning werden die Ladungsträger benachbarter Pixel zu einem „Superpixel“ zusammengefasst. Dies kann horizontal, vertikal oder in beiden Richtungen erfolgen. Durch Binning wird die Anzahl der Pixel halbiert oder geviertelt, so dass die Bildrate höher wird. Da die lichtempfindliche Fläche der „Superpixel“ doppelt bzw. viermal so groß wie die der einzelnen Pixel ist, ist der „Superpixel“ deutlich lichtempfindlicher. Binning ist im Allgemeinen nur bei CCD-Sensoren möglich.

8.2.2 Aufnahme von Farbbildern

In 8.1.5 wurde gezeigt wie die Halbleitersensoren für die Aufnahme von Farbbildern aufgebaut sind. In diesem Abschnitt soll es darum gehen, wie aus den Rohdaten des Sensors ein Farbbild entsteht. Dahinter steht ein Prozess der das Demosaicing, die Farbkalibrierung, den Weißabgleich und die Ausgabe in einem verwendbaren Format umfasst.

Demosaicing: Wie bereits in 8.1.5 beschrieben, werden in der industriellen Bildverarbeitung meistens Ein-Chip-Kameras mit Farbfiltermaske verwendet. Da bei diesen Kameras für jedes Pixel nur der Wert eines der üblicherweise drei Farbkanäle vorliegt, müssen die Werte der übrigen beiden Farbkanäle aus den benachbarten Pixeln interpoliert werden. Diese Interpolation der Farbwerte wird als Demosaicing bezeichnet und wird im Allgemeinen ebenfalls vom Prozessor der Kamera durchgeführt. Beim in Abbildung 8.6 ganz links abgebildeten Bayer-Pattern, das der de-facto-Standard bei den Farbfiltermasken ist, ist die Hälfte der Pixel grün und jeweils ein Viertel der Pixel rot beziehungsweise blau. Für die Hälfte der Pixel muss also der Grünwert interpoliert werden und für jeweils drei Viertel der Pixel muss der Rotwert beziehungsweise der Blauwert interpoliert werden. Für diese Interpolation gibt es eine Vielzahl von verschiedenen Berechnungsmethoden. Bei der einfachsten werden die Farbwerte aus der Nachbarschaft einfach wiederholt, die komplexeren Methoden betrachten aktuell beispielsweise eine bis zu 7×7 Pi-

xel große Umgebung um den Punkt und führen aufwändige Berechnungen aus, um die Farbwerte zu interpolieren. Die einfachen Ansätze haben den Vorteil einer schnellen Ausführung, dafür bilden sich an Kanten häufig sogenannte Farbsäume, also parallel zur Kante verlaufende Streifen in einer anderen Farbe. Die aufwändigeren Methoden erzeugen Bilder mit schärferen Kanten ohne Farbsäume und mit einer genaueren Farbwiedergabe, wie in Abbildung 8.11 zu sehen ist. Letztendlich ist die Wahl der Interpolationsmethode damit ein Kompromiss zwischen Bildrate und Bildqualität, wobei es von der Prüfaufgabe abhängt, was für die spezifische Anlage wichtiger ist.

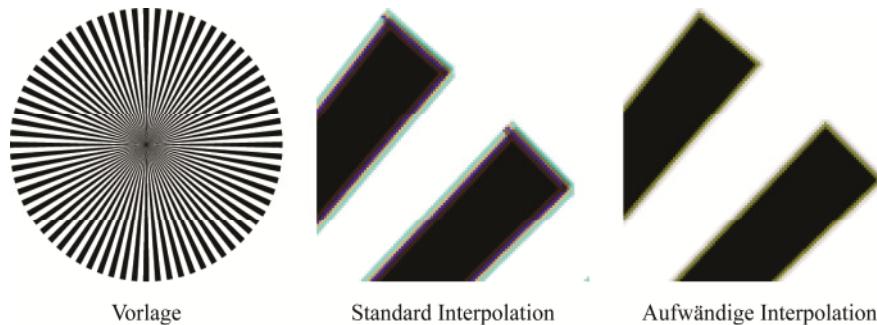


Abbildung 8.11: Vergleich zweier Demosaicing-Verfahren

Farbkalibrierung: Die Rohdaten und die daraus interpolierten RGB-Daten einer digitalen Farbkamera sind geräteabhängig, da sie von der spektralen Empfindlichkeit des Halbleitersensors und den Farben der Farbfiltermatrix abhängen. Wenn keine großen Anforderungen an die Qualität der Farberfassung gestellt werden, beispielsweise bei einer einfachen Klassifizierung von farbigen Objekten, sind diese gerätespezifischen Daten völlig ausreichend.

Wenn mit der Kamera Farbe exakt gemessen wird oder die Kamera für Visualisierungsaufgaben wie z. B. in Medizintechnik, Mikroskopie oder Polygraphie verwendet wird, müssen die gerätespezifischen Farbdaten in einen genormten Farbraum transformiert werden. Dies wird als Farbkalibrierung der Kamera bezeichnet. Verbreitete genormte Farbräume sind:

- CIE-XYZ-Farbraum, er umfasst alle sichtbaren Farben und baut auf den Farben Rot, Grün und Blau auf und wurde in den dreißiger Jahren des 20. Jahrhunderts von der Internationalen Beleuchtungskommision (französisch Commission Internationale de l'Éclairage, abgekürzt CIE) entwickelt.
- CIE-Lab-Farbraum, wurde zur farbmetrischen Bewertung von Farben ebenfalls von der CIE definiert.
- sRGB-Farbraum, ist für die Darstellung auf Computermonitoren angepasst.

Weißabgleich: Das Licht, das von einer Farbkamera erfasst wird, wird bestimmt durch die Farbe des Objekts und der Farbe des Lichts der Beleuchtung. Um vergleichbare Farbwerte zu erfassen, muss man eine Kamera deshalb an die aktuellen

Beleuchtungsbedingungen anpassen. Dieses Anpassen wird als Weißabgleich oder White Balance bezeichnet. Dabei werden die einzelnen Kanäle mit verschiedenen Faktoren verstärkt, um so das Ungleichgewicht zwischen den Kanälen auszugleichen.

Die spektrale Verteilung des Lichts einer Lichtquelle wird als Farbtemperatur bezeichnet und in Grad-Kelvin (abgekürzt °K) gemessen. Die Temperatur die angegeben wird, ist die Temperatur eines Körpers der Licht mit der gleichen spektralen Zusammensetzung als Wärmestrahlung abstrahlt. Übliche Farbtemperaturen sind 5000°K und 6500°K für Tageslicht (abgekürzt D50 und D65) und 2856°K (auch als Normlicht B bezeichnet) für das Licht von Glühbirnen.

Der Weißabgleich muss mit der Beleuchtung durchgeführt werden, die bei der eigentlichen Prüfaufgabe verwendet wird, da die Kamera an deren Farbe angepasst werden soll. Die Weißabgleichsfunktion optimiert die Verstärkungsfaktoren der einzelnen Kanäle so, dass möglichst viele Pixel identische Rot-, Grün- und Blauwerte haben, also grau sind. Deshalb sollte die Kamera zum Weißabgleich möglichst nur Grautöne aufnehmen. Einen optimalen Weißabgleich erzielt man, wenn die Kamera dabei nur einen Graukeil aufnimmt.

Ausgabeformate: Die Daten eines Farbbildes können in verschiedenen Formaten, sogenannten Ausgabeformaten, an den Bildverarbeitungs-PC übertragen werden. Man kann zwei Arten von Ausgabeformaten unterscheiden:

- Rohdaten
- Gerechnete Formate

Rohdaten: Als Rohdaten bezeichnet man die Originaldaten vom Sensor ohne Demosaicing in der Kamera. Die Farbinterpolation muss damit im PC durchgeführt werden, was einerseits den Rechner stärker belastet, andererseits die Möglichkeit eröffnet, eine speziell auf die vorliegenden Anwendung optimierte Farbinterpolation durchzuführen.

Gerechnete Formate: Typische gerechnete Formate sind das RGB- und YUV-Format. In der Bildverarbeitung ist das RGB-Format sehr verbreitet, da es ein sehr hardwarenahes Format ist. Sowohl Kameras als auch die meisten Monitore arbeiten auf der Basis von RGB. Das YUV-Format wird vor allem für Visualisierungsanwendungen eingesetzt, da man in diesem Format die Datenmenge so reduzieren kann, dass die Minderung der Bildqualität einem menschlichen Betrachter kaum auffällt. Für die industrielle Bildverarbeitung hat das YUV-Format dennoch nur geringe Bedeutung.

8.2.3 Kennwerte von Digitalkameras

Die Qualität und Leistungsfähigkeit einer Kamera kann durch eine Reihe von Kennwerten, wie z. B. die spektrale Empfindlichkeit (siehe 8.1.6), das Signal-Rausch-Verhältnis (englisch Signal Noise Ratio, abgekürzt SNR), die Dynamik oder die Linearität, beschrieben werden. Diese Parameter sowie die Messverfah-

ren um sie zu erfassen, sind im Standard EMVA 1288 der European Machine Vision Association (abgekürzt EMVA) definiert. Dieser Standard wird kontinuierlich durch ein Gremium, dem Kamera- und Sensorhersteller, Distributoren und Vertreter aus der Forschung angehören, weiterentwickelt, um den neuesten Entwicklungen in der Kamera- und Sensortechnik Rechnung zu tragen. Die Qualität einer Kamera hängt natürlich stark vom Kamerasensor ab, so dass viele Kennwerte wie beispielsweise die Linearität oder die sogenannte Homogenität sich maßgeblich auf Sensoreigenschaften beziehen, die wir im Abschnitt 8.1 dargestellt haben.

Signal-Rausch-Verhältnis: Eines der wichtigsten Qualitätsmerkmale einer Digitalkamera ist das Signal-Rausch-Verhältnis. Dabei ist die Bildqualität umso besser, je höher das Signal-Rausch-Verhältnis ist. Eines der in einer Kamera als Gesamtsystem auftretenden Rauscharten ist das Photonenrauschen, das bereits im Sensor entsteht und das wir daher bereits in Abschnitt 8.1.6 beschrieben haben. Daneben gibt es aber auch Rauscharten die ihre Ursache nicht nur im Sensor haben:

Thermisches Rauschen: Auch wenn kein Licht auf die Pixel trifft, werden durch thermische Effekte im Halbleiter Ladungen erzeugt. Diese Ladungsgeneration erfolgt in allen elektronischen Bauelementen einer Digitalkamera, d. h. in den Pixel, in den Transportregistern, in der Auslese- und Verstärkerelektronik, und wird als Dunkelstrom (englisch Dark signal) bezeichnet. Die Varianz des Dunkelstromes nennt man das thermische Rauschen, das sich von Pixel zu Pixel unterscheidet. Es trägt erst bei längeren Belichtungszeiten von über einer halben Sekunde einen relevanten Anteil zum Gesamtrauschen bei.

Verstärkerrauschen: Das Verstärker- oder Ausleserauschen entsteht beim Umwandeln der Ladungen in ein digitales Signal und wird deshalb maßgeblich durch die Qualität der Kameraelektronik bestimmt. Es dominiert nur bei sehr niedriger Aussteuerung und normalen Belichtungszeiten, da unter diesen Bedingungen das Photonenrauschen und das thermische Rauschen kleiner sind.

Dynamikumfang: Ein weitere wichtige Kennzahl für die Kameraqualität ist der Dynamikumfang. Dies ist der Quotient von größtem Helligkeitswert zu dem kleinsten Helligkeitswert, der sich vom Rauschen unterscheidet. Typische Werte hochwertiger Kameras liegen bei mehr als 50 dB.

Linearität: Die Linearität ist maßgeblich vom Sensor abhängig und wird in Abschnitt 8.1.6 beschrieben.

Homogenität: Die Homogenität ist ein Kennwert für den in Abschnitt 8.1.2 und 8.1.6 beschriebenen Fixed Pattern Noise von CMOS-Sensoren.

8.2.4 Einsatzbedingungen in Industrienumgebungen

Temperatur: Für die Qualität der Bildaufnahmen und die Lebensdauer der eingesetzten Kamera spielen die Betriebs- und Umgebungstemperatur eine entscheidende Rolle. Ein Teil des Kamerarauschens ist abhängig von der Sensortemperatur und steigt mit dieser an. Wärmequellen sind zum einen die Kamera selbst, bei der die Verlustleistung ein Maß für die erzeugte Wärme ist, und zum anderen externen Quellen, wie beispielsweise die Beleuchtung oder andere elektrische Verbraucher in der Umgebung der Kamera. Ein Qualitätsmerkmal hochwertiger Industriekameras ist deshalb eine möglichst geringe Verlustleistung. Aufgrund der begrenzten Platzverhältnisse in industriellen Sichtprüf anlagen sind oft Kameras mit kleinen Gehäusen notwendig, was allerdings die Wärmeabfuhr verschlechtert. Speziell bei solchen Kameras sollte man deshalb auf eine gute thermische Kopplung des Kameragehäuses mit der Kamerahalterung achten.

Schutzgrad: In vielen industriellen Prozessen sind die Kameras ungünstigen Umgebungsbedingungen wie Feuchtigkeit oder Staub ausgesetzt. Die Bildqualität einer Kamera wird natürlich erheblich verschlechtert, wenn Staub oder Wasser zwischen den Sensor und das Objektiv gelangen. Um solche Probleme zu vermeiden und auch unter schlechten Umgebungsbedingungen eine hohe Zuverlässigkeit sicherzustellen, können Digitalkameras mit einem speziellen Schutz gegen Fremdkörper und Feuchtigkeit ausgestattet werden. Dieser besteht meistens aus einem zusätzlichen Gehäuse um die Kamera und das Objektiv, wie sie in Abbildung 8.12 zu sehen sind. Für Anwendungen mit besonderen hygienischen Anforderungen, wie in der Lebensmittelindustrie, werden für diese Gehäuse korrosionsfeste Materialien wie z. B. Edelstahl verwendet. Der Grad des Schutzes der Kamera wird durch sogenannten IP-Schutzzahlen beschrieben, die in der Norm DIN EN 60529 festgelegt sind. Bei einer Kamera die mit der Aufschrift IP67 gekennzeichnet ist, bedeutet die 6, dass das Gehäuse staubdicht ist und das Berühren aktiver Teile mit einem Draht unmöglich ist. Mit der 7 garantiert der Hersteller, dass die Kamera zeitweilig in Wasser untergetaucht werden kann, ohne beschädigt zu werden.



Abbildung 8.12: Digitalkameras mit IP67-Schutzgehäuse (mit freundlicher Genehmigung der Firma Baumer Optronic (Radeberg))

Elektrischer Anschluss: Damit Digitalkameras zuverlässig funktionieren, sind eine stabile Spannungsversorgung und Schaltsignale mit einer guten Signalqualität unerlässlich. Unsaubere Spannungsverhältnisse und Störquellen führen in der industriellen Bildverarbeitung immer wieder zu Fehlverhalten von Kameras. Die Anforderungen, die an sämtliche elektrischen Bauteile hinsichtlich der elektrischen Pegel, der elektromagnetischen Verträglichkeit (EMV) und der elektrischen Sicherheit gestellt werden, sind in der Norm für Speicherprogrammierbare Steuerungen (Teil 2) IEC 61131-2:2007 beschrieben.

Komponenten: Für den zuverlässigen industriellen Einsatz von Digitalkameras in der industriellen Bildverarbeitung spielen neben geeigneten Kameras auch die weiteren Komponenten, wie z. B. Kabel, Schnittstellenkarten für den Bildverarbeitungsrechner, die Spannungsversorgung usw. eine entscheidende Rolle. Dabei ist es nicht ausreichend, dass die einzelnen Komponenten grundsätzlich geeignet sind, sondern sie müssen darüber hinaus auch aufeinander abgestimmt sein.

8.3 Bilddatenübertragung

Bis Ende der 90er Jahre nutzte die industrielle Bildverarbeitung vornehmlich analoge Kameras, die für die Bildübertragung die aus der Fernsehtechnik bekannten analogen Übertragungsstandards verwendeten. Die analogen Kamerasignale wurden mit speziellen Einstektkarten, sogenannten Frame Grabbern, im Bildverarbeitungsrechner digitalisiert und im Speicher des Rechners als digitale Bilder zur Weiterverarbeitung zur Verfügung gestellt. Mitte der 90er Jahre standen erstmals digitale Übertragungsverfahren zur Verfügung, die es erlaubten, die in der Kamera digitalisierten Bilddaten als digitale Signale störungsfrei und mit hohen Datenraten direkt auf den Rechner zu übertragen. Mittlerweile hat die digitale Datenübertragung die analoge bei neuen Anlagen beinahe vollständig verdrängt, weshalb wir hier nur auf die digitale Bildübertragung eingehen.

Die ersten voll digitalen Kameras kamen mit proprietären Schnittstellen wie LVDS-Schnittstellen oder serielle Schnittstellen wie HotLink™ auf den Markt. Aus der LVDS-Schnittstelle ging mit CameraLink® die erste speziell für die Bilddatenübertragung standardisierte Schnittstelle für Kamera, Kabel und Frame Grabber hervor. Während CameraLink® gezielt für die Bildverarbeitung entwickelt wurde, werden auch eine Reihe von digitalen Schnittstellen aus dem Consumer-Markt in der industrielle Bildverarbeitung eingesetzt. Den Anfang machte der Standard IEEE1394, besser bekannt als FireWire®, gefolgt von Universal Serial Bus (abgekürzt USB) und schließlich Gigabit-Ethernet (abgekürzt GigE). Diese Schnittstellen werden im Folgenden ausführlich dargestellt. Einen kurzen Überblick über die Eigenschaften der Schnittstellen gibt die Tabelle 8.3.

Tabelle 8.3: Übersicht gebräuchlicher digitaler Kameraschnittstellen

	CameraLink®	FireWire®	USB 2.0	Gigabit-Ethernet
Übertragungsrate	6,8 Gbit/s b: 800 Mbit/s	a: 400 Mbit/s b: 800 Mbit/s	480 Mbit/s	1000 Mbit/s
Kabellänge	10 m	4,5 m	5 m	100 m
Energieversorgung	11–13 V, 0,4 A*	8–30 V, max 15 W	5 V, 0,5 A	37–57 V, max. 13 W**
Verbindung	Point to Point	Bus	Bus	Netzwerk
Schnittstelle am PC	Frame Grabber	Teilweise vor- handen	Alle PCs	Alle PCs
Echtzeitfähigkeit	Sehr gut	Gut	Mäßig	Gut
Prozessorlast	Sehr niedrig	Niedrig	Mittel bis hoch	Niedrig bis mit- tel
Verfügbarkeit von Kom- ponenten	Mittel	Gut	Sehr gut	Sehr gut
Kosten	Hoch	Mittel	Niedrig	Mittel
Industrietauglichkeit	Gut	Gut	Mäßig	Gut

* nur bei Power over CameraLink® (abgekürzt PoCL)

** nur bei Power over Ethernet (abgekürzt PoE)

8.3.1 CameraLink®

Der CameraLink®-Standard war die erste speziell für die Übertragung von Bilddaten entwickelte Schnittstelle für die industrielle Bildverarbeitung. Die Entwicklung und Standardisierung wurde von der US-amerikanischen Automated Imaging Association (abgekürzt AIA) geleitet. Obwohl dieser Standard schon in den neunziger Jahren entwickelt wurde, ist er durch die hohe Übertragungsgeschwindigkeit speziell in High-Speed-Anwendungen nach wie vor die Schnittstelle der Wahl.

Da CameraLink® keine Standard-PC-Schnittstelle ist, benötigt man spezielle Einstektkarten, sogenannte CameraLink® Frame Grabber, um die Bilddaten in den Bildverarbeitungsrechner zu übertragen. Die Verwendung von Frame Grabbern hat den Vorteil, dass dadurch die Übertragung der Bilddaten unabhängig von den Latenzen des Rechnerbetriebssystems ist. Außerdem sinkt die Belastung des Prozessors des Bildverarbeitungscomputers, da der Frame Grabber den Empfang der Bilddaten selbstständig verwaltet und die Bilddaten mittels Direct Memory Access (abgekürzt DMA) direkt in den Arbeitsspeicher des Rechners überträgt. Allerdings handelt es sich bei CameraLink® Frame Grabber um teure Spezialhardware. Häufig können Frame Grabber neben der reinen Bildübertragung auch noch Bildvorverarbeitungsfunktionen, wie z. B. Filterungen, vor der Übertragung in den Arbeitsspeicher durchführen. Damit kann der Bildverarbeitungsrechner noch weiter entlastet werden, allerdings wird die Anwendung damit von der verwendeten Hardware abhängig.

Die Datenübertragung im CameraLink®-Standard basiert auf der sogenannten Channel-Link-Technologie der Firma National Semiconductor (Santa Clara, CA, USA). Ein Channel Link kann bei einer Taktfrequenz von bis zu 85 MHz in einem Takt 28 Bit übertragen. Diese 28 Bit werden zu drei so genannten Ports mit jeweils acht Bit zusammengefasst. Von den übrigen vier Bits dienen drei dazu anzuzeigen, dass die Bildübertragung (englisch Frame valid), die Zeilenübertragung (englisch Line valid) und die Datenübertragung (englisch Data valid) abgeschlossen ist. Für unterschiedliche Bandbreitenbedürfnisse sind verschiedene sogenannte Konfigurationen definiert, die in Tabelle 8.4 zusammengestellt sind. Für jede Konfiguration ist die Zuordnung der Ports zu den Bilddaten für unterschiedliche Bittiefen und Pixelformate definiert.

Tabelle 8.4 CameraLink® Konfigurationen

Konfiguration	Channel Links	Ports	Datenrate	Kabelanzahl
Base	1	3	2,04 Gbit/s	1
Medium	2	6	4,08 Gbit/s	2
Full	3	8	5,44 Gbit/s	2

Neben der Übertragung der Daten von der Kamera zum Bildverarbeitungscomputer stellt CameraLink® zwei serielle Übertragungskanäle für die Kommunikation zwischen Computer und Kamera zur Verfügung. Damit ist es möglich die Kamera ohne zusätzliche Hardware zu konfigurieren. Da das Format der Konfigurationsdaten nicht standardisiert ist, sind zur Konfiguration herstellerspezifische Programme nötig. Darüber hinaus stehen vier Steuerbits zur Verfügung, mit denen beispielsweise die Kamera vom Frame Grabber aus getriggert werden kann. Diese Steuerbits werden als CC1, CC2, CC3 und CC4 bezeichnet.

Ein Nachteil von CameraLink® gegenüber Bussystemen wie FireWire oder USB war, dass es nicht möglich war, die Kameras über das Datenkabel mit Spannung zu versorgen, so dass immer ein zusätzliches Kabel für die Spannungsversorgung erforderlich war. Mit der Version 1.2 wurde dieser Nachteil beseitigt durch die Einführung von Power over CameraLink® (abgekürzt PoCL), also der Spannungsübertragung über das Datenkabel. Der Standard sieht vor, dass die Kamera mit einer Spannung von 11–13 V bei einem maximalen Strom von 0,4 A versorgt wird.

Die nächste geplante Erweiterung des Standards ist die Steigerung der Datenübertragungsrate auf 6,8 Gbit/s über zehn Ports.

Wie man der Tabelle 8.4 entnehmen kann sind für die Verbindung zwischen Kamera und Frame Grabber je nach Konfiguration ein oder zwei CameraLink®-Kabel erforderlich. Da ein Anschluss eines Frame Grabbers nur für eine Kamera genutzt werden kann, handelt es sich um eine Punkt-zu-Punkt Architektur. Damit kann es nicht zu einer Konkurrenz um die Bandbreite kommen, statt dessen steht der Kamera immer die volle Bandbreite zur Verfügung.

Da die Daten parallel übertragen werden, benötigen CameraLink®-Kabel relative viele Adern, wodurch hohe Qualitätsanforderungen an Kabel und Stecker ge-

stellt werden. Dies schlägt sich natürlich im Preis der Komponenten nieder und begrenzt außerdem die Kabellänge auf zehn Meter, allerdings ermöglichen neue Kabel mit geringerer Dämpfung auch größere Längen. Als Steckerformat definierte der Standard ursprünglich nur große 26-polige Stecker im Mini-Delta-Ribbon-Format. Dem Trend der Miniaturisierung folgend, wurde der Standard auch auf Stecker im Shrunk-Delta-Ribbon-Format erweitert.

8.3.2 FireWire®

FireWire® ist ein serieller Bus, der von den Firmen Apple Inc. (Cupertino, CA, USA) und Sony Corporation (Tokio, Japan) entwickelt wurde, um Multimediaräte, wie Camcorder, Videorekorder, digitale Fotokameras oder externe Festplatten an Computer anzuschließen. Die erste Spezifikation als Standard erfolgte im Jahre 1995 unter der Bezeichnung IEEE1394a-1995. Der Standard definiert die serielle Datenübertragung mit einer maximalen Übertragungsrate von 400 Mbit/s. Im Jahre 2002 wurde der erweiterte Standard IEEE1394b mit Übertragungsraten von 800 Mbit/s, 1600 Mbit/s und 3200 Mbit/s veröffentlicht. Während für Übertragungsraten von 800 Mbit/s bereits 2003 erste Chips verfügbar waren, sind für 1600 Mbit/s und 3200 Mbit/s bislang keine Implementierungen auf dem Markt. Die Abbildung 8.13 zeigt eine Digitalkamera mit einem IEEE1394b-Anschluss.

Die hohe Datenübertragungsrate und die einfache Handhabung von FireWire® führten dazu, dass er schnell als Schnittstelle für die Bildübertragung von Digitalkameras in der industriellen Bildverarbeitung eingesetzt wurde. Weitere Gründe sind die große Störunempfindlichkeit, die Plug-and-Play-Fähigkeit und die Möglichkeit, die angeschlossenen Kameras mit ausreichend Spannung zu versorgen, um auch leistungsfähige Kameras mit großen Bildformaten ohne zusätzliches Versorgungskabel zu betreiben. In den letzten Jahren hat sich FireWire® in vielen Anwendungen der industriellen Bildverarbeitung als stabile und äußerst zuverlässige Schnittstelle erwiesen. Die wichtigsten Merkmale von FireWire® sind:

- Plug-and-Play-Fähigkeit; das An- und Abstecken von Kameras im laufenden Betrieb wird detektiert.
- Spannungsversorgung; von typischen PC-Einsteckkarten werden die Kameras mit 12 V und bis zu 500–750 mA elektrischer Energie versorgt.
- Erkennung der Übertragungsraten; der sogenannte Busmaster erkennt automatisch die maximale Übertragungsraten der Kameras.
- Das Kaskadieren mehrerer Geräte ist möglich (sogenannte Daisy Chain).
- Abwärtskompatibilität; Kameras die den neueren IEEE1394b Standard unterstützen, sind abwärtskompatibel zum IEEE1394a Standard.

Für die weitere Verwendung von FireWire® könnte es problematisch sein, dass USB sich als Schnittstelle von PC-Peripherie weitgehend durchgesetzt hat und deshalb FireWire®-Schnittstellen nicht mehr standardmäßig in PCs eingebaut sind.



Abbildung 8.13: IEEE1394b-Kamera (oben FireWire-Anschluss, unten rechts Triggereingang, mit freundlicher Genehmigung der Firma Baumer Optronic (Radeberg))

Isochrone Bilddatenübertragung: Für die Übertragung von digitalen Bildern über FireWire® wurde die Option einer sogenannten isochronen Bilddatenübertragung vorgesehen. Dabei wird vom sogenannten Busmaster alle 125 µs ein Signal generiert, an das alle am FireWire®-Bus angeschlossenen Kameras ihre Bilddaten anhängen können. Die Größe der Datenblöcke, die angehängt werden können, kann konfiguriert werden und bestimmt die Übertragungsrate. In Tabelle 8.5 sind typische Paketgrößen und die daraus resultierenden Übertragungsraten zusammengestellt. Die isochrone Bilddatenübertragung bietet keine Möglichkeit um Übertragungsfehler zu erkennen, so dass im übertragenen Bild falsche Pixelwerte auftreten können. Insbesondere in störungsreichen Umfeldern, wie sie in der industriellen Bildverarbeitung häufig vorkommen, kann dies zu Problemen führen.

Tabelle 8.5 Paketgröße und maximale Datenrate bei isochroner Bilddatenübertragung

Übertragungsrate	Paketgröße	Maximale Datenrate
100 Mbit/s	1024 Byte	8 MByte/s
200 Mbit/s	2048 Byte	16 MByte/s
400 Mbit/s	4096 Byte	32 MByte/s
800 Mbit/s	8192 Byte	64 MByte/s

DCAM-Standard: Basierend auf der isochronen Bilddatenübertragung wurde die sogenannte 1394-based Digital Camera Specification (abgekürzt DCAM-Standard, in den USA als Instrumentation and Industrial Digital Camera Standard, abgekürzt IIDD-Standard, bezeichnet) entwickelt, die einheitliche Befehle und Formate für die Steuerung und Datenübertragung digitaler Kameras definiert. Damit ist es möglich die Kameras unterschiedlicher Hersteller mit einer Implementierung zu steuern. Allerdings ist der im Standard definierte Funktionsumfang gering.

Asynchrone Bilddatenübertragung: Im Gegensatz zur isochronen Bilddatenübertragung bietet die sogenannte asynchrone eine gesicherte Übertragung, da jedes Datenpaket unter anderem eine Prüfsumme enthält, mit der der Empfänger feststellen kann, ob die Daten unbeschädigt sind. Wenn der Empfänger einen Feh-

ler feststellt, wird das Datenpaket von der Kamera erneut angefordert. Da unbeschädigte Datenpakete vom Empfänger quittiert werden, kann die Kamera bei ausbleibender Quittierung verlorenen Pakete erneut übertragen. Mit der asynchronen Bilddatenübertragung lassen sich höhere Datenraten erzielen als mit der isochronen Bilddatenübertragung.

Tabelle 8.6 Paketgröße und maximale Datenrate bei asynchroner Bilddatenübertragung

Übertragungsrate	Paketgröße	Maximale Datenrate
400 Mbit/s	2048 Byte	40 MByte/s
800 Mbit/s	4096 Byte	80 MByte/s

Für den Einsatz von FireWire-Kameras sind eine Reihe von zusätzlichen Komponenten erforderlich, beispielsweise:

- Interfacekarten; für den Anschluss von FireWire®-Kameras an den Bildverarbeitungscomputer, falls keine FireWire®-Schnittstelle standardmäßig auf dem Motherboard verfügbar ist.
- Hubs; erlauben das Anbinden mehrerer FireWire®-Kameras und können als Repeater zur Verlängerung der Leitungslänge eingesetzt werden.
- Kabel; im IEEE1394-Standard wird eine Leitungslänge von bis zu 4,5 m spezifiziert. Allerdings erlauben hochwertige FireWire®-Industriekabel eine sichere Datenübertragung mit Leitungslängen von bis zu zehn Metern auch im industriellen Umfeld.

Leider sind nicht alle am Markt verfügbaren Komponenten für den Einsatz in der industriellen Bildverarbeitung geeignet, da sie entweder nicht mit einer ausreichenden Zuverlässigkeit funktionieren oder die sonstigen technischen Anforderungen nicht erfüllen. Im Allgemeinen stellen die Hersteller von FireWire®-Kameras Kompatibilitätslisten zur Verfügung, denen man entnehmen kann, welche Komponenten für den Einsatz in der industriellen Bildverarbeitung geeignet sind.

8.3.3 USB

Der Universal Serial Bus (abgekürzt USB) ist die wohl wichtigste Schnittstelle für die Verbindung von PCs mit Peripherie-Geräte des letzten Jahrzehnts. Sie wurde in den 90er Jahren von der Firma Intel Corporation (Santa Clara, CA, USA) entwickelt. Im Jahr 2000 wurde die Version 2.0 veröffentlicht, die eine Datenrate von 480 Mbit/s erlaubt. Da damit eine ausreichende Datenrate für die Übertragung von Bilddaten zur Verfügung stand, wurde USB auch in der industriellen Bildverarbeitung eingesetzt. Vorteilhaft dafür war sicherlich die sehr große Verbreitung von USB im Consumer-Bereich, sowie die Plug-and-Play-Fähigkeit und die Möglichkeit Kameras über das Datenkabel mit Spannung zu versorgen. Vor allem die gegenüber FireWire geringere EMV-Festigkeit, die höhere Belastung des Prozessors

des Bildverarbeitungscomputer und die geringere Echtzeitfähigkeit verhinderten, dass sich USB als Bilddatenschnittstelle in der industriellen Bildverarbeitung durchsetzen konnte. So blieb auch eine Standardisierung der Bilddatenübertragung mit USB für die industrielle Bildverarbeitung bislang aus. In wie weit sich die 2008 veröffentlichte Version 3.0 in der industriellen Bildverarbeitung etablieren kann, bleibt abzuwarten.

Tabelle 8.7 Datenraten von USB

Spezifikation	Geschwindigkeitsklasse	Datenrate
1.1	Low-Speed	1,5 Mbit/s
	Full-Speed	12 Mbit/s
2.0	High-Speed	480 Mbit/s
3.0	Super-Speed	5 Gbit/s

8.3.4 Gigabit-Ethernet

Das Anfang der achtziger Jahre entstandene Ethernet, hat sich mittlerweile als de-facto-Standard für lokale Computer-Netzwerke durchgesetzt. Die dritte Generation wird aufgrund ihrer Datenrate von einem 1 Gbit/s als Gigabit-Ethernet (abgekürzt GigE) oder 1000BASE-X bezeichnet. Diese Datenrate macht GigE auch für die Übertragung von digitalen Bilddaten interessant.

GigE Vision: Unter dem Dach der AIA, die bereits für die Entwicklung des CameraLink®-Standards verantwortlich war, wurde der GigE Vision genannte Standard für die Bilddatenübertragung von Digitalkameras in der industriellen Bildverarbeitung entwickelt. Wie der Name schon sagt, wird Gigabit-Ethernet als Übertragungsmedium genutzt. Der Standard verwendet dabei einige in der Informationstechnik verbreitete Protokolle, wie in Tabelle 8.8 zusammengefasst.

Tabelle 8.8 Protokollsichten von GigE Vision

Schicht im OSI-Referenzmodell	Protokoll	Standard	Aufgabe
Application (7)	Nutzdaten		Verarbeitung der Bilddaten
Session (5)	GVCP/GVSP	GigE Vision	Bildübertragung und Kamerasteuerung
Transport (4)	User Datagram Protocol (abgekürzt UDP)	RFC768	Datenaustausch zwischen Endteilnehmern
Network (3)	Internet Protocol (abgekürzt IP)	RFC791	Adressierung der Daten
Datalink (2)	Ethernet Frame	IEEE802.3	Synchronisation zwischen Sender und Empfänger
Physical (1)	Hardware	IEEE802.3	Übertragung

Der GigE-Vision-Standard definiert ein Protokoll, über das eine Reihe grundlegender Funktionen realisiert werden, beispielsweise:

- Erkennung von GigE-Vision-Kameras im Netzwerk
- Überwachung der angeschlossenen Geräte
- Senden und Empfangen der Bild- und Steuerdaten
- Fragmentierung großer Datenmengen
- Fehlererkennung und Fehlerkorrektur

Diese Funktionen sind in den verschiedenen Elementen zusammengefasst, aus denen der GigE Vision-Standard besteht:

- GigE Device Discovery Protocol, der Mechanismus für das Auffinden von GigE-Vision-Kameras im Netzwerk und deren Einbindung.
- GigE Vision Control Protocol, für die Steuerung einer Kamera, den Austausch von Nachrichten, die Synchronisierung von Kamera und Bildverarbeitungscomputer und das Absenden asynchroner Nachrichten durch die Kamera.
- GigE Vision Streaming Protocol, die Beschreibung der Bilddatenströme und der Datentypen.

Außerdem definiert GigE-Vision noch einige Kameraregister (so genannte bootstrap register) und deren Inhalte, die für die Abwicklung der Kommunikation zwischen Bildverarbeitungsrechner und Kamera erforderlich sind.

Netzwerk-Topologie: Die einfachste Möglichkeit eine Kamera mit dem Bildverarbeitungscomputer zu verbinden ist die Punkt-zu-Punkt-Verbindung. Dabei wird die Kamera direkt mit einem Ethernet-Port am PC verbunden, so dass für die Übertragung der Bilddaten die volle Bandbreite zur Verfügung steht.

Wenn mehrere Kameras an einem PC mit Punkt-zu-Punkt-Verbindungen angeschlossen werden, sind auch mehrere Ethernet-Ports im PC nötig. Um deren Anzahl zu reduzieren, kann man ein Netzwerk aufbauen, in dem man mehrere Kameras mit einem Switch verbindet und den Switch mit einem Ethernet-Port des PCs. Damit müssen die Bilddaten aller Kameras über die eine Verbindung von Switch zu PC übertragen werden, so dass sich die Kameras die Übertragungsbandbreite dieser Verbindung teilen müssen. Somit reduziert sich durch den Einsatz von Switches die für jede einzelne Kamera zur Verfügung stehende Datenbandbreite. Das Netzwerk aus Kameras und PC muss außerdem verwaltet werden, damit die verfügbare Datenbandbreite optimal genutzt wird und die Kameras und der PC problemlos miteinander kommunizieren können. Mit einem derartigen Netzwerk ist es auch möglich, die Bilddaten einer Kamera an mehrere PCs zu übertragen. So kann das Bild zum einen für die Auswertung zum Bildverarbeitungsrechner übertragen werden und gleichzeitig an das HMI der Anlage übertragen werden und dort angezeigt werden.

Komponenten: Ethernet wurde für den Austausch von Daten zwischen Computern entwickelt, so dass die Übertragung von Bilddaten ungewöhnliche Anforderungen an das Übertragungsmedium stellt. Für den Einsatz von Gigabit Ethernet

als Schnittstelle in der industrielle Bildverarbeitung muss man deshalb darauf achten, dass alle Komponenten, insbesondere die Ethernet-Adapter im PC und möglicherweise eingesetzte Switches, den besonderen Anforderungen gewachsen sind. Wie auch bei FireWire stellen die Kamerahersteller von Gigabit Ethernet Kameras Listen mit geeigneten Komponenten zur Verfügung.

Steckverbinder: Als Steckverbinder werden die bei Ethernet üblichen genormten achtpoligen Modular Plugs mit einer Belegung nach RJ-45 eingesetzt, die für den Einsatz in der industriellen Bildverarbeitung häufig eine zusätzliche Schraubensicherung haben, wie sie in Abbildung 8.14 zu sehen ist. Für höhere Schutzgrade sind diese Steckverbinder nicht geeignet, so dass häufig M12-Rundsteckverbinder verwendet werden. Die sind allerdings bisher nur für Ethernet bis 100 Mbit/s (100BASE-T) mit vier Polen als sogenannte D-Codierung genormt. Für eine achtpolige Steckerbelegung, wie sie für Gigabit Ethernet erforderlich ist, existieren bislang nur proprietäre Lösungen.



Abbildung 8.14: Gigabit Ethernet Kamera (oben Ethernet-Anschluss, unten links Spannungsversorgung, unten rechts Triggereingang, mit freundlicher Genehmigung der Firma Baumer Optronic (Radeberg))

Kabel: Heute werden für die Verbindung der Kameras mit dem Bildverarbeitungsrechner oder Switch ausschließlich sogenannte Twisted-Pair-Kabel verwendet, während die sonst als Netzwerkkabel durchaus üblichen Glasfaserkabel hier nicht verwendet werden. Die Qualität der Kabel ist ein wichtigster Aspekt für den zuverlässigen Betrieb von GigE Vision Systemen in der industriellen Bildverarbeitung. Im industriellen Umfeld sollten mindestens Kabel der Qualitätsstufe Cat. 6 verwendet werden, um eine ausreichende Schirmung gegenüber Störfeldern zu haben.

Power over Ethernet: Genauso wie CameraLink[®] verfügt Ethernet nicht über die Möglichkeit Kameras über das Datenkabel mit Spannung zu versorgen. Entsprechend CameraLink[®] wurde der Ethernet-Standard erweitert um das sogenannte Power over Ethernet (abgekürzt PoE). Bei PoE wird ein Gleichstrom in die Doppeladern der Twisted-Pair-Kabel eingespeist, so dass Kameras mit Spannung versorgt werden können. Die Kennwerte der Spannungsversorgung von PoE sind:

- Speisespannung: 44–57 V
- Maximaler Speisestrom: 350 mA
- Maximaler Schleifenwiderstand: 20 Ω
- Mindestens verfügbare Leistung am Empfänger: 12,95 W

Die Spannung wird von sogenannten Midspan Devices oder Power Injectoren in die Netzwerkkabel eingespeist, die zwischen dem Bildverarbeitungsrechner und den Kameras eingebaut werden. Statt diesen Geräten kann man auch sogenannte PoE-fähige Switches verwenden um die Versorgungsspannung einzuspeisen.

Steigerung der Übertragungsbandbreite: Bei hochauflösenden Sensoren oder hohen Bildraten gelangt Gigabit Ethernet mit 1 Gbit/s an die Grenzen der Übertragungsbandbreiten. Um diese weiter zu steigern ist insbesondere der Wechsel zu 10G-Ethernet der nächsten Ethernet-Generation mit 10 Gbit/s Übertragungsbandbreite möglich. Als Alternative bietet sich die Bündelung mehrerer physikalischer Gigabit-Ethernet-Verbindungen zu einem logischen Kanal an, wofür beispielsweise das sogenannte Link Aggregation Control Protocol (abgekürzt LACP) eingesetzt werden kann.

8.4 Zeilenkameras

Bislang haben wir in diesem Kapitel immer vorausgesetzt, dass die Kamera einen Matrixsensor besitzt und somit ein rechteckiges Bild aufnimmt und an den Rechner überträgt. Solche Kameras bezeichnet man auch als Flächenkameras.

Im Gegensatz hierzu enthalten Zeilenkameras einen Zeilensor der nur über eine einzige Zeile lichtempfindlicher Elemente verfügt. Dieser kann eine sehr hohe Auflösung besitzen, typische Auflösungen sind 2048, 4096 oder 8192 Pixel. Zeilenkameras werden üblicherweise für die Prüfung bewegter oder sehr großer Objekte eingesetzt oder für die Überwachung der Fertigung kontinuierlicher Materialien, wie beispielsweise Tapeten. Sie eignen sich sehr gut zur Prüfung rotations-symmetrischer Objekte, die vor der Kamera gedreht werden, wie es Abbildung 8.15 zeigt. Zeilenkameras nutzen häufig CameraLink® als Bilddatenschnittstelle.

8.4.1 Typen von Zeilenkamera-Anwendungen

Da eine Zeilenkamera kein zusammenhängendes rechteckiges Bild aufnimmt wie eine Matrixkamera, sondern immer nur einzelne Zeilen großer Länge, folgt die Verarbeitung von Zeilenkamerabildern eigenen Gesetzen. Generell gilt immer, dass zwischen der Kamera und dem Prüfteil eine Relativbewegung vorhanden sein muss, da die Kamera sonst in ihrer Sensorzeile immer dieselbe Information aufnehmen würde. Je nachdem, wie diese Bewegung in ein Bild umgesetzt wird, kann man einige grundsätzliche Typen von Zeilenkamera-Anwendungen unterscheiden.

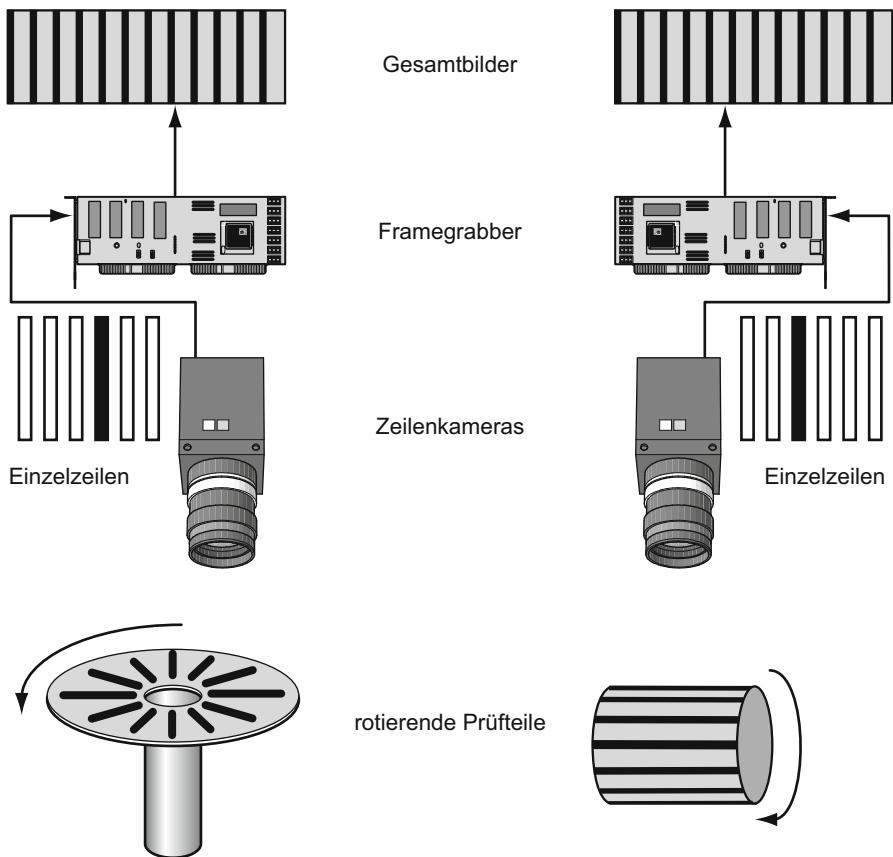


Abbildung 8.15: Abbildung rotierender Objekte durch eine Zeilenkamera

Zeilenweise Verarbeitung. Viele preiswerte Zeilenkamerasysteme verarbeiten immer nur die gerade aktuelle Zeile. Fragestellungen wie die Anwesenheit eines Teils, das Vorhandensein von Löchern in einem Teil, sogar einfache Vermessungen, bei denen nur festgestellt werden muss, ob eine bestimmte Mindestzahl von Zeilen hintereinander hell oder dunkel ist, lassen sich auf diese Weise durchaus lösen. Solche Systeme werden oft als *intelligente Kameras* aufgebaut, d. h. das Kameragehäuse enthält einen Prozessor und Speicher, so dass die gesamte Verarbeitung in der Kamera durchgeführt werden kann. Solche Kameras werden üblicherweise mit speziellen Einrichtprogrammen auf einem PC konfiguriert und die fertigen Verarbeitungsprogramme werden dann seriell oder über ein Netzwerk auf die Kamera heruntergeladen. Der Vorteil solcher Systeme liegt in ihrer geringen Baugröße und dem niedrigen Anschaffungspreis, aber durch die umständliche Konfiguration und die einfache Verarbeitungsweise eignen sie sich nicht für komplexe Aufgaben. Man könnte sie in etwa mit hochentwickelten Lichtschranken vergleichen.

Objektweise Verarbeitung. Bei dieser Methode sammelt die Kamera oder der Frame Grabber eine gewisse Anzahl von Zeilen und setzt sie zu einem rechteckigen Bild zusammen. Diese Methode ist vor allem für solche Anwendungen geeignet, in denen es weniger um die Fähigkeit der Zeilenkamera zur kontinuierlichen Bildaufnahme geht, sondern um die Aufnahme ungünstig geformter Einzelobjekte, z. B. sehr großer oder runder Prüfteile, wie in Abbildung 8.15. In einem solchen Falle löst man die Rotation und die Bildaufnahme durch dasselbe Triggersignal aus und sorgt dafür, dass während der gesamten Rotation die von der Zeilenkamera aufgenommenen Zeilen im Frame Grabber oder in der Kamera gesammelt werden. Danach liegt ein rechteckiges digitales Bild vor, das eine Abwicklung der Prüfteiloberfläche von 360° oder mehr darstellt. Dieses Bild kann dann auf dieselbe Weise untersucht werden wie das Bild einer Flächenkamera.

Kontinuierliche Verarbeitung. Die aufwändigste Methode kombiniert beide Verfahren. Wie bei der objektweisen Verarbeitung werden rechteckige Bilder verwendet, damit man komplexe Merkmale bilden kann, die sich nicht aufgrund der Information nur einer einzigen Zeile oder durch simples Zählen von Zeilen berechnen lassen. Andererseits wird das Bild – wie bei der zeilenweisen Verarbeitung – mit jeder neu aufgenommenen Zeile aktualisiert. Das stellt dann auch spezielle Anforderungen an den Frame Grabber, der in einem Ringpuffer stets die jüngste Zeile anhängen und die älteste verwerfen muss, so dass immer ein aktuelles rechteckiges Bild bestimmter Größe zur Verarbeitung verfügbar ist. Diese Verarbeitung muss mit der Scanrate Schritt halten, d. h. bevor die nächste neue Zeile eintrifft, muss das gesamte derzeitige Bild verarbeitet sein. Mit dieser Methode lassen sich kontinuierliche Materialströme, wie etwa in der Stahl- oder Textilindustrie, überwachen und mit komplexen Algorithmen auswerten. Die hierfür notwendigen Übertragungs- und Rechenleistungen stellen erhebliche Anforderungen an die Rechnerhardware.

Wir wollen uns hier mit der für PC-basierte Systeme besonders interessanten objektweisen Verarbeitung beschäftigen. Auch wenn dabei, ähnlich wie bei Anwendungen mit Flächenkameras, ein einzelnes rechteckiges Bild verarbeitet wird, sind bestimmte Besonderheiten zu beachten, auf die wir im Folgenden eingehen werden.

8.4.2 Ortsauflösung bei Zeilenkameras

Mit Flächenkameras aufgenommene Bilder haben eine in x- und y-Richtung für eine gegebene Anordnung feste Ortsauflösung. Diese wird durch das aufgenommene Bildfeld und die Spalten- und Zeilenzahl des Sensors bestimmt. Bei einer Zeilenkamera gilt eine solche feste Beziehung, die durch die Sensorgröße, die Optik und das Bildfeld bestimmt wird, nur in Richtung der Sensorzeile. Senkrecht zur Sensorzeile wird die Ortsauflösung von der Geschwindigkeit der Relativbewegung zwischen Objekt und Kamera und der Zeilenfrequenz bestimmt.

In 8.4.1 hatten wir bereits erwähnt, dass für die Aufnahme eines Bildes mit einer Zeilenkamera immer eine Relativbewegung zwischen Objekt und Kamera vorhanden sein muss. Diese kann sowohl translatorisch als auch rotatorisch erfolgen. Nehmen wir an, ein Werkstück von 20 mm Durchmesser werde vor einer Zeilenkamera mit einer Umdrehungsgeschwindigkeit von 1/s gedreht. Nehmen wir weiter an, dass die Kamera mit einer Zeilenfrequenz von 1 kHz arbeite, also 1000 Zeilen in der Sekunde aufnimmt. Während einer vollen Umdrehung des Teils werden dann gerade 1000 Zeilen aufgenommen. Bei einem Durchmesser von 20 mm wird folglich eine Mantelfläche mit einem Umfang von $\pi \times 20\text{mm} \approx 62,8\text{ mm}$ in das digitale Bild „abgewickelt“. Die Ortsauflösung beträgt in dieser Richtung folglich 0,0628 mm/Pixel.

Wenn jetzt die Umdrehungsgeschwindigkeit des Teils verdoppelt wird, dann wird *dieselbe Mantelfläche* in der halben Zeit aufgenommen, folglich entsteht nur die Hälfte der Zeilen und die Größe eines Bildpunktes in dieser Richtung vergrößert sich auf das Doppelte, nämlich 0,1256 mm/Pixel. Wenn umgekehrt die Zeilenfrequenz der Kamera auf 2 kHz verdoppelt wird, dann erhöht sich bei der ursprünglichen Umdrehungsgeschwindigkeit die Anzahl der Zeilen auf 2000, folglich sinkt die Bildpunktegröße auf die Hälfte, 0,0314 mm/Pixel.

Man beachte, dass sich dabei die Verhältnisse senkrecht zur Bewegungsrichtung, entlang der Sensorzeile, überhaupt nicht verändert haben. Wenn in der ursprünglichen Konfiguration aufgrund der Auslegung von Optik und Aufnahmeabstand ein 50 mm breiter Streifen der Objektmantelfläche auf die Sensorzeile abgebildet wurde, dann wird auch bei doppelter Umdrehungsgeschwindigkeit oder doppelter Zeilenfrequenz genau diese Länge auf die Sensorzeile projiziert. Da auch diese ihre Beschaffenheit nicht ändert, bleibt die Ortsauflösung in Richtung der Sensorzeile genau gleich.

Es bleibt also festzuhalten, dass die Ortsauflösung von Zeilenkamerabildern in Richtung der Sensorzeile wie bei einer Flächenkamera von Bildfeld und Sensorauflösung bestimmt wird, senkrecht dazu durch das Verhältnis von Zeilenfrequenz und Relativbewegung². Auf die Problematik variierender Bewegungsgeschwindigkeiten gehen wir in 8.4.4 noch ein.

8.4.3 Belichtung bei Zeilenkameras

Neben den oben beschriebenen geometrischen Randbedingungen unterscheiden sich Zeilen- und Flächenkameraaufnahmen auch im Hinblick auf die Belichtung. Bei einer Flächenkamera wird ein Bild oft mehrere Millisekunden lang belichtet. Diese Belichtungszeit steht dann natürlich auch für jede Zeile beziehungsweise jedes Pixel des Bildes zur Verfügung. Bei einer Zeilenkamera hängt die maximale Belichtungszeit für eine Zeile von der Zeilenfrequenz ab. Bei der noch relativ moderaten Frequenz von einem Kilohertz erhält jede Zeile, und damit natürlich auch

² Hierbei ist vorausgesetzt, dass die Relativbewegung exakt im 90°-Winkel zur Sensorzeile erfolgt. Das ist in der Regel auch sinnvoll.

jedes Pixel, nur höchstens eine Millisekunde lang Licht (kürzer belichten kann man selbstverständlich immer). Dies ist deutlich weniger als man normalerweise bei Flächenkameras hat und hat zwei wichtige Konsequenzen:

- Bei einer Zeilenkamera-Aufnahme muss man viel Licht auf das Teil bringen, was in Anbetracht der geometrischen Randbedingung in der Regel schwierig ist. Da die Kamera nur genau eine Zeile aufnimmt, muss die Beleuchtung entsprechend präzise justiert sein, um das Licht auch dorthin zu bringen, wo es benötigt wird.
- Meist wird man mit relativ weit geöffneter Blende arbeiten müssen und, wie wir in 8.5.3 sehen werden, dadurch verringert sich der Schärfentiefebereich. Es ist daher schwierig, Objekte oder Objektteile, die unterschiedlichen Abstand von der Kamera haben, gleichzeitig scharf abzubilden.

Man mag jetzt einwenden, dass auch mit Flächenkameras Bilder mit Belichtungszeiten im Bereich von 0,1–1 ms und weniger aufgenommen werden. Dem steht allerdings entgegen, dass die geometrischen Verhältnisse den Aufbau einer leistungsstarken Beleuchtung bei Flächenkamera-Anwendungen in der Regel einfacher machen als bei Zeilenkamera-Anwendungen. Vor allem aber kann man kurze Aufnahmen mit Flächenkameras mit einer Blitzbeleuchtung vornehmen, die in kurzer Zeit sehr viel Licht auf den Sensor bringen. Bei Aufnahmen mit Zeilenkameras ist dies nicht möglich, denn man müsste ja jede Zeile einzeln blitzen. Dafür wäre eine Blitzbeleuchtung erforderlich, die kontinuierlich mit 1–10 kHz Blitzfrequenz betrieben werden kann und außerdem noch die erforderliche Lichtgeometrie bereitstellt. Dies ist in der Praxis aber nicht durchführbar, so dass grundsätzlich festzuhalten bleibt, dass Zeilenkamera-Anwendungen oft ein Problem hinsichtlich der Lichtausbeute darstellen.

8.4.4 Steuerung von Zeilenkameras

Bei Zeilenkameras ist es häufig nötig den Start der Zeilen extern zu synchronisieren, was als ExSync bezeichnet wird. Dies ist in vielen industriellen Anwendungen erforderlich, aufgrund der in 8.4.2 und 8.4.3 beschriebenen Zusammenhänge zwischen Relativgeschwindigkeit, Zeilenfrequenz und Lichtausbeute. Je nach Art der Zeilensynchronisation kann man die im Folgenden beschriebenen Betriebsarten unterscheiden:

Freilaufend: In diesem Modus nimmt die Kamera in regelmäßiger Folge Zeilen auf und überträgt diese. Die Frequenz, mit der die Zeilen erfasst werden, ist in der Kamera konfiguriert. Dieser Modus ist dem Verhalten einer synchronen Flächenkamera vergleichbar.

Feste ExSync-Frequenz: Bei dieser Methode sendet eine Signalquelle, oft der Frame Grabber selbst, in regelmäßigen Abständen ein ExSync-Signal, um die

Aufnahme und Übertragung einer Bildzeile zu veranlassen. Die Unterschiede zum freilaufenden Betrieb liegen darin, dass

- die Zeilenfrequenz nicht an der Kamera konfiguriert wird, sondern vom Bildverarbeitungssystem über den Framegrabber gesteuert werden kann. Da das Bildverarbeitungssystem ohnehin zur Bildaufnahme mit dem Framegrabber kommunizieren muss, erspart man sich die Unterstützung einer zusätzlichen Schnittstelle und eines Befehlssatzes zur Konfiguration der Kamera.
- die Aufnahme von Zeilen nicht kontinuierlich erfolgen muss, sondern vom Bildverarbeitungssystem gestartet und gestoppt werden kann.

Extern synchronisierte ExSync-Frequenz: Aus dem in 8.4.2 beschriebenen Zusammenhang zwischen Relativbewegung, Zeilenfrequenz und Ortsauflösung folgt, dass eine gleichbleibende Ortsauflösung auch ein gleichbleibendes Verhältnis zwischen Relativbewegung und Zeilenfrequenz erfordert. Die Bewegung eines Werkstückes in der Rotation oder im linearen Vorschub wird in der Regel nicht völlig gleichmäßig sein. In jedem Fall gibt es bei Einzelwerkstücken, die vor der Kamera rotiert werden, eine Anlaufphase, aber auch danach wird die Drehgeschwindigkeit leichten Variationen unterworfen sein. Diese äußern sich dann im digitalen Bild in einer lokalen Veränderung der Bildgeometrie. Um dies auszuschalten, muss die Aufnahme der Bildzeilen mit der Bewegung des Werkstücks synchronisiert werden. In der Regel werden hierzu die ExSync-Signale von einem an die Motorachse gekoppelten Drehgeber erzeugt. Gegebenenfalls werden diese Signale durch einen Frequenzteiler vervielfacht, um höhere Zeilenfrequenzen zu erreichen, als die Auflösung des Drehgebers zulässt.

8.5 Optische Grundlagen

Bisher haben wir uns in diesem Kapitel damit befasst, wie aus der Helligkeitsinformation auf dem Kameratasensor ein digitales Bild entsteht, das sich im Rechner auswerten lässt. Vorher aber muss man dafür sorgen, dass auf dem Sensor überhaupt ein brauchbares Bild entsteht. Hierin besteht die Aufgabe der optischen Bauelemente, d. h. vor allem des Objektivs: Auf dem CCD-Sensor muss ein scharfes, kontrastreiches Bild der prüfenden Bildszene entstehen, dessen Auflösung für die zu prüfenden Details ausreichen muss. Die Auswahl des richtigen Objektivs ist von einer Reihe von Faktoren abhängig, z. B. der Objektgröße, dem einzuhaltenden Abstand, den Beleuchtungsbedingungen, der Bauform der Kamera und Besonderheiten der Prüfaufgabe. Wir werden uns daher in diesem Kapitel zunächst mit einigen optischen Grundlagen befassen, dann mit den Eigenschaften spezieller Objektive.

Wir möchten hierbei ausdrücklich darauf hinweisen, dass es sich um eine vereinfachte Darstellung handelt, die ein Verständnis für die Grundlagen und Eigenschaften von Linsenabbildungen vermitteln soll und sich an die in der Physik übli-

che Notation und Vorgehensweise anlehnt – z. B. durch die Verwendung betragsmäßiger Größen. Für die Berechnung und Konstruktion von Objektiven muss man sich aufwändigerer Verfahren bedienen und den inneren Aufbau der Objektive und die Beziehungen zwischen Material und Geometrie der Linsen in Betracht ziehen. Diese Dinge sind Gegenstand der technischen Optik, auf deren Lehrbücher wir den interessierten Leser hier verweisen.

8.5.1 Blendenzahl

Bevor wir zu den Gesetzmäßigkeiten der optischen Abbildung kommen, müssen wir den Begriff der *Blendenzahl* einführen, der im Folgenden mehrfach benötigt wird.

Abbildung 8.16 zeigt schematisch die Verhältnisse hinter dem Objektiv einer Kamera. Ein Strahlenbündel fällt durch die Blende und beleuchtet den Sensor. Zur ausreichenden Belichtung des Sensors ist eine bestimmte Lichtmenge je Flächeneinheit erforderlich. Die Lichtmenge wird bestimmt von der Belichtungszeit und dem Durchmesser des Strahlenbündels. Die Fläche, auf der sich diese Lichtmenge verteilt, hängt andererseits vom Abstand zwischen dem Objektiv und dem Sensor ab. Wenn unendlich (oder jedenfalls hinreichend) weit entfernte Gegenstände abgebildet werden, ist dieser Abstand gerade die Brennweite f der Linse, wie wir im Abschnitt 8.5.2 sehen werden.

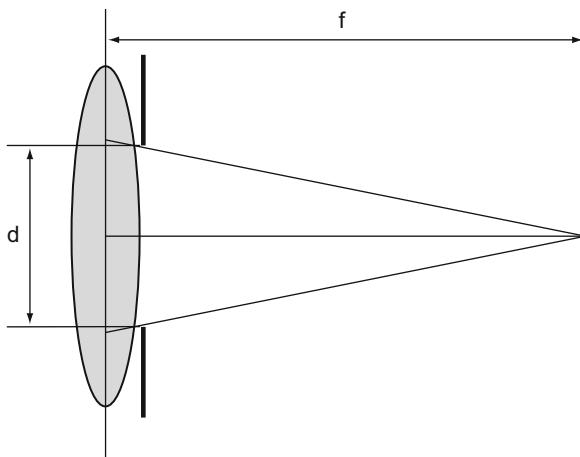


Abbildung 8.16: Relative Öffnung eines Objektivs

Das Verhältnis zwischen Blendendurchmesser d und Brennweite f des Objektivs bezeichnet man als *relative Öffnung*, seinen Kehrwert als *Blendenzahl* k :

$$k = \frac{f}{d}. \quad (8.3)$$

Die Beleuchtungsstärke ist also proportional zum Quadrat der relativen Öffnung und umgekehrt proportional zum Quadrat der Blendenzahl:

$$E \sim \left(\frac{1}{k}\right)^2. \quad (8.4)$$

Das bedeutet, dass bei einer Verdoppelung der Blendenzahl k die Beleuchtungsstärke auf ein Viertel reduziert wird.

Internationale Blendenreihe: Die internationale Blendenreihe besteht aus folgenden Blendenzahlen:

1,0 1,4 2 2,8 4 5,6 8 11 16 22 32

Die einzelnen Werte stehen jeweils in einem Verhältnis von $\sqrt{2}$. Da die Beleuchtungsstärke, wie wir eben sahen, umgekehrt proportional zum Quadrat der Blendenzahl ist, wird von Blendenzahl zu Blendenzahl die Beleuchtungsstärke gerade halbiert. Um dann auf dem Sensor dieselbe Lichtmenge zu erhalten, muss die Belichtungszeit entsprechend verdoppelt werden.

Lichtstärke: Die größte relative Öffnung eines Objektivs bezeichnet man als seine *Lichtstärke*. Sie begrenzt die maximale Lichtmenge, die durch das Objektiv hindurch treten kann, also auch die minimale Beleuchtung, bei der noch Aufnahmen mit diesem Objektiv möglich sind (neben anderen Faktoren, wie z. B. der Empfindlichkeit des Sensors).

8.5.2 Abbildungsgesetze

In diesem Abschnitt werden wir die grundlegenden Gesetze für die Abbildung einer Bildszene auf einem Sensor erläutern – sei es nun ein CCD-Array, ein Film oder die Netzhaut des Auges.

Betrachten wir zunächst den idealisierten Fall einer so genannten *dünnen Linse*, d. h. einer Linse, deren eigene Ausdehnung keine Rolle für die Abbildung spielt. Die Konstruktion des Linsenbildes gehorcht in diesem Idealfall den folgenden Gesetzen nach (Schröder und Treiber 2002), die in Abbildung 8.17 graphisch dargestellt sind:

1. Ein parallel zur optischen Achse einfallender Strahl verläuft hinter der Linse durch den Brennpunkt F.
2. Ein Strahl, der durch den vor der Linse liegenden Brennpunkt F' einfällt, verläuft hinter der Linse parallel zur optischen Achse.
3. Ein Strahl, der durch den Mittelpunkt der Linse verläuft, verändert seine Richtung nicht.

Aus diesen drei Gesetzen ergibt sich, wie Abbildung 8.17 zeigt, dass sich die vom Punkt A des Gegenstandes ausgehenden Strahlen auch in einem einzigen Punkt A'

hinter der Linse wieder schneiden. In dieser Entfernung von der Mittelebene der Linse, der so genannten Bildweite b , wird dieser Punkt wieder als Punkt, also scharf abgebildet. Sowohl davor als auch dahinter hat der Punkt dagegen eine endliche Ausdehnung, wird also zur unscharfen Fläche.

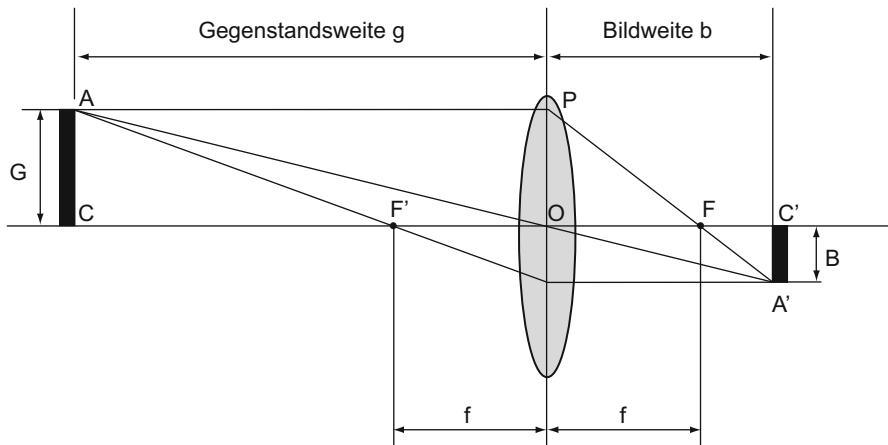


Abbildung 8.17: Konstruktion des Bildes einer dünnen Linse

Abbildungsmaßstab: Aus der Ähnlichkeit der Dreiecke ACO und $A'C'O$ ergibt sich, dass Gegenstandsweite g und Bildweite b im gleichen Verhältnis stehen müssen wie Gegenstandshöhe G und Bildhöhe B . Daraus folgt die Gleichung für den Abbildungsmaßstab, d. h. das Verhältnis von Bildgröße zu Gegenstandsgröße zu:

$$\frac{B}{G} = \frac{b}{g}. \quad (8.5)$$

Entsprechend folgt aus der Ähnlichkeit der Dreiecke POF und $A'C'F$, dass mit der Brennweite f auch folgende Beziehung gelten muss:

$$\frac{B}{PO} = \frac{b-f}{f}. \quad (8.6)$$

Da offensichtlich die Strecke PO und die Gegenstandshöhe G identisch sind, sind die rechten Seiten beider Gleichungen ebenfalls identisch, so dass gilt:

$$\frac{b}{g} = \frac{b-f}{f}. \quad (8.7)$$

Abbildungsgleichung für dünne Linsen: Aus Gleichung (8.7) ergibt sich durch einfache Umformung die Abbildungsgleichung für dünne Linsen:

$$\frac{1}{f} = \frac{1}{g} + \frac{1}{b}. \quad (8.8)$$

Mit Gleichung (8.8) lässt sich die Bildweite unmittelbar aus Gegenstandsweite und Brennweite berechnen:

$$b = \frac{gf}{g-f}. \quad (8.9)$$

Wenn man dies in die Gleichung für den Abbildungsmaßstab (8.5) einsetzt, erhält man bei fester Gegenstandsgröße G die Bildgröße in Abhängigkeit von der Gegenstandsweite und der Brennweite zu:

$$B = \frac{f}{g-f} G. \quad (8.10)$$

Daraus erkennt man zwei Eigenschaften von Linsenabbildungen:

- Eine Vergrößerung der Brennweite führt zu einer Vergrößerung des Bildes. Das ist der Grund für die Länge von Teleobjektiven..
- Wenn man sich dem Gegenstand nähert, also g kleiner wird, wird das Bild ebenfalls größer. Diese unmittelbar einleuchtende Beziehung veranschaulicht Abbildung 8.18. Zusammen mit der Bildgröße wächst wegen Gleichung (8.5) auch die Bildweite, d. h. zur Scharfstellung muss der Abstand zwischen Sensor und Objektiv vergrößert werden. Da der Sensor fest in der Kamera montiert ist, wird dazu der Objektivtubus herausgedreht.

Aus dem Abbildungsgesetz in Gleichung (8.8) folgt weiter, dass ein Gegenstand in unendlichem Abstand genau in der Brennebene scharf abgebildet wird:

$$\begin{aligned} \lim_{g \rightarrow \infty} \frac{1}{b} &= \frac{1}{f} - \lim_{g \rightarrow \infty} \frac{1}{g} \\ &= \frac{1}{f}. \end{aligned} \quad (8.11)$$

Einen interessanten Sonderfall erhält man, wenn die Gegenstandsweite genau doppelt so groß ist wie die Brennweite:

$$b = \frac{gf}{g-f} \text{ für } g = 2f, \quad (8.12)$$

$$b = 2f . \quad (8.13)$$

Man bezeichnet dies als *symmetrische Abbildung*: das Bild ist genauso groß wie der Gegenstand und entsteht im gleichen Abstand von der Linse.

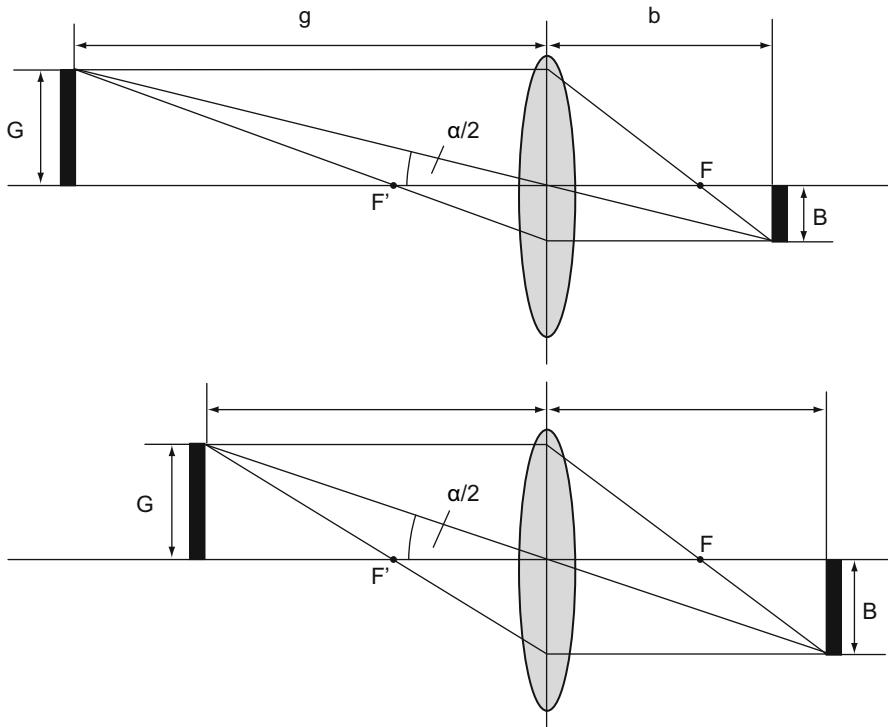


Abbildung 8.18: Gegenstandsweite, Bildweite und Bildgröße

Bildfeld: Die Frage, wie groß das Blickfeld einer Kamera ist, lässt sich mithilfe der Gleichungen (8.5) für den Abbildungsmaßstab und (8.8) für den Zusammenhang mit der Brennweite beantworten. Genau lautet die Frage also: Wie groß kann ein Gegenstand sein, damit er bei der durch den Sensor vorgegebenen maximalen Bildgröße noch abgebildet werden kann?

Aus Gleichung (8.5) folgt für die Größe des G des Gegenstandes abhängig von Gegenstandsweite g , Bildweite b und Bildgröße B :

$$G = \frac{g}{b} B . \quad (8.14)$$

Für die Bildweite b gilt nach der Abbildungsgleichung (8.8):

$$b = \frac{g f}{g - f}. \quad (8.15)$$

womit für die Größe des Gegenstandes bei gegebener Bildgröße B folgt:

$$G = \frac{g - f}{f} B. \quad (8.16)$$

Üblicherweise verwendet man folgende Näherung, mit der man bei praktikablen Arbeitsabständen keinen wesentlichen Fehler macht:

$$G \approx \frac{g}{f} B. \quad (8.17)$$

Tabelle 8.9 führt die Bildfeldgrößen für verschiedene Sensorgrößen bei Verwendung eines Objektivs mit 16 mm Brennweite und einem Objektabstand von 100 mm auf. Zu dieser Tabelle ist noch zu bemerken, dass sie sich, wie alle in diesem Abschnitt dargestellten Beziehungen und die noch folgenden Tabellen, auf den idealen Fall einer einzelnen dünnen Linse bezieht. Die tatsächlichen Werte für die komplizierten Linsensysteme handelsüblicher Objektive können aus konstruktiven Gründen abweichen.

Tabelle 8.9. Bildfeldgrößen in Abhängigkeit von der Sensorgröße

Maß	Sensor		Bildfeld	
	Breite [mm]	Höhe [mm]	Breite [mm]	Höhe [mm]
2/3"	8,8	6,6	55,00	41,25
1/2"	6,4	4,8	40,00	30,00
1/3"	4,8	3,6	30,00	22,50
1/4"	3,6	2,7	22,50	16,88

Objektivformat: 1" (siehe Abschnitt 8.4.6 bezüglich des Zusammenhangs zwischen Objektiv- und Sensorgröße).

Brennweite: 16 mm.

Gegenstandsweite: 100 mm.

Öffnungswinkel: Wie man aus Abbildung 8.18 entnehmen kann, gilt für den Winkel zwischen der optischen Achse und dem Randstrahl zum äußersten Punkt des Gegenstandes G mit der Näherung aus Gleichung (8.17):

$$\frac{G}{g} = \tan\left(\frac{\alpha}{2}\right)$$

$$\alpha = 2 \arctan\left(\frac{G}{g}\right)$$

$$\alpha = 2 \arctan\left(\frac{B}{f}\right).$$

Der maximale Öffnungswinkel wird erreicht, wenn das Bild des Gegenstandes den Sensor vollständig ausfüllt. In diesem Fall ist die Bildgröße B entsprechend der Darstellung in Abbildung 8.17 gleich der halben Seitenlänge des Sensors: $B = S/2$. Für den Öffnungswinkel ergibt sich dann in Abhängigkeit von Sensorgroße und Brennweite:

$$\alpha = 2 \arctan\left(\frac{S}{2f}\right). \quad (8.18)$$

Tabelle 8.10 listet die Öffnungswinkel für einige typische Brennweiten und Sensorgrößen auf. Die Öffnungswinkel sind dabei auf die längere Seite der Sensoren bezogen (typische Sensorgrößen enthält Tabelle 8.1). Wir erwähnten schon bei Tabelle 8.9, dass die tatsächlichen Werte handelsüblicher Objektive von den hier für den Idealfall einer dünnen Linse berechneten aus konstruktiven Gründen abweichen können. Die in Klammern hinzugefügten Angaben sind (Pentax 2008) entnommen und zeigen, dass diese Abweichungen in der Regel sehr gering sind.

Tabelle 8.10. Öffnungswinkel in Abhängigkeit von Brennweite und Sensorgroße

f	Sensorgroße			
	1/4"	1/3"	1/2"	2/3"
8,5 mm	23,91 (24,02)	31,53 (31,87)	41,26 (42,09)	54,74 (56,49)
12,5 mm	16,39 (16,53)	21,74 (21,97)	28,72 (29,12)	38,78 (39,56)
16 mm	12,84 (12,70)	17,06 (16,91)	22,62 (22,48)	30,75 (30,72)
25 mm	8,24 (8,23)	10,97 (10,95)	14,59 (14,55)	19,96 (19,87)
50 mm	4,12 (4,13)	5,50 (5,48)	7,32 (7,32)	10,06 (10,03)
75 mm	2,75 (2,75)	3,67 (3,65)	4,89 (4,88)	6,72 (6,57)

Alle Angaben in °.

Angaben in Klammern für handelsübliche Objektive nach (Pentax 2008).

Dicke Linsen: Reale Objektive bestehen zur Korrektur von Abbildungsfehlern aus einem ganzen System von Linsen. Die Idealisierung einer im Verhältnis zu ihrer Brennweite sehr dünnen Linse lässt sich daher nicht aufrechterhalten. Am Prinzip der Abbildungsgesetze ändert sich dabei nichts, jedoch muss man alle Abstände auf die Hauptebenen der Linse beziehen, wie Abbildung 8.19 zeigt. In der

praktischen Anwendung verwendet man einfach die für dünne Linsen hergeleiteten Beziehungen mit den entsprechend angegebenen Kennwerten der Objektive.

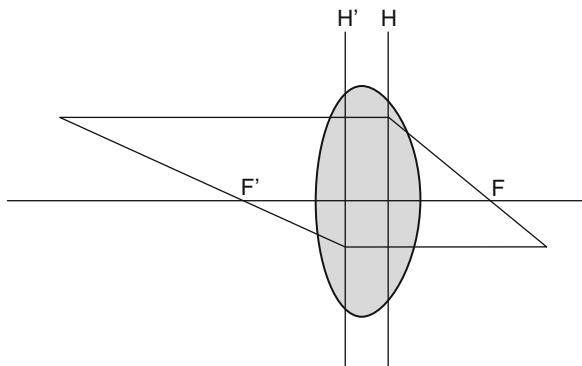


Abbildung 8.19: Hauptebenen einer dicken Linse

8.5.3 Schärfentiefe

Die Abbildungsgleichung (8.9) stellt einen exakten Zusammenhang zwischen Brennweite, Gegenstandsweite und Bildweite her. Bei einem bestimmten Abstand zwischen Linse und Sensor ist die Bildweite gegeben, womit sich die Gegenstandsweite exakt berechnen lässt. Bei mathematisch idealem Verhalten von Linse und Lichtstrahlen werden nur Objekte in genau dieser Entfernung mathematisch exakt scharf abgebildet. Praktisch gilt jedoch, dass innerhalb eines bestimmten Bereiches keine *merkliche* Unschärfe auftritt. Eine Unschärfe äußert sich dadurch, dass ein punktförmiges Objekt nicht als Punkt, sondern als so genannter *Zerstreuungskreis* abgebildet wird. Abbildung 8.20 zeigt die Verhältnisse in der Kamera für folgende Voraussetzungen:

- Ein Objekt im Abstand g_0 vom Objektiv wird in einer Bildebene im Abstand b_0 hinter der Linse scharf abgebildet.
- Ein näher am Objektiv liegendes Objekt im Abstand g_1 würde im Abstand b_1 hinter der Linse scharf abgebildet.
- Da die Randstrahlen eines Punkts des zweiten Objekts die Bildebene zu früh schneiden, entsteht aus dem Punkt dort ein Zerstreuungskreis mit dem Durchmesser C .

Solange dieser Zerstreuungskreis kleiner ist als ein einzelnes Sensorelement, wird trotz der Auffaserung nur ein einziges Element belichtet, der Punkt bleibt also auch auf dem Sensor punktförmig, er erstreckt sich nicht über mehrere Bildelemente. Deshalb lassen sich Objekte innerhalb eines gewissen Entfernungsbereiches scharf abbilden. Diesen Bereich bezeichnet man als *Schärfentiefe*.

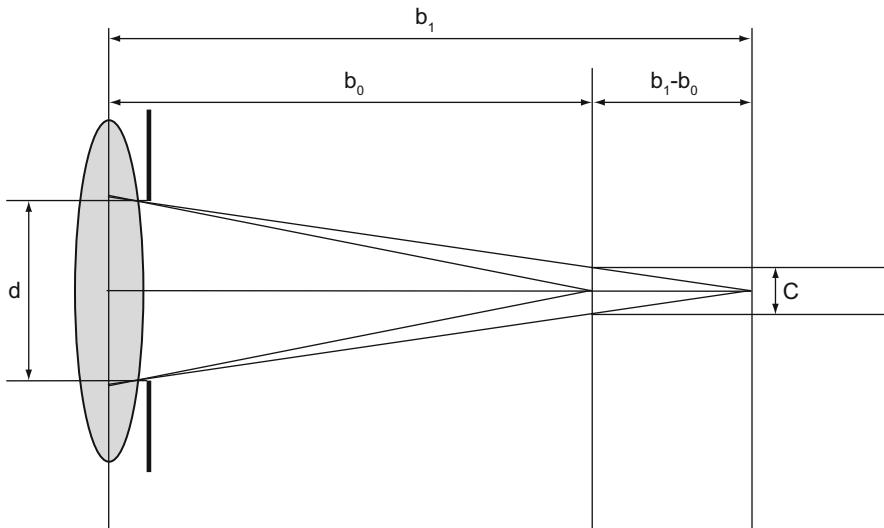


Abbildung 8.20: Entstehung eines Zerstreuungskreises

Aus Abbildung 8.20 geht außerdem hervor, dass eine Verkleinerung der Blende die Schärfentiefe verbessert. Wenn die hinter der Linse angedeutete Blende verengt wird, also der Durchmesser d kleiner wird, wird auch der Öffnungswinkel der Strahlenbündel kleiner. Damit wird auch der Zerstreuungskreis kleiner. Wenn der Zerstreuungskreis zuvor etwas größer war als ein Sensorspalt, kann man ihn durch Verringern der Blendenöffnung auf einen Bildpunkt fokussieren. Gute Ausleuchtung verbessert also die Erkennbarkeit von Objekten nicht nur durch besseren Kontrast, sondern auch durch verbesserte Bildschärfe, da die Blende weiter geschlossen werden kann. Aus demselben Grund machen sich Fehlsichtigkeiten beim Menschen um so stärker bemerkbar, je dunkler es wird: Die Pupille kann nicht mehr weit genug verengt werden, um ausreichende Schärfentiefe zu erreichen.

Ob ein Punkt eines Objekts scharf abgebildet wird, hängt nach Abbildung 8.18 von folgenden Faktoren ab:

- Blendenöffnung
- Bildweite b_0 des exakt fokussierten Gegenstandes
- Bildweite b_1 des weniger weit (oder weiter) entfernten Gegenstandes
- Zulässiger Durchmesser C des Zerstreuungskreises

Die letzte Größe, C , hängt vom verwendeten Sensor ab und ist bis zu einem gewissen Grade willkürlich. Eine typische Vorgabe ist die halbe Größe eines Sensorspaltes, denn wenn ein Punkt der Bildszene auf dem Sensor keine größere Ausdehnung hat, kann er bei einem Interline-Transfer-Sensor niemals zwei Bildelemente gleichzeitig belichten, d. h. es kann keine wahrnehmbare Unschärfe ent-

stehen. Über die Berechnung der Schärfentiefe herrscht in der Literatur einige Verwirrung, daher geben wir in Anhang A.2 eine vollständige Herleitung an. Die für die praktische Arbeit wichtigen Ergebnisse sind im Folgenden zusammengefasst.

Berechnung der Schärfentiefe: Ab einem Nennabstand größer als g_∞ werden alle Objekte scharf in der Brennebene abgebildet, bei kleineren Nennabständen ergibt die Näherung nach Gleichung (8.19) keinen wesentlichen Fehler.

$$\Delta g \approx \frac{2Ckg_0(g_0 - f)}{f^2} \quad \text{für } g_0 \ll \infty, \quad (8.19)$$

$$g_\infty > \frac{f^2}{Ck}. \quad (8.20)$$

Darin sind:

- g_0 Nennabstand, auf den das Objektiv fokussiert ist.
- g_∞ Nennabstand, ab dem alle weiter entfernten Objekte scharf erscheinen.
- Δg Größe des Schärfentiefbereiches
- f Brennweite des Objektivs
- k Blendenzahl
- C zulässiger Streukreisdurchmesser, abhängig vom Auflösungsvermögen des Sensors; Bei CCD-Sensoren typischerweise die halbe Größe eines Sensorselements.

Abhängigkeit der Schärfentiefe von der Blendeneinstellung: Man sieht in der Gleichung (A.25) für den vorderen Grenzabstand g_{\min} , dass dieser mit zunehmender Blendenzahl k kleiner wird, d. h. der vorderste Punkt, an dem ein Gegenstand noch scharf abgebildet wird, rückt näher an das Objektiv heran. Umgekehrt wird g_{\max} mit zunehmender Blendenzahl größer, der fernste scharf abgebildete Punkt entfernt sich also vom Objektiv. Mit zunehmender Blendenzahl wird also der gesamte Schärfentiefbereich größer. Das geht auch aus Abbildung 8.20 hervor: Wenn der Blendendurchmesser verringert wird, verengen sich die Strahlenbündel, folglich wird auch der Zerstreuungskreis kleiner. Anders ausgedrückt: Je besser die Bildszene ausgeleuchtet ist, desto größer ist der scharf abgebildete Bereich, da die Blende weiter geschlossen werden kann.

Tabelle 8.10 zeigt die vorderen und hinteren Grenzabstände und die Größe des Schärfentiefbereiches für ein Objektiv mit einer Brennweite von acht Millimeter bei einem Objektabstand von 100 mm und einer zulässigen Unschärfe von fünf Mikrometer für alle Einstellungen der internationalen Blendenserie. Das Anwachsen des Schärfentiefbereichs mit zunehmender Blendenzahl ist deutlich zu sehen.

Tabelle 8.10. Schärfentiefe in Abhängigkeit von der Blendeneinstellung

Blende	g_{\min} [mm]	g_{\max} [mm]	Schärfentiefe [mm]
1,00	99,29	100,72	1,44
1,40	99,00	101,02	2,01
2,00	98,58	101,46	2,88
2,80	98,03	102,05	4,03
4,00	97,00	102,96	5,75
5,60	96,13	104,19	8,06
8,00	94,56	106,10	11,54
11,00	92,00	108,59	15,91
16,00	89,69	112,99	23,31
22,00	86,35	118,78	32,44
32,00	81,30	129,87	48,57

Brennweite: 8 mm

Eingestellter Objektabstand: 100 mm

Zulässige Unscharfe: 5 μ m

Abhängigkeit der Schärfentiefe von der Brennweite: Man kann leicht zeigen, dass bei einer Vergrößerung der Brennweite der vordere Grenzabstand zunimmt, gleichzeitig der hintere Grenzabstand abnimmt (siehe Anhang A.2.3). Die Grenzpunkte, an denen noch scharf abgebildet wird, rücken also mit zunehmender Brennweite dichter zusammen, die Schärfentiefe wird kleiner. Man kann das gut bei Teleobjektiven beobachten, die zwar sehr entfernte Gegenstände abbilden können, aber nur eine sehr geringe Schärfentiefe aufweisen. Tabelle 8.11 zeigt die Schärfentiefen für einige in der digitalen Bildverarbeitung typische Brennweiten. Während das 2,6-mm-Objektiv bei den eingestellten Werten bis ins Unendliche scharf abbildet, misst der Schärfentiefebereich beim 75-mm-Objektiv nur noch Zehntelmillimeter.

Tabelle 8.11. Schärfentiefe in Abhängigkeit von der Brennweite

Brennweite	g_{\min} [mm]	g_{\max} [mm]	Schärfentiefe [mm]
2,60	46,45	∞	∞
8,00	89,69	112,99	23,31
12,00	95,34	105,14	9,80
16,00	97,44	102,70	5,25
25,00	99,05	100,97	1,92
50,00	99,84	100,16	0,32
75,00	99,96	100,04	0,07

Blendenzahl: 16

Eingestellter Objektabstand: 100 mm

Zulässige Unscharfe: 5 μ m

Abhängigkeit der Schärfentiefe vom Abstand: Wenn der Nennabstand g_0 zunimmt, wachsen beide Grenzabstände, der hintere jedoch schneller als der vordere, wie man leicht durch Einsetzen in die entsprechenden Gleichungen nachprüfen kann. Der Schärfentiefenbereich wird also mit zunehmendem Abstand immer größer. Tabelle 8.12 zeigt dies anhand eines 12-mm-Objektivs. Diese Vergrößerung des Schärfenbereichs mit wachsendem Nennabstand ermöglicht die beim Fotoapparat wohlbekannte ∞ -Einstellung.

Tabelle 8.12. Schärfentiefe in Abhängigkeit vom Nennabstand

Nennabstand	g_{\min} [mm]	g_{\max} [mm]	Schärfentiefe [mm]
100,00	96,75	103,48	6,73
200,00	186,60	215,47	28,87
500,00	421,45	614,54	193,10
1000,00	726,03	1606,07	880,04
2000,00	1136,81	8309,29	7172,48
5000,00	1721,09	∞	∞

Blendenzahl: 11

Brennweite: 12 mm

Zulässige Unscharfe: 5 μm

8.5.4 Typische Aufnahmesituationen

Anhand des Abstandes zwischen Objekt und Kamera bzw. Objektiv kann man drei typische Aufnahmesituationen unterscheiden:

- Fernaufnahmen
- Makroaufnahmen
- Mikroaufnahmen

Fernaufnahmen: Von einer Fernaufnahme spricht man dann, wenn die Gegenstandsweite erheblich größer ist als die Brennweite der Linse. In diesem Fall sind die Bildweite und die Brennweite ungefähr gleich groß. Das ist die typische Situation bei Foto- und Filmaufnahmen und auch in vielen Bildverarbeitungsanwendungen, vor allem aus dem Bereich der Anwesenheitskontrolle, bei denen es darauf ankommt, eine Gesamtprüfung großer Teile vorzunehmen, nicht jedoch auf besondere Messgenauigkeit. Wegen der vergleichsweise geringen Brennweite hat man hier einen großen Schärfentiefenbereich, so dass die Positionierung weniger kritisch ist.

Beispiel: Bei einem Objektiv mit Brennweite $f = 16$ mm, einer Blende $k = 11$ und einer Gegenstandsweite $g_0 = 1000$ mm umfasst der Schärfentiefenbereich nach Gleichung (8.19) für eine zulässige Unscharfe von fünf Mikrometer mehr als 440 mm.

Makroaufnahmen: Kennzeichnend für Makroaufnahmen ist, dass die Bildweite etwa der Gegenstandsweite entspricht, also gemäß der Gleichung (8.13) für symmetrische Abbildungen ungefähr bei der doppelten Brennweite. Wegen der Abnahme der Schärfentiefe mit abnehmender Gegenstandsweite ist klar, dass nur noch ein sehr kleiner Bereich scharf abgebildet werden kann. Außerdem nähert man sich hier der Grenze, bei der mit normalen Kameraobjekten scharfe Abbildung überhaupt möglich sind.

Beispiel: Wenn wir im obigen Beispiel für die Fernaufnahmen nur die Gegenstandsweite verändern, also weiterhin mit einer Brennweite $f = 16$ mm, einer Blende $k = 16$ und einer zulässigen Unscharfe von fünf Mikrometer arbeiten, erhalten wir für den symmetrischen Fall einer Gegenstandsweite $g_0 = 2f = 32$ mm einen Schärfentiefenbereich von gerade 0,22 mm.

Mikroaufnahmen: Bei Mikroaufnahmen ist die Bildweite deutlich größer als die Brennweite und damit die Gegenstandsweite nahezu gleich der Brennweite. Dies ist die typische Aufnahmesituation in einem Mikroskop, bei der der Schärfentiefenbereich weiter verengt wird.

Die Gegenstandsweite hängt nach Gleichung (8.5) mit der Bildweite über die Vergrößerung zusammen:

$$g = \frac{B}{G} b . \quad (8.21)$$

Wenn man eine 50-fache Vergrößerung ($B = 50G$) annimmt, dann ist folglich $g = 50b$. Da außerdem nach der Abbildungsgleichung (8.8) gilt

$$g = \frac{bf}{b-f} . \quad (8.22)$$

kann man berechnen, dass die Gegenstandsweite in diesem Fall $1,02f$ betragen muss. Bei ansonsten unveränderten Daten ($f = 16$ mm, $k = 11$, $C = 5$ µm) erhält man einen Schärfentiefenbereich von nur zwei Mikrometer. Wenn man berücksichtigt, dass derartige Aufnahmen meist eine voll geöffnete Blende ($k = 1$) verlangen, um die erforderliche Lichtausbeute zu erzielen, verringert sich dies sogar auf 0,2 µm. Man sieht also, dass Mikroskope tatsächlich nur in einer ganz bestimmten Entfernung scharf abbilden können.

Es sei angemerkt, dass sich derartige Vergrößerungen natürlich nicht mit einem normalen Kameraobjektiv erreichen lassen, die enorme Bildweite würde eine Scharfstellung unmöglich machen.

8.5.5 Abbildungsfehler

Bei einfachen Linsen sind eine Reihe von Abbildungsfehlern zu beobachten. Wir erwähnen sie hier der Vollständigkeit halber, bei modernen Linsensystemen spielen sie jedoch in der Praxis, außer bei Spezialanwendungen, keine Rolle mehr.

Chromatische Aberration: Licht unterschiedlicher Wellenlänge wird unterschiedlich stark gebrochen, daher sind die Brennweiten für verschiedene Farben nicht identisch. Theoretisch ist also bei einer einfachen Linse stets nur Licht einer einzigen Farbe exakt fokussiert, so dass sich Farbränder bilden. Auch korrigierte Linsensysteme (Achromate, Apochromate) arbeiten nicht im gesamten Frequenzbereich ohne Verfälschung, sondern nur für wenige Wellenlängen. Für hochgenaue Messaufgaben ist daher monochromatisches Licht ideal.

Dieser Abbildungsfehler kann um so besser beobachtet werden, je größer die Wellenlängenunterschiede sind. Dabei kann es zu interessanten Effekten kommen. Wie wir in Abschnitt 8.1.6 sahen, sind CCD-Sensoren für nahes Infrarot empfindlich. Abbildung 8.21 zeigt einen laserbeschrifteten IC, links unter einem Fluoreszenz-Ringlicht, rechts zusätzlich mit Infrarot-LEDs beleuchtet. Ohne dass die Kameraeinstellungen geändert wurden, ist das rechte Bild schärfer als das linke. Die Wellenlänge der Infrarot-LEDs liegt mit etwa 950 nm deutlich über der des Ringlichts von etwa 600 nm. Offensichtlich fokussiert das Objektiv mit der dort gewählten Einstellung in diesem Wellenlängenbereich besser. Da Objektive gewöhnlich für Abbildungen mit sichtbarem Licht entworfen und eingestellt werden, ist der umgekehrte Effekt häufiger, bei dem unkontrollierte Infraroteinstrahlungen zu einer Unschärfe führt. Mit dem bloßen Auge ist selbstverständlich kein Unterschied sichtbar, da wir Infrarot ja nicht wahrnehmen können.

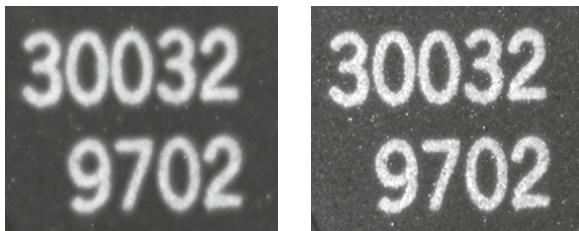


Abbildung 8.21: Veränderung der Bildschärfe durch Infrarotbeleuchtung

Sphärische Fehler: Strahlen, die weit von der optischen Achse entfernt sind, treffen in flacherem Winkel auf die Oberfläche der Linse, so dass sie stärker gebrochen werden. Diesem Fehler kann durch Abblenden begegnet werden, da nur noch achsennähere Lichtstrahlen durch die engere Blende hindurch treten können.

Bildfeldwölbung: Da das Bild auf einer gewölbten Fläche erzeugt wird, erscheinen die Bildränder unscharf.

Verzeichnung: Bei der Abbildung eines rechteckigen Gitters werden die Geraden nach außen oder innen gekrümmt, und zwar um so stärker, je weiter sie von der optischen Achse entfernt sind. Ursache hierfür sind unterschiedliche Abbildungsmaßstäbe für verschiedene Bildwinkel. Typische Verzeichnungswerte liegen zwischen 0,2 % und 1,0 %; es können also Abweichungen von bis zu fünf Bildpunkten auftreten.

8.5.6 Objektivbestimmung

Wir haben bisher in diesem Abschnitt eine Menge über Abbildungsgesetze gesagt, wovon wir glauben, dass es zum Verständnis der Eigenschaften eines optischen Systems sehr wichtig ist. In der praktischen Anwendung stößt man jedoch noch auf eine Reihe weiterer Begriffe. Die Gebrauchsanweisungen und Kataloge der Hersteller erläutern den Umgang mit Objektiven ausführlicher, als wir es hier tun können, aber einen kurzen Einblick wollen wir dennoch versuchen.

Auflagemaß: Unter dem Auflagemaß versteht man den Abstand von der Objektivauflage an der Kamera bis zur Sensorebene. Das Auflagemaß wird für Kamera und Objektive angegeben. Im Allgemeinen werden heute C-Mount-Kameras und C-Mount-Objektive verwendet, wobei das C-Mount-Auflagemaß auf 17,526 mm festgelegt ist. Ebenfalls genormt ist das CS-Mount-Auflagemaß mit 12,526 mm, allerdings gibt es heute kaum noch CS-Mount-Kameras oder CS-Mount-Objektive.

Von der Kameraseite her gesehen, bedeutet das Auflagemaß die kürzeste Entfernung, die zwischen Linse und Sensor überhaupt eingestellt werden kann, da das Objektiv in dieser Entfernung aufliegt. Es handelt sich dabei um eine konstruktive Eigenschaft der Kamera, so dass sich daran nichts ändern lässt.

Für das Objektiv bedeutet das Auflagemaß dagegen die größte Entfernung, in der sich der Sensor befinden darf, damit ein unendlich ferner Gegenstand scharf abgebildet werden kann. Wie wir aus Abschnitt 8.5.2 wissen, hat ein Linsensystem dann seine kürzeste Bildweite, wenn der abzubildende Gegenstand unendlich weit entfernt ist. Für jede geringere Entfernung muss die Linse weiter vom Sensor entfernt werden. Die geringst mögliche Entfernung zwischen Linse und Sensor ist durch das Auflagemaß gegeben.

Für ein C-Mount-Objektiv gilt demnach, dass der Sensor bei Einstellung unendlich mindestens 17,526 mm von der Objektivauflage entfernt sein muss. Entsprechend muss für ein CS-Mount-Objektiv dieser Abstand 12,526 mm betragen. Daraus folgt, dass sich bei einer C-Mount-Kamera keine CS-Mount-Objektive verwenden lassen, da der konstruktiv bedingte Mindestabstand zwischen Objektivauflage und Sensor 17,526 mm beträgt, so dass die Linse nicht dicht genug an den Sensor herangebracht werden kann.

Umgekehrt kann man bei einer CS-Mount-Kamera sowohl C- als auch CS-Mount-Objektive verwenden, nur muss man bei einem C-Mount-Objektiv einen 5 mm starken Zwischenring aufschrauben, um das für dieses Objektiv erforderliche

Auflagemaß von 17,526 mm zu erreichen. Abbildung 8.22 stellt diese Zusammenhänge noch einmal graphisch dar.

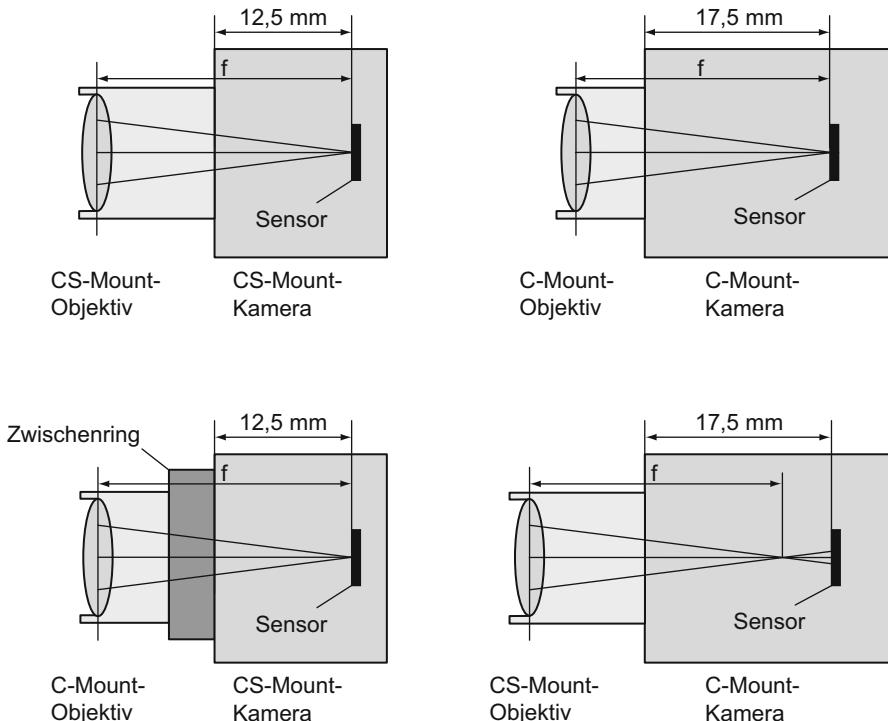


Abbildung 8.22: Kombination von C-Mount- und CS-Mount-Auflagemaßen

Nahaufnahmen: Wie Abbildung 8.18 zeigt, vergrößert sich die Bildweite, wenn man sich einem Objekt nähert. Der Gegenstand wird in größerer Entfernung hinter der Linse scharf abgebildet, daher muss die Entfernung zwischen Objektiv und Sensor für Nahaufnahmen vergrößert werden. Beim Scharfstellen eines nahen Gegenstandes wird dazu der Objektivtubus herausgedreht. Diese Bewegung hat einen Anschlag, der die geringste Gegenstandsweite definiert, auf die ein Objektiv noch fokussiert werden kann³. Um näher an das Objekt heranzukommen, muss also der Abstand zwischen Objektiv und Sensor vergrößert werden. Dazu kann man Zwischenringe benutzen, die in verschiedenen Stärken erhältlich sind. Typische Größen sind beispielsweise ein, zwei, fünf, zehn Millimeter, usw.

Beim Einsetzen eines Zwischenrings muss man berücksichtigen, dass der Schärfentiefenbereich verkleinert wird. Weit entfernte Gegenstände können jetzt nicht mehr scharf abgebildet werden. Der Minimalabstand zwischen Objektiv und

³ Davon unberührt ist natürlich, dass Gegenstände innerhalb der Brennweite überhaupt nicht fokussiert werden können, da ihre Randstrahlen nach der Linse divergieren. Die Bildweite nach Gleichung (8.8) würde dann negativ.

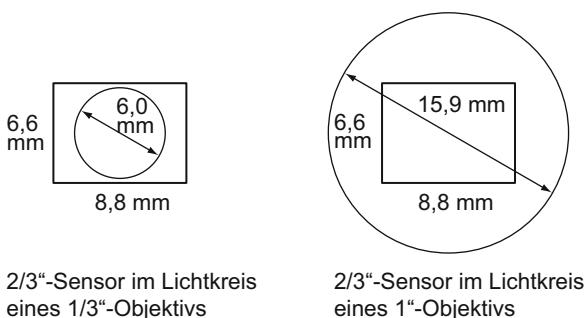
Sensor ist normalerweise genauso groß wie die Brennweite des Objektivs um unendlich ferne Gegenstände scharf abbilden zu können. Wenn bei einem 16-mm-Objektiv ein Zwischenring von fünf Millimeter Stärke eingesetzt wird, wächst dieser minimale Abstand auf 21 mm. Kleinere Bildweiten sind dann nicht mehr möglich. Aus der Abbildungsgleichung (8.8) folgt für die Gegenstandsweite g mit Bildweite b und Brennweite f :

$$\frac{gf}{g-f} = b \geq b^*, \quad (8.23)$$

$$g \leq \frac{fb^*}{b^* - f}. \quad (8.24)$$

und mit $f = 16$ mm und $b^* = 21$ mm ist $g = 67,2$ mm. Auf weiter entfernte Gegenstände lässt sich das Objektiv dann nicht mehr fokussieren.

Zusammenhang zwischen Objektivgröße und Optischem Format: In Abschnitt 8.1.6 war bereits die Rede davon, dass es CCD- und CMOS-Sensoren unterschiedlicher Größe gibt und dass die Größen von Sensor und Objektiv aufeinander abgestimmt sein müssen. Dabei wird nicht die absolute Größe angegeben, sondern eine Größenstufe. Die Größenangabe ein Zoll bei einem Objektiv besagt, dass es einen Lichtkreis ausleuchtet, in dem ein ein Zoll Sensor Platz findet. Entsprechend leuchtet eine 1/3"-Objektiv einen Lichtkreis auf, in den gerade ein 1/3"-Sensor hineinpasst. Daraus folgt, dass man Objektive mit Sensoren kleinerer Größenangabe verwenden kann, nicht aber umgekehrt, da das Objektiv den Sensor nicht vollständig ausleuchten würde. Abbildung 8.23 veranschaulicht diesen Zusammenhang.



2/3"-Sensor im Lichtkreis eines 1/3"-Objektivs

2/3"-Sensor im Lichtkreis eines 1"-Objektivs

Abbildung 8.23: Zusammenhang zwischen Sensor- und Objektivformat

8.5.7 Besondere Objektivtypen

In diesem Abschnitt wollen wir noch einige spezielle Objektivtypen erwähnen, die aufgrund ihrer besonderen Eigenschaften für bestimmte Anwendungen bevorzugt verwendet werden.

Telezentrische Objektive: Dieser Objektivtyp wird vor allem in Vermessungsanwendungen verwendet. Er zeichnet sich dadurch aus, dass innerhalb des so genannten Telezentrie-Bereiches eine Änderung des Abstands zwischen Gegenstand und Objektiv keine Größenänderung des Bildes zur Folge hat. Das ist nur möglich, wenn das Objektiv auf der Gegenstandsseite einen parallelen Strahlengang aufweist. In diesem Fall gehen alle Strahlen der Abbildung stets durch den Brennpunkt. Man erreicht diesen telezentrischen Strahlengang dadurch, dass man die Austrittspupille in die sensorseitige Brennebene des Objektivs legt. Außerdem muss man noch voraussetzen, dass der Gegenstand nicht größer sein darf als das Objektiv. Andernfalls können die Randstrahlen entsprechend Abbildung 8.17 nicht parallel einfallen und ändern ihren Winkel je nach Entfernung des Gegenstandes. Telezentrische Objektive können daher nur für relativ kleine Objekte gebaut werden. Bei einer Gegenstandsgröße von über 100 mm erreichen solche Objektive eine Länge von über einem halben Meter und eine Masse von etwa vier Kilogramm.

Superweitwinkel-Objektive: Superweitwinkel-Objektive werden auch als Fischauge-Objektive bezeichnet. Sie verfügen aufgrund ihrer Konstruktion über einen extrem großen Öffnungswinkel von mehr als 90° . Dadurch sind 360° -Aufnahmen von Gegenständen möglich. Die Abbildung einer solchen Linse weist eine extreme kissenförmige Verzeichnung auf, genaue Maße lassen sich daraus nicht entnehmen. Für die Anwesenheitskontrolle im Inneren rotationssymmetrischer Teile sind solche Aufnahmen jedoch gut geeignet, wie Abbildung 8.24 am Beispiel eines Rollenlagers zeigt.

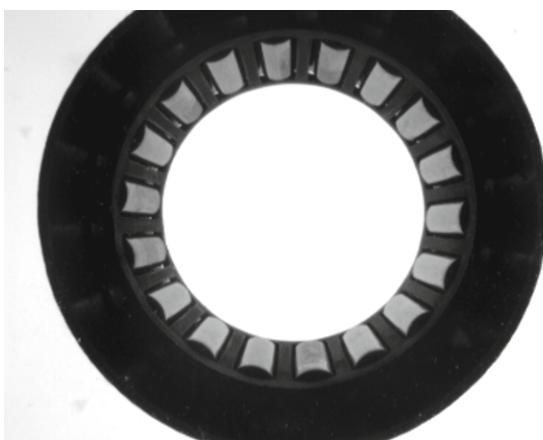


Abbildung 8.24: Aufnahme eines Rollenlagers mit einem Superweitwinkel-Objektiv

Endoskope: Bei dem Stichwort „Endoskopie“ denken die meisten Menschen wahrscheinlich zunächst an medizinische Anwendungen. In der Medizin werden Endoskope für die Untersuchung von Körperhöhlen verwendet. Entsprechend kann man Endoskope auch in technischen Anwendungen zur Inspektion unzugänglicher Bereiche eines Werkstückes benutzen. (Pschyrembel 2007) beschreibt ein Endoskop als röhrenförmige Instrument mit einem optischen System aus Objektiv und Okular und einer Beleuchtungseinrichtung, die meist auf der Einspiegelung von Kaltlicht beruht. Am häufigsten werden Glasfaserendoskope verwendet. In den gebündelten Glasfasern werden die Lichtstrahlen durch vielfache Totalreflexion am Inneren der Fasern zunächst von der Beleuchtung zum Objekt geleitet, dann vom Objekt zurück zum Sensor.

Spezialoptiken: Es gibt eine Vielzahl von Spezialoptiken für besondere Zwecke. Da man solche Optiken nicht nur per Katalog kaufen, sondern sich auch für besondere Zwecke von verschiedenen Herstellern konstruieren und fertigen lassen kann, ist es unmöglich, eine auch nur annähernd umfassende Übersicht zu geben. Wir belassen es darum bei einigen Hinweisen auf die vielfältigen Möglichkeiten solcher Optiken. Ein Beispiel sind Zweimessstellenobjektive, die Bilder weit auseinander liegender Objektpunkte in eine Linse geringeren Durchmessers einspielen. Dadurch lässt sich die effektive Bildauflösung für Messzwecke erheblich vergrößern, da der gesamte Bereich des Prüfteils zwischen den beiden Messstellen nicht im Bild erscheint. Andere Beispiele sind Prismenköpfe für die Inspektion von Innenbohrungen oder konische Spiegel zur Aufnahme der Außenwand zylindrischer Objekte mit einer einzelnen Matrixkamera.

8.6 Beleuchtungstechnik

Der Beleuchtungsaufbau für eine konkrete Anwendung wird im Allgemeinen individuell auf der Basis von Experimenten erfolgen müssen. Es ist daher kaum möglich, allgemeine Richtlinien hierfür aufzustellen. Das experimentelle Herangehen wird aber durch Grundkenntnisse verschiedener Beleuchtungsverfahren erheblich erleichtert. In den Anwendungsbeispielen der übrigen Kapitel sind wir deshalb auch immer wieder auf die dort verwendeten Beleuchtungstechniken eingegangen. Diese Bemerkungen wollen wir hier durch eine Übersicht über typische Beleuchtungsverfahren ergänzen.

8.6.1 Leuchtmittel

In Anbetracht der Vielfalt möglicher Beleuchtungskonfigurationen ist es fast etwas überraschend, dass nur wenige verschiedene Lichtquellen verbreitet zum Einsatz kommen. Gewöhnliche Glühlampen werden wegen ihrer hohen Wärmeentwicklung bei gleichzeitig geringer Lichtausbeute und der wechselstrombedingten

Helligkeitsschwankungen praktisch nicht verwendet. Damit verbleiben als wichtige Lichtquellen:

- Leuchtdioden
- Laser, meist als Laserleuchtdioden
- Leuchtstoffröhren
- Halogenlampen

Im Folgenden gehen wir kurz auf die Eigenschaften und typischen Verwendungszwecke dieser Lichtquellen ein.

Leuchtdioden: Leuchtdioden (englisch Light Emitting Diode, abgekürzt LED) sind heute die häufigste Lichtquelle in der industriellen Bildverarbeitung. Beleuchtungen mit Leuchtdioden bestehen meist aus vielen LEDs, die beliebig angeordnet werden können und damit eine sehr große geometrische Variationen beim Aufbau von Beleuchtungen erlauben. Sie lassen sich in Zeilen als Linienlichter oder Balkenbeleuchtungen anordnen, in Rechtecken als Leuchtfelder, als Ringlichter fast beliebiger Größe und als Lichtquelle in Dom-Beleuchtungen. Für punktuelle Beleuchtungen stehen LED-Spots zur Verfügung, außerdem kann das Licht von LEDs auch in Glasfaserlichtleiter eingekoppelt werden. Im Gegensatz zu Glühlampen oder Leuchtstoffröhren senden LEDs ihr Licht in einem verhältnismäßig engen Lichtkegel aus. Bei gerichteter Beleuchtung muss man deshalb auf die Ausrichtung der LEDs achten.

Durch die Möglichkeit, jede LED einzeln anzusteuern, kann man geschaltete Beleuchtungen aufbauen, durch die sich beispielsweise ein Objekt automatisch aus mehreren Richtung nacheinander beleuchten lässt. Durch den Vergleich von Bildern, die unter diesen verschiedenen Beleuchtungsrichtungen aufgenommen wurden, können Oberflächendefekte festgestellt werden.

LEDs leuchten in schmalen Wellenlängenbändern, in Rot, Grün, Blau Ultraviolet oder Infrarot. Die schmalen Wellenlängenbänder erlauben es, Fremdlichteinflüsse mit anderen Wellenlängen durch optische Filter abzuschirmen und damit den Kontrast im Bild zu erhöhen. Obwohl LEDs an sich kein weißes Licht erzeugen können, sind Weißlicht-LEDs heutzutage sehr verbreitet. Um weißes Licht mit LEDs zu erzeugen gibt es zwei Wege. Die eine Variante sind blaue LEDs mit einer Fluoreszenz-Schicht, die einen Teil des blauen Lichts in eine breites Fluoreszenz-Spektrum umwandelt. Das umgewandelte Licht ist gelb, das in der Mischung mit dem restlichen blauen Licht ein weißes Licht ergibt. Diese Art von LEDs haben oft ein etwas blästichiges, kaltes Weiß. Die andere Variante ist das Zusammenschalten von je einer roten, grünen und blauen LED. Mittels additiver Farbmischung ergibt das Licht der drei LEDs zusammen weißes Licht.

Ein großer Vorteil von LEDs ist ihre sehr hohe Lebensdauer, allerdings verlieren LEDs einen Teil ihrer Leuchtkraft bevor sie ausfallen. LEDs sind sparsam im Energieverbrauch, da sie im Vergleich zu anderen Lichtquellen wenig Wärme produzieren. Dennoch stellt die Wärme das größte Risiko für die Laufzeit von LEDs dar, weshalb auf eine gute Ableitung der Wärme geachtet werden sollte.

Schließlich sind LEDs sehr robust gegen Erschütterungen und Vibrationen. Der vielleicht größte Vorteil von LEDs ist aber, dass man LEDs nahezu verschleißfrei blitzen kann. Da LEDs sehr schnell auf das Anlegen einer Spannung reagieren, sind sie wie keine andere Lichtquelle für Blitzbeleuchtungen geeignet.

Laser: Laserlicht eignet sich aufgrund seiner besonders guten Fokussierung vor allem für strukturierte Beleuchtungen. Eine typische Anwendung sind Laserlinien, die eine Überprüfung von Teilegeometrien auch dann zulassen, wenn kein Grauwertkontrast vorhanden sind. Ein Beispiel für eine derartige Anwendung ist in Abschnitt 9.4 gezeigt. Dort wird das Vorhandensein schwarzer Haltestreben auf schwarzem Kunststoff anhand der charakteristischen Reflexionen einer Laserlinie überprüft.

Leuchtstoffröhren: Leuchtstoffröhren werden heute vorwiegend für sehr großflächige Beleuchtungen eingesetzt, um eine homogene Beleuchtung über größere Entfernung zu erzielen. Bei der Verwendung in der Bildverarbeitung muss man beachten, dass handelsübliche Leuchtstoffröhren infolge der Netzfrequenz periodische Helligkeitsschwankungen aufweisen, die sich im Kamerabild sehr unangenehm bemerkbar machen können. Man sollte daher grundsätzlich Hochfrequenz-Vorschaltgeräte verwenden, die außerdem noch den Effekt haben, dass die Röhren flackerfrei starten. Das ist sehr wichtig, wenn man Leuchtstoffröhren in geschalteten Beleuchtungssystemen verwenden will.

Leuchtstoffröhren können in verschiedenen Beleuchtungsanordnungen eingesetzt werden. Es gibt beispielsweise Leuchtplatten, bei denen mehrere Röhren unter halbtransparentem Kunststoff montiert sind. Von außen betrachtet erscheint der Kunststoff als homogen leuchtende Fläche. Solche Leuchtplatten eignen sich für diffuse Auflichtbeleuchtungen und als Durchlicht für einfache Vermessungsaufgaben. Eine andere Bauform sind Ringleuchten, meist mit Durchmessern von einigen Zentimetern. Ein Nachteil dieser Ringleuchten ist, dass sie nicht rundum homogen sind, da an einer Stelle der Starter eingebaut sein muss.

Halogenlampen: Wegen ihres sehr „ungemütlichen“ weißen Lichts werden diese Lampen auch als *Kaltlichtquellen* bezeichnet, obwohl sie alles andere als kalt sind. Sie werden meistens in Verbindung mit Glasfaserlichtleitern benutzt. Mit einem Glasfaserlichtleiter mit Spotvorsatz lassen sich bestimmte Punkte eines Objekts sehr gezielt beleuchten. Es gibt außerdem ringförmige Vorsätze, in die häufig ein Polarisationsfilter eingebaut ist, wie er beispielsweise in Abschnitt 5.2.1 verwendet wurde. Mit Vorsätzen, die ähnlich den oben erwähnten Leuchtplatten das Licht des Faserbündels auf eine größere Fläche verteilen, lassen sich sehr homogene weiße Durchlichtbeleuchtungen aufbauen.

Da Halogenlampen mit Gleichspannung betrieben werden können, entfällt die für Leuchtstoffröhren typische Problematik der Wechselspannungsfrequenz. Allerdings muss man wegen der Alterung der Lampe von Zeit zu Zeit rekalibrieren bzw. Lampen mit Helligkeitsregelung verwenden.

Den Halogenlampen im Aufbau sehr ähnlich sind Xenon-Lampen, die als Blitzbeleuchtungen verwendet werden, da sich Helligkeit und Frequenz dieser

Lampen sehr gut steuern lässt. Die Helligkeit von Xenon-Blitzbeleuchtungen kann dabei die Helligkeit von LEDs übertreffen.

8.6.2 Auflicht

Man kann zwei grundsätzliche Arten von Beleuchtungsanordnungen unterscheiden, Auflicht und Durchlicht. Beim Auflicht befindet sich die Beleuchtung auf der gleichen Seite des Objekts wie die Kamera. Je nach Form und Position der Lichtquelle und Winkel des Lichteinfalls lassen sich damit sehr unterschiedliche Effekte erzielen, wie wir gleich sehen werden.

Die Informationen in diesem und dem folgenden Abschnitt sind zum Teil (Illuminating Solutions 1996) entnommen (mit freundlicher Genehmigung der Dolan-Jenner Europe B.V.).

Diffuses Auflicht: Unter diffuser Beleuchtung versteht man eine Beleuchtung ohne klar feststellbare Richtung. Dadurch werden Schatten und starke Reflexionen vermieden. Diffuse Beleuchtung entsteht, wenn das Licht von einer großen Fläche abgestrahlt wird. Dies lässt sich beispielsweise durch Einbau der Lichtquelle hinter milchigem Glas oder Kunststoff oder durch Reflexion an weißen Schirmen erreichen. Das Prinzip zeigt Abbildung 8.25.

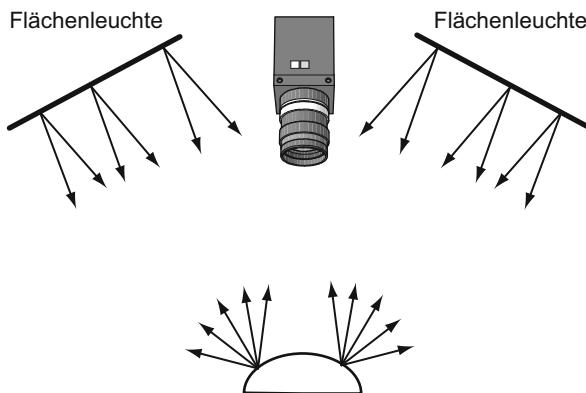


Abbildung 8.25: Prinzip der diffusen Auflichtbeleuchtung

Gerichtetes Auflicht: Bei dieser Art von Auflichtbeleuchtung wird die Lichtquelle so platziert, dass das Licht die Oberfläche nicht allseitig, sondern unter einem bestimmten Winkel trifft. Für die Platzierung der Kamera gibt es dann zwei Möglichkeiten:

Hellfeldbeleuchtung: Von *Hellfeldbeleuchtung* spricht man, wenn die Kamera im (theoretischen) Ausfallswinkel des einfallenden Lichts positioniert wird. Wenn die beleuchtete Oberfläche die Reflexionseigenschaften eines Spiegels aufweist, wird die Kamera dann voll vom reflektierten Licht getroffen, so dass die Oberfläche

maximal hell erscheint. Daraus ergibt sich auch die Bezeichnung *Helffeld*. Helffeldbeleuchtung wurde in Abschnitt 9.5 für die Kontrolle von Bohrungen benutzt. Die Bohrungen reflektieren natürlich kein Licht, so dass sie gegen die polierte Metalloberfläche sehr gut kontrastieren.

Dunkelfeldbeleuchtung: Von *Dunkelfeldbeleuchtung* spricht man, wenn die Kamera außerhalb der direkten Reflexionsrichtung des einfallenden Lichts platziert wird. Der Name ergibt sich daraus, dass eine völlig glatte Oberfläche dunkel erscheint, da das Licht vollständig an der Kamera vorbeireflektiert wird. Unregelmäßigkeiten der Oberfläche erscheinen dagegen hell, da sich der Reflexionswinkel ändert.

Dunkelfeldbeleuchtung wurde in Abschnitt 5.3 benutzt, um die Vertiefungen sichtbar zu machen, die beim Prägen von Ziffern in Metall auftreten. Durch die diffuse Reflexion an der Oberflächenstruktur des Metalls erscheint die Oberfläche annähernd gleichmäßig hell. Deutliche Vertiefungen und Erhöhungen führen dagegen zu anderen Reflexionswinkeln und zu Abschattungen, so dass ein deutlicher Kontrast zur Oberfläche entsteht, wie Abbildung 8.26 zeigt.

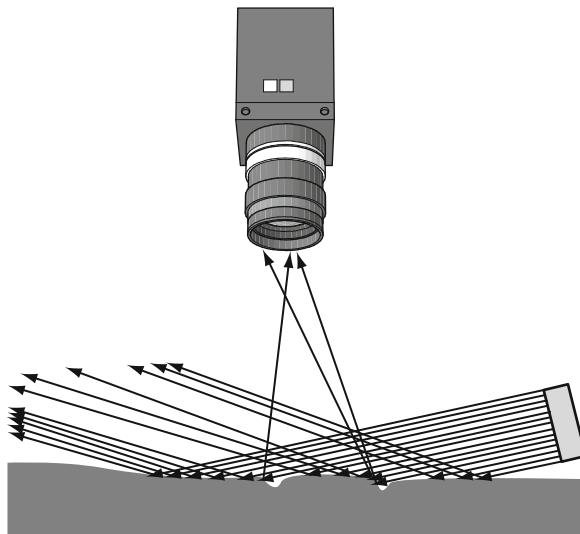


Abbildung 8.26: Wirkung einer Dunkelfeldbeleuchtung auf Unregelmäßigkeiten einer Oberfläche

Im Extremfall kann man nahezu parallel zur Oberfläche beleuchten und dazu beispielsweise einen Linienprojektor verwenden, um einen geradlinigen Lichteinfall zu garantieren. Man erhält auf diese Weise maximalen Kontrast auf texturierten Oberflächen und sehr scharfe Schattenbildung. Diese Methode wurde in Abschnitt 7.4 zur Vermessung der Zähne eines Zahnradiers benutzt. Man beobachtet oft, dass Menschen ein Objekt aus verschiedenen Winkeln betrachten, es drehen und kippen, um Oberflächenstrukturen zu erkennen. Denselben Effekt kann man mit

geringerem mechanischen Aufwand dadurch erzielen, dass man mehrere Lichtquellen in verschiedenen Höhen anbringt und nacheinander schaltet. Solche dynamischen Beleuchtungen lassen sich gut mit LEDs aufbauen, beispielsweise in Form einer Beleuchtungskuppel, die sowohl allseitig diffuse Beleuchtung ermöglicht als auch radiale und konzentrische Lichtstrukturen.

Polarisiertes Licht: Polarisationsfilter können zur Vermeidung von Reflexionen eingesetzt werden. Polarisiertes Licht schwingt nur in einer einzigen Richtung quer zu seiner Ausbreitungsrichtung. Wenn eine Oberfläche mit polarisiertem Licht beleuchtet wird, sind alle durch spiegelnde Reflexion zurückgeworfenen Strahlen vollständig polarisiert. Durch einen im rechten Winkel zu dieser Polarisationsrichtung angeordneten Polarisationsfilter vor der Kamera lassen sich diese Lichtstrahlen zurückhalten. Die Reflexionen sind dann im Kamerabild nicht sichtbar.

Ringbeleuchtung: Ringförmige Beleuchtungen liefern intensives, schattenfreies Licht, das entlang der optischen Achse der Kamera einfällt. Sie werden oft in Verbindung mit Polarisationsfiltern eingesetzt, durch die störende Reflexionen beseitigt werden. Diese Beleuchtungsmethode wurde bei der Erkennung lasergravierter Zeichen auf einem IC in Abschnitt 5.2.1 benutzt. Abbildung 8.27 zeigt das Prinzip dieses Beleuchtungsaufbaus.

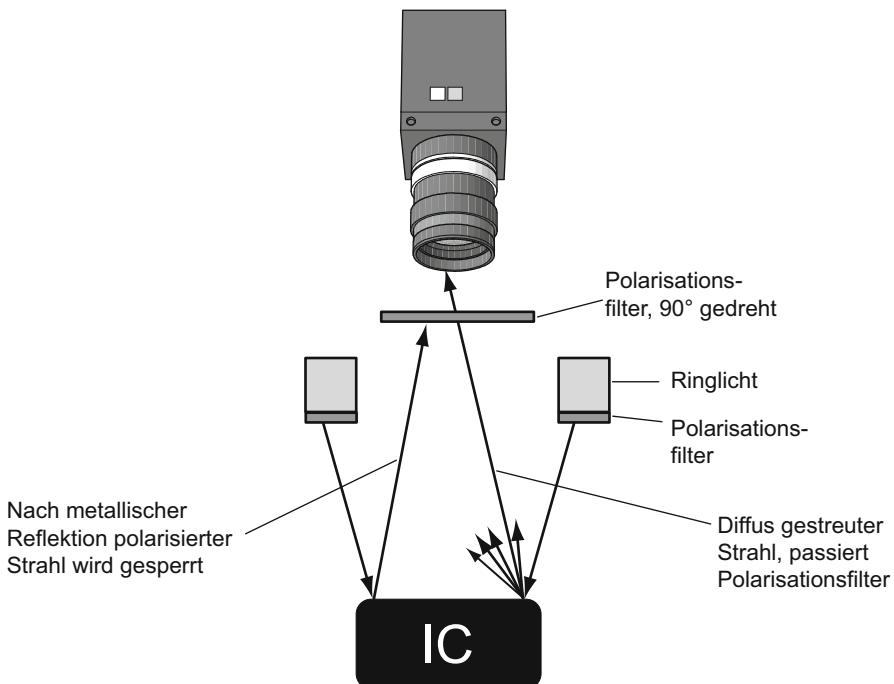


Abbildung 8.27: Ringlicht mit Kreuzpolarisation

Ringbeleuchtungen kommen in verschiedenen Bauformen vor. Es gibt ringförmige Leuchtstoffröhren, Ringlichtaufsätze für Faseroptiken zum Anschluss an Kaltlichtquellen und Ringlichter, die aus LEDs aufgebaut sind. Diese letzte Möglichkeit wurde z. B. für den Aufbau eines sehr großen Ringlichts für die Prüfung eines Befestigungsflansches in Abschnitt 9.3 benutzt.

Beleuchtung im Strahlengang: Mithilfe spezieller Aufsätze oder Adapter ist es möglich, Licht in den Strahlengang eines Objektivs einzuspiegeln. Damit lassen sich beispielsweise Innenbohrungen beleuchten. Auch bei faseroptischen Endoskopen bedient man sich dieser Methode, um Licht an das Objekt zu bringen. Das Licht wird in das Faserbündel eingespiegelt und beleuchtet den unzugänglichen Gegenstand durch die Optik hindurch. Das reflektierte Licht wird dann durch das Faserbündel zum Sensor zurückgeführt.

Strukturierte Beleuchtung: Unter strukturierter Beleuchtung versteht man die Projektion von Linien, Gittern oder Kreisen auf ein Objekt. Dadurch lassen sich dreidimensionale Charakteristika des Objekts feststellen sowie Vermessungen und Anwesenheitskontrollen auch dann vornehmen, wenn kein Helligkeitskontrast vorhanden ist. Neben Linienprojektoren werden hierfür häufig Laser verwendet, deren gute Fokussierungseigenschaften sehr genaue Projektionen erlauben. Diese Beleuchtungsmethode wird in Abschnitt 9.4 für eine kontrastlose Anwesenheitskontrolle verwendet. Abbildung 8.28 zeigt eine kontrastlose Höhenmessung an einem schwarzen Kunststoffteil. Der Abstand zwischen den oberen und unteren reflektierten Linien liefert ein Maß für die Höhe der Stufe zwischen den beiden Abschnitten des Prüfteils. Die Maßlinien wurden schraffiert dargestellt, damit sie im Graubilddruck noch zu erkennen sind.

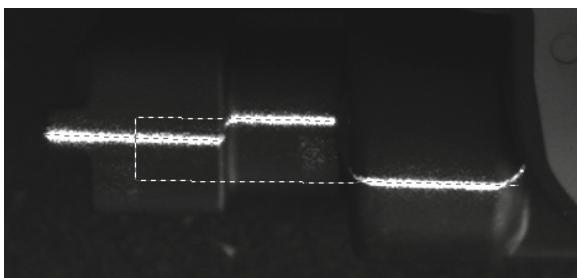


Abbildung 8.28: Höhenmessung mit strukturierter Beleuchtung

8.6.3 Durchlicht

Bei Durchlichtbeleuchtung befindet sich das Objekt zwischen Lichtquelle und Kamera. Die Kamera sieht das Objekt als Schatten. Diese Art der Beleuchtung erzeugt daher ausgezeichnete Kontraste und ist gut für Vermessungsaufgaben geeignet. Wie im vorigen Abschnitt entstammen die folgenden Informationen zum

Teil (Illuminating Solutions 1996) (mit freundlicher Genehmigung der Dolan-Jenner Europe B.V.).

Diffuse Hintergrundbeleuchtung: Ähnlich wie diffuse Auflichtbeleuchtung durch Abstrahlung von großen Flächen erzielt wird, erreicht man eine diffuse Hintergrundbeleuchtung durch die Verwendung von Leuchtplatten. Die Lichtquelle wird dabei mit halbtransparenten Kunststoff- oder Glasplatten abgedeckt. Als Leuchtmittel werden meistens LED-Platten verwendet. Diffuse Hintergrundbeleuchtung wurde im Abschnitt 7.3 für die Beleuchtung von Stanzteilen für die Elektronik verwendet. Abbildung 8.29 zeigt das Prinzip. Das Objekt hält das Licht von der Leuchtplatte zurück und erscheint im Kamerabild als Schatten.

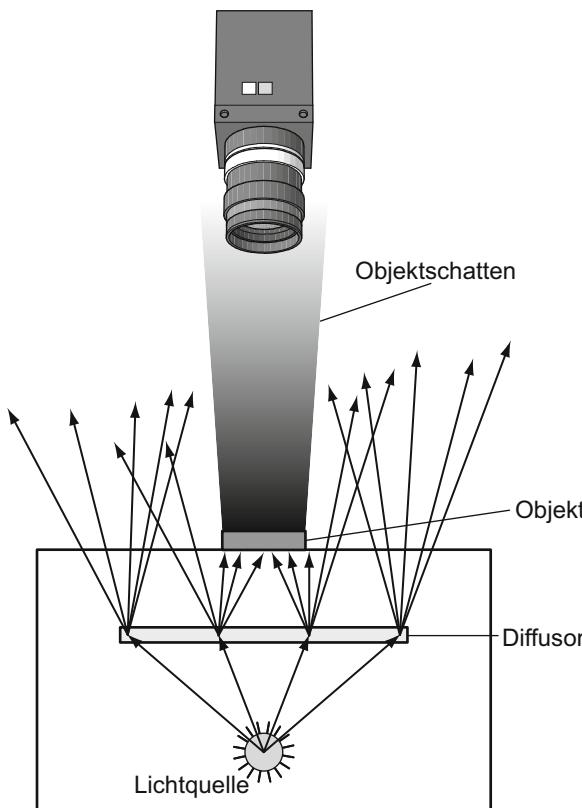


Abbildung 8.29: Prinzip der diffusen Hintergrundbeleuchtung

Gerichtete Hintergrundbeleuchtung: Ein Problem bei diffusen Hintergrundbeleuchtungen stellt das Auftreten von Seitenlicht dar. Abbildung 8.30 zeigt das am Beispiel eines Gewindes. Im rechten Bild erhält das Gewinde aus anderen Bereichen der Leuchtplatte seitliches Licht, was die stabile Lokalisierung der Gewinde-

kanten erheblich erschwert. Im linken Bild wurde dieses Seitenlicht abgeschirmt, wodurch die Kontur deutlich präziser wird.

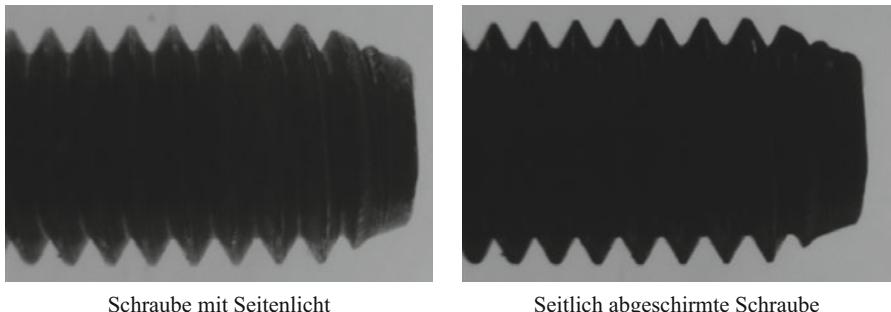


Abbildung 8.30: Wirkung von Seitenlicht auf Objektkonturen

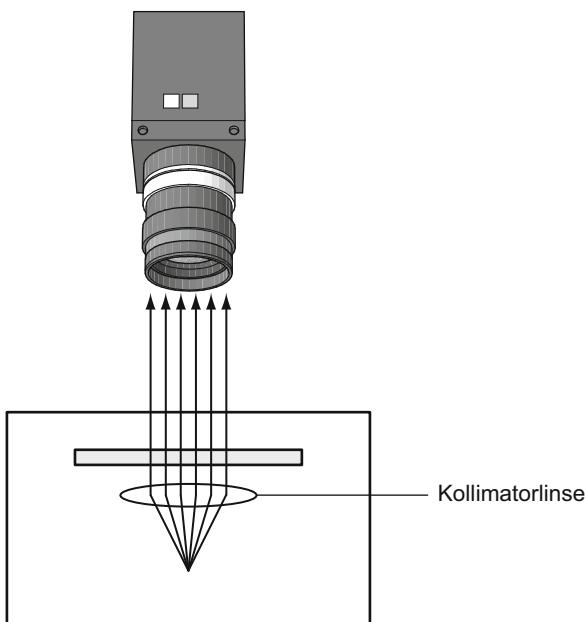


Abbildung 8.31: Telezentrische Hintergrundbeleuchtung

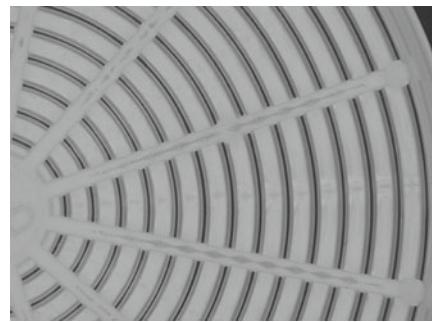
Für hochgenaue Vermessungen erzielt man noch weitaus bessere Ergebnisse bei Verwendung einer telezentrischen Hintergrundbeleuchtung. Analog zum telezentrischen Objektiv weist diese Beleuchtungsform einen parallelen Strahlengang auf, der durch eine Kollimatorlinse zwischen Lichtquelle und Leuchtplatte erreicht wird, wie Abbildung 8.31 zeigt. Diese Beleuchtung wurde bei der Gewindeflanschvermessung in Abschnitt 7.6 benutzt, da nur so das Vermessungsergebnis vom exakten Abstand zwischen Beleuchtung und Messebene unabhängig gemacht

werden kann. Dies ist besonders wichtig bei Objekten, die wegen ihrer Form und Ausdehnung nicht unmittelbar auf einer Leuchtplatte platziert werden können. Das ist beispielsweise bei zylindrischen Objekten wie Schrauben der Fall, bei denen der Abstand zwischen der sichtbaren Kontur und der Leuchtplatte mindestens dem Radius des Objekts entspricht.

Polarisierte Hintergrundbeleuchtung: Wie bei Auflicht, so können auch bei Hintergrundbeleuchtung Polarisationsfilter benutzt werden. Man kann dadurch den Kontrast bei transparenten Teilen verbessern und sogar Oberflächenspannungen sichtbar machen. Man belegt hierzu die Leuchtplatte mit einem Polarisationsfilter und ordnet vor der Kamera einen zweiten Polarisationsfilter an. Je nach dem Winkel der beiden Polarisationsfilter entstehen unterschiedlich starke Kontraste. Abbildung 8.32 zeigt dies am Beispiel eines transparenten Kunststoffsiebs.



Sieb im Durchlicht



Sieb in kreuzpolarisiertem Durchlicht

Abbildung 8.32: Verwendung von polarisiertem Durchlicht zur Untersuchung transparenter Objekte

8.7 Zusammenfassung

Halbleitersensor: In der industriellen Bildverarbeitung werden praktisch ausschließlich Kameras mit Halbleitersensoren verwendet. Das einfallende Licht setzt im lichtempfindlichen Teil des Sensors Ladungen frei, die über die Belichtungszeit gesammelt werden und schließlich als proportionale Spannung ausgelesen werden.

Pixel: Bezeichnung für ein einzelnes Sensorelement eines CCD-Sensors. Pixel typischer Halbleitersensoren haben eine Größe von weniger als zehn Mikrometer.

CCD-Sensor: Der heute noch in der industriellen Bildverarbeitung dominierende Halbleitersensortyp ist der CCD-Sensor. CCD-Sensoren haben meistens nur einen Ladungsvstärker durch den die gesammelten Ladungen aller Pixel in Spannun-

gen umgewandelt werden. Im Vergleich zu CMOS-Sensoren ist das Rauschen bei ihnen im Allgemeinen geringer.

CMOS-Sensor: Bei CMOS-Sensoren verfügt jeder Pixel über einen eigenen Ladungsverstärker der die während der Belichtungszeit integrierten Ladungen in eine Spannung umwandelt. CMOS-Sensoren sind im Consumer-Markt sehr verbreitet, da sie sehr kostengünstig und hoch integriert produziert werden können. Diese weitverbreitete Art von CMOS-Sensoren ist aufgrund des Rolling Shutters für die industrielle Bildverarbeitung nur eingeschränkt verwendbar.

Farbsensoren: Für die Aufnahme von Farbbildern sind zwei Arten von Farbsensoren verbreitet:

- Drei-Chip-Farbкамeras mit einem Strahlteiler, durch den das Licht je nach Farbe auf einen von drei Halbleitersensor gelenkt wird.
- Ein-Chip-Farbкамeras mit einer Farbfiltermatrix, durch die auf die Pixel des einzigen Halbleitersensors nur das Licht einer Farbe fällt.

Quanteneffizienz: Die Quanteneffizienz beschreibt wie gut die eingestrahlten Photonen Elektronen freisetzen. Sie gibt dazu das Verhältnis zwischen den freigesetzten Elektronen und eingestrahlten Photonen an.

Spektrale Empfindlichkeit: Die Quanteneffizienz von Halbleitersensoren ist von der Wellenlänge abhängig. Im langwelligen Bereich reicht ihre Empfindlichkeit bis in den Infrarottbereich.

Digitalkameras: Kameras diesen Typs wandeln die Sensorsignale in digitale Werte um und übertragen diese über standardisierte, digitale Schnittstellen. Damit ist eine verlustfreie Übertragung der Bilddaten vom Sensor bis zum Bildverarbeitungsrechner sichergestellt.

Demosaicing: Bei Ein-Chip-Farbкамeras müssen die Farbwerte für einen Pixel aus den Helligkeitswerten der benachbarten Pixel interpoliert werden. Dies wird als Demosaicing bezeichnet, für das verschiedene Verfahren existieren.

CameraLink: Dies ist die erste speziell für die Bildverarbeitung entwickelte digitale Datenschnittstelle. Aufgrund ihrer hohen Bandbreite ist sie für High-Speed-Anwendungen nach wie vor die Schnittstelle der Wahl. Für die Übertragung der Bilddaten zum Bildverarbeitungsrechner muss dieser über einen Frame Grabber verfügen.

FireWire: Der als IEEE1394 genormte serielle Datenbus ist eine leistungsfähige und äußerst zuverlässige Datenschnittstelle. Sie erlaubt Datenraten von 400 Mbit/s (IEEE1394a) bzw. 800 Mbit/s (IEEE1394b). Über den Bus ist es möglich angegeschlossene Kameras mit Spannung zu versorgen.

GigE Vision: Dieser Standard für die Übertragung von Bilddaten nutzt Gigabit-Ethernet als Übertragungsmedium. Als einzige digitale Schnittstelle erlaubt Gigabit-Ethernet Kabellängen bis zu 100 Metern.

Zeilenkameras: Dieser Kameratyp hat keinen matrixförmigen Sensor, sondern eine einzelne Zeile von Sensorelementen. Diese Zeile kann jedoch sehr lang sein, so dass sich entsprechend hohe Bildauflösungen erreichen lassen. Zeilenkameras werden häufig für Oberflächenprüfungen an kontinuierlichen Materialien oder zum Abscannen ungünstiger Werkstückgeometrien eingesetzt. Für die Bilddatenübertragung wird häufig CameraLink verwendet.

Blendenzahl: Relative Öffnung eines Objektivs, bezogen auf seine Brennweite. Die Beleuchtungsstärke auf dem Sensor ist umgekehrt proportional zum Quadrat der Blendenzahl.

Lichtstärke: Größtmögliche relative Öffnung eines Objektivs.

Abbildungsmaßstab: Bildgröße B und Gegenstandsgröße G stehen im gleichen Verhältnis wie Gegenstandsweite g und Bildweite b , d. h. die Entfernung zwischen Objektiv und Gegenstand einerseits und Objektiv und Sensor andererseits:

$$\frac{B}{G} = \frac{b}{g}.$$

Abbildungsgleichung: Die Abbildungsgleichung für dünne Linsen setzt Brennweite f , Gegenstandsweite g und Bildweite b zueinander in Beziehung:

$$\frac{1}{f} = \frac{1}{g} + \frac{1}{b}.$$

Bildfeld: Das Bildfeld einer Kamera ergibt sich aus Gegenstandsweite g , Brennweite f und Sensorgröße B näherungsweise zu:

$$G \approx \frac{g}{f} B.$$

Schärfentiefe: Bereich, innerhalb dessen die Gegenstandsweite variieren kann, ohne dass die zulässige Unschärfe überschritten wird. Meist wird die halbe Größe eines Sensorelementes als zulässige Unschärfe verwendet. Die Schärfentiefe ist abhängig von:

- Blendenzahl: je größer die Blendenzahl, desto größer die Schärfentiefe
- Brennweite: je größer die Brennweite, desto kleiner die Schärfentiefe
- Abstand: je größer der Abstand, desto größer die Schärfentiefe; ab einem gewissen Grenzabstand werden alle Objekte scharf abgebildet

Auflagemafß: Abstand zwischen Objektivaufgabe und Kamerasensor, damit gleichzeitig der minimale Abstand zwischen Objektiv und Sensor. Man unter-

scheidet C-Mount (17,5 mm) und CS-Mount (12,5 mm). C-Mount-Objektive können durch Einsetzen eines Zwischenrings mit CS-Mount-Kameras verwendet werden. Umgekehrt geht das nicht, weil sich die Linse nicht dicht genug an den Sensor heranbringen lässt.

Zwischenring: Durch Einsetzen von Zwischenringen kann man das Objektiv näher an den Gegenstand heranbringen.

Objektivgröße und optisches Format: Halbleitersensoren werden in verschiedenen Größen gefertigt, die man je nach Größe des erforderlichen Lichtkreises in Gruppen einteilt. Dementsprechend gibt es 1"-, 2/3"-, 1/2"-, 1/3"-, 1/4"-Kameras. Objektive größerer Formats können auf Kameras kleineren Formats verwendet werden, da sie den Sensor vollständig ausleuchten. Die umgekehrte Kombination ist nicht möglich.

Telezentrische Objektive: Diese Objektive haben einen objektseitig parallelen Strahlengang. Innerhalb des Telezentriebereiches ist daher die Bildgröße entfernungsunabhängig. Sie werden für Vermessungsaufgaben verwendet, wobei das Objektiv mindestens so groß sein muss wie der aufzunehmende Gegenstand.

Endoskope: Endoskope werden zur Untersuchung unzugänglicher Objektbereiche, wie z. B. Hohlräumen, eingesetzt. Sie bestehen meist aus einem Glasfaserlichtleiter mit einer Optik und einer Vorrichtung zur Einspiegelung von Licht aus einer Kaltlichtquelle.

LEDs: Leuchtdioden ermöglichen einen geometrisch sehr flexiblen Beleuchtungsaufbau. Sie sind nahezu verschleißfrei und lassen Blitzbeleuchtungen mit hoher Intensität zu.

Laser: Wegen der guten Fokussierbarkeit lassen sich Laser gut für strukturierte Beleuchtungen einsetzen.

Leuchtstoffröhren: Diese Lichtquelle ist besonders für diffuse Auflicht- aber auch Hintergrundbeleuchtungen geeignet. Sie sollte immer mit Hochfrequenzorschaltgeräten verwendet werden, um Schwebungen in der Beleuchtungsstärke aufgrund der Netzfrequenz auszuschließen.

Kaltlichtquellen: Halogenlampen liefern in Verbindung mit stabilisierten Gleichspannungsquellen sehr hochwertiges Licht, das gut für Vermessungsaufgaben geeignet ist.

Auflichtbeleuchtung: Bei Auflichtbeleuchtung befindet sich die Lichtquelle auf der gleichen Seite des Gegenstandes wie die Kamera. Man unterscheidet unter anderem:

- Diffuses Auflicht
- Gerichtetes Auflicht (Dunkelfeldbeleuchtung) zur Sichtbarmachung von Unregelmäßigkeiten auf Oberflächen
- Ringbeleuchtung, für intensives, schattenfreies Licht

- Beleuchtung im Strahlengang, vor allem für Hohlräume, Bohrungen und dgl.
- Strukturierte Beleuchtung, zur Sichtbarmachung geometrischer Eigenschaften ohne Helligkeitskontrast

Hintergrundbeleuchtung/Durchlicht: Bei Hintergrundbeleuchtung befindet sich das Objekt zwischen Lichtquelle und Kamera. Die Kamera sieht also einen Schatten. Man unterscheidet vor allem:

- Diffuse Hintergrundbeleuchtung
- Gerichtete Hintergrundbeleuchtung
- Telezentrische Hintergrundbeleuchtung für hochgenaue Vermessungsaufgaben

Polarisiertes Licht: Polarisationsfilter können zur Unterdrückung von Reflexen verwendet werden (vor allem bei Ringbeleuchtung) und zur sichtbar Machung von Strukturen in transparenten Objekten.

9. Anwesenheitskontrolle

Von der Aufgabenstellung her betrachtet, erscheint Anwesenheitskontrolle als ein klar fassbares und strukturell einfaches Gebiet, so dass es oft am Anfang einer Einführung in die Bildverarbeitung steht. Tatsächlich ist der Begriff der Anwesenheitskontrolle aber recht schwierig einzugrenzen: Geht es hier nur um einfaches Zählen von Objekten oder spielen Eigenschaften dieser Objekte eine Rolle? Sind diese Eigenschaften einfache Merkmale, oder ist das Gesamtaussehen des Objekts von Bedeutung? Beziehen sich Eigenschaften jeweils nur auf Einzelobjekte, oder müssen, wie bei der Montagekontrolle, auch Bezüge zwischen Objekten berücksichtigt werden.

So weit gefächert wie diese Fragestellungen sind auch die Methoden, die in diesem Bereich zum Einsatz kommen. Oft dienen Verfahren aus den bisher besprochenen Anwendungsbereichen, wie Klassifikationen und Vermessungen, als Hilfsmittel der Anwesenheitskontrolle. Ein weiterer Grund, die Anwesenheitskontrolle nicht an den Anfang unserer Einführung in Bildverarbeitung zu stellen, lag darin, dass dieser Anwendungsbereich weniger durch bestimmte Methoden und Algorithmen gekennzeichnet ist als durch die Art der Fragestellung. Wenn Vermessung als Hilfsmittel der Montagekontrolle benutzt wird, interessiert man sich weniger dafür, dass ganz bestimmte Maße mit sehr geringer Toleranz eingehalten werden, als dafür, dass Objekte in bestimmten Relationen zueinander stehen. Bei Klassifikationen im Rahmen der Anwesenheitsprüfung geht es weniger darum, in der Art einer Schrifterkennung zu entscheiden *welche* Objekte vorhanden sind, sondern ob das *richtige* Objekt vorliegt.

Aufgrund dieser Vielfalt der Aufgabenstellungen und Methoden ist die Anwesenheitskontrolle vom Herangehen an die Prüfaufgabe her gesehen, ein verhältnismäßig schwieriges Gebiet. Bei Vermessungs- und Klassifikationsaufgaben ist die Vorgehensweise meist jeweils sehr ähnlich und entspricht der Art und Weise, in der ein Mensch die Aufgabe intuitiv beschreiben würde. Bei Aufgaben aus der Anwesenheitskontrolle ist der prinzipielle Ansatz oft schwerer zu finden und weniger intuitiv einsichtig.

Mit den Beispielen dieses Kapitels wollen wir einen Eindruck von der Vielfalt des Bereiches der Anwesenheitskontrolle geben und von der grundsätzlichen Denkweise, mit der man an diese Art von Prüfaufgabe herangeht.

9.1 Anwesenheitskontrolle mit Schwenk-Neigekopf-Zoomkameras

Unser erstes Beispiel für den Bereich der Anwesenheitskontrolle ist in sofern ein einfaches Beispiel, als es lediglich die Anwesenheit von Objekten und deren korrekte Verbauung prüft. Das Prüfobjekt ist ein Kfz-Motor, an dem bei einer Reihe von Sensoren, Schellen und Klammern die korrekte Montage geprüft werden muss. Insgesamt müssen rund 60 derartige sehr verschiedene Prüfmerkmale geprüft werden. Die Prüfmerkmale sind oft durch Binarisierung nicht vom Hintergrund zu trennen oder lassen sich aufgrund ihrer binären Kontur kaum von Reflexen unterscheiden. Deshalb werden alle Prüfmerkmale durch Template Matching gesucht und anhand des Matching-Ergebnisses ihre einwandfreie Montage überprüft wird. Wir haben dieses Beispiel weniger wegen der relativ einfachen Prüfstrategie ausgewählt, sondern mehr um zu zeigen wie man die Bilder von so vielen unabhängigen Prüfmerkmale effizient aufnehmen kann.

9.1.1 Prüfteilgeometrie

Die offensichtlichste Schwierigkeit bei dieser Prüfaufgabe ist die Größe des Motors über den die verhältnismäßig kleinen Sensoren, Schellen und Klammern verteilt sind. Bei einer Aufnahme des gesamten Motors sind beispielsweise die Sensoren nur wenige Bildpunkte groß. Bei einer so geringen Objektgröße kann man nur prüfen ob ein heller Fleck vorhanden ist oder nicht. Die Unterschiede zwischen korrektem Einbau und fehlerhaftem sind kleiner als ein Pixel und damit nicht mehr zu prüfen. Und ob es sich bei dem hellen Punkt tatsächlich um den gesuchten Sensor oder einen Lichtreflex handelt kann man ebenfalls unmöglich entscheiden. Man könnte damit zu dem Schluss kommen, dass für die Prüfungen der einzelnen Sensoren und Bauteile jeweils einzelne Kameras erforderlich sind, was einen erheblichen Hardware-Einsatz zur Folge hätte.

Tatsächlich kann man mit deutlich weniger Kameras auskommen, wenn man dafür spezielle verwendet. Für solche Anwendungen bieten sich sogenannte Schwenk-Neigekopf-Zoomkameras an. Diese bestehen aus einem Kamerakopf, der so auf seinem Fuß montiert ist, dass er mit den im Fuß integrierten Motoren sowohl horizontal als auch vertikal gedreht werden kann. Damit kann die Blickrichtung der in den Kamerakopf integrierten CCD-Kamera verändert werden, während mit dem ebenfalls integrierten Zoom-Objektiv die Größe des aufgenommenen Bildfeldes verändert werden kann. Die Kameras werden vor allem für Videokonferenzen und Überwachungsaufgaben hergestellt, können aber auch für die industrielle Bildverarbeitung eingesetzt werden. Die vorliegende Motorenprüfung ist eine typische Anwendung für den Einsatz dieses Kamera-Typs, bei der nacheinander eine Reihe von Detailansichten des Motors aufgenommen wird. Die komplexe Form des Motors und die Vielzahl der Prüfmerkmale erforderte insge-

samt fünf Schwenk-Neigekopf-Zoomkameras, die den Motor von verschiedenen Seiten und aus unterschiedlichen Höhen aufnehmen.

So praktisch Schwenk-Neigekopf-Zoomkameras sind, bringt ihre Verwendung doch einige Probleme mit sich. Zunächst erfordert die Auswahl der Kameraposition sehr viel Sorgfalt. So muss die Kamera von ihrem Standort aus ein freies Blickfeld auf alle Prüfmerkmale haben, die sie aufnehmen soll.

Aber auch die Blickrichtung der Kamera ist nicht ganz unproblematisch. In den seltensten Fällen wird man alle Prüfmerkmale unter dem optimalen Winkel aufnehmen können. Meistens wird man schräg darauf schauen. Deshalb muss man frühzeitig ausprobieren, ob alle Prüfmerkmale von der geplanten Position aus geprüft werden können. In unserem Beispiel ist das bei einigen Prüfmerkmalen tatsächlich ein Problem. So sind manche durch andere Bauteile teilweise verdeckt, während bei anderen der Unterschied zwischen korrekter und falscher Montage fast parallel zur Blickrichtung liegt und damit kaum zu erkennen ist. In Abbildung 9.1 ist auf der linken Seite ein aufgrund des Blickwinkels nur schwer zu prüfender Sensor zu sehen und auf der rechten Seite ein durch ein Rohr fast verdecktes Prüfmerkmal.

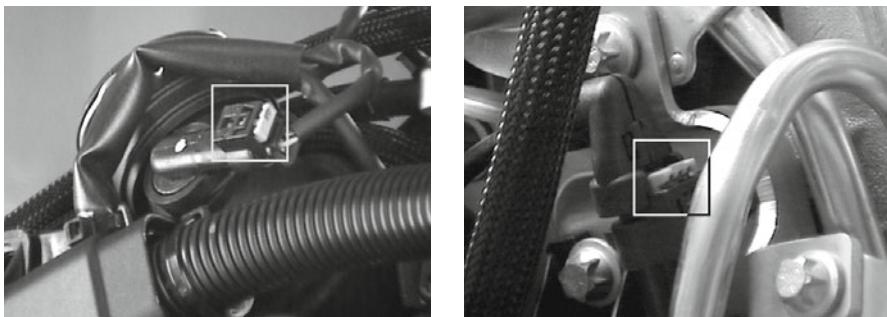


Abbildung 9.1: Ungünstiger Blickwinkel und nahezu verdecktes Prüfmerkmal

Bei der Auswahl der Kameraposition ist noch ein weiterer Gesichtspunkt zu bedenken: die Möglichkeiten des Zoom-Objektivs. Da das Objektiv fest in die Kamera integriert ist, kann man es nicht durch eines mit einer anderen Brennweite ersetzen. Damit ist der weiteste Sichtwinkel ebenso festgelegt wie die maximale Vergrößerung. Kleine Prüfmerkmale müssen somit nah genug bei der Kamera sein, damit die Auflösung der Kamera ausreicht um sie zu prüfen. Selbstverständlich kann man keinen Zwischenring hinter dem Objektiv einbauen um den minimalen Arbeitsabstand zu verkleinern. Deshalb darf der Abstand zwischen den Prüfmerkmalen und der Kamera nicht zu klein sein.

9.1.2 Beleuchtung

Die in alle Richtungen gekrümmten Oberflächen des Motors und seiner Komponenten, so wie die vielen leicht versteckten Prüfmerkmale, erfordern eine Beleuchtung die sehr diffus aus möglichst vielen Raumrichtungen kommt. Eine individuelle Beleuchtung jedes einzelnen Prüfmerkmals wäre aufgrund ihrer Anzahl ein erheblicher Aufwand. Mit einer Reihe großflächiger Leuchtelemente mit Leuchtstoffröhren kann man den Motor aus allen Richtungen so diffus beleuchten, dass fast alle Prüfmerkmale geprüft werden können. Für einige wenige Prüfungen ist es besser einzelne Elemente der Beleuchtung auszuschalten, um so den Kontrast zu verbessern oder störende Reflexe zu vermeiden.

9.1.3 Positionierung

Der Motor wird in der Prüfstation fixiert, so dass seine Position fest ist. Allerdings kann er um zwei Achsen verkippt sein. Dadurch entstehen in den Bildern Positionsabweichungen, die sich zudem von Perspektive zu Perspektive ändern. Die Verdrehung in den Bildern kann dabei vernachlässigt werden, aber durch die Größe des Motors führen auch die kleinen Winkel zu erheblichen Änderungen der Positionen von Prüfmerkmalen im Bild. Da die Positionsänderungen umso größer werden, je weiter der aufgenommene Motorenausschnitt von der Auflage entfernt ist, kann man keine globale Positionskorrektur verwenden, sondern muss für jedes Bild die Abweichung bestimmen.

Da die Prüfmerkmale mit Template Matching gesucht werden, könnte man versuchen, ohne vorherige Positionierung das Prüfmerkmal direkt in einem ausreichend großem Arbeitsbereich zu suchen. Dagegen spricht die auch in der Detailansicht noch relativ geringe Größe der Prüfmerkmale. Diese zwingt zum einen zu einer Suche mit einer sehr feinen Schrittweite, was entsprechend zeitaufwändig ist. Des Weiteren entstehen an den Oberflächen des Motors und der Komponenten Reflexe, die im ungünstigsten Fall zu Treffern beim Template Matching führen können und damit zu Fehlsegmentierungen.

Aus diesen Gründen wird eine Positions korrektur durchgeführt. Die Suche nach der Referenz für die Positions korrektur erfolgt ebenfalls mit Template Matching, wobei eine großflächige und in seiner Form unverwechselbare Referenz gewählt wird, wie beispielsweise die Schraube in Abbildung 9.2 links. Dabei muss es sich aber keineswegs um ein einzelnes Bauteil wie die Schraube handeln. Auch der Knick eines Rohres kann als Referenz verwendet werden, wie das rechte Bild in Abbildung 9.2 zeigt. Natürlich sollte es sich bei der Referenz um ein möglichst großflächiges Objekt mit wenig Details handeln, um eine große Schrittweite bei der Suche zu erlauben.

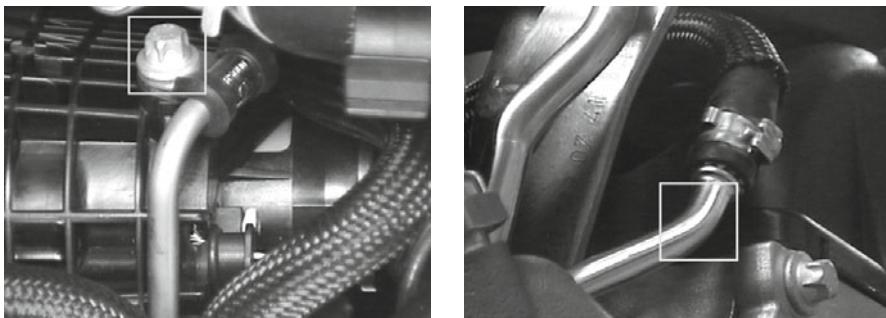


Abbildung 9.2: Verschiedene Positionsreferenzen

9.1.4 Objektdetektion

Wie oben beschrieben werden die Prüfobjekte mit Template Matching gefunden. Der naheliegende Ansatz ist dabei das korrekt verbaute Prüfobjekt als einziges Muster zu verwenden. Wenn dieses im Bild gefunden wird, wird aufgrund des Grades der Übereinstimmung zwischen dem Template und dem gefundenem Objekt entschieden, ob das vorliegende Prüfmerkmal korrekt verbaut wurde oder nicht. Diese Methode funktioniert in einer ganzen Reihe von Fällen sehr gut, es gibt aber Prüfmerkmale, bei denen der Unterschied zwischen richtiger und falscher Verbauung nur geringfügig verschiedene Korrelationswerte bewirkt. In diesen Fällen wird die Suche erweitert, indem nicht nur mit einem Muster eines korrekt verbauten Prüfobjekts gesucht wird, sondern auch mit dem Muster eines falsch verbauten. Abbildung 9.3 zeigt auf der linken Seite einen „korrekt“ verbauten Sensor und rechts einen „falsch“ verbauten. In beiden Fällen wird das Objekt mit einer ausreichenden Korrelation gefunden. Anhand des Musters mit dem der Sensor gefunden wird, kann sicher zwischen richtiger und falscher Verbauung unterschieden werden. D. h. es wird geprüft, dass der Sensor mit dem „korrektem“ Muster und einer ausreichend hohen Korrelation gefunden wird. Andernfalls liegt ein Fehler vor, da entweder das Objekt ganz fehlt oder falsch verbaut ist.

Programmhinweis. Zum Aussortieren von Objekten Anhand ihrer Klassifizierung bietet NeuroCheck den Prüfschritt *Klassen auswerten*. Bei aktivierter Solldatenüberprüfung kann man eine Liste der Klassen angeben, die akzeptiert werden sollen. Alle Arbeitsbereiche von Klassen die nicht in der Liste stehen führen zu einem Fehler. In unserem Fall darf also die NIO-Klasse nicht in die Liste der akzeptablen Klassen, so dass Arbeitsbereiche, die mit dem NIO-Muster gefunden werden, als Fehler gewertet werden. Mit der Einstellung Muster löschen werden als Fehler gewertete Arbeitsbereiche aus der Arbeitsbereichsliste entfernt, ohne dass der Prüfschritt mit Sollwertfehler abbricht.

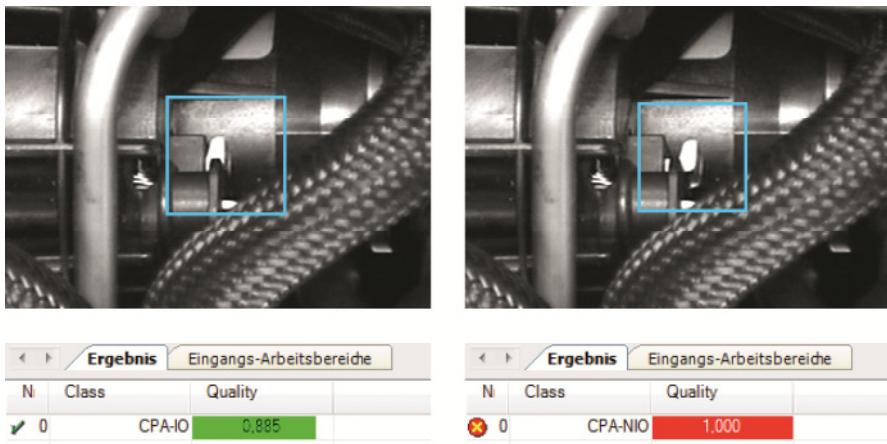


Abbildung 9.3: Richtig und falsch verbauter Sensor

9.1.5 Verifikation des Ergebnisses

Eigentlich ist die Prüfung bereits vollständig, allerdings kann man sich vorstellen, dass der Fall eintritt, dass bei der Suche nach dem Prüfmerkmal zum Beispiel ein Reflex gefunden wird. Die eine Möglichkeit dem zu begegnen ist eine höhere Qualität beim Template Matching zu fordern. Dies kann aber dazu führen, dass korrekt montierte Sensoren nicht mehr gefunden werden. Deshalb wird in diesem Fall ein anderer Weg beschritten. Für die Positionskorrektur haben wir bereits eine Referenz im Bild gesucht. Der Abstand zwischen dieser Referenz und dem Prüfmerkmal ist im Allgemeinen innerhalb geringer Toleranzen fest. Wenn die Suche nach dem Prüfmerkmal an einer völlig falschen Stelle einrastet, so weichen die Abstände zwischen Prüfmerkmal und Positionsreferenz erheblich von den normalen Werten ab. D. h. mit den Abständen in x- und y-Richtung zwischen Positionsreferenz und Prüfmerkmal hat man eine gute Möglichkeit, zu prüfen ob die Ergebnisse von Template Matching plausibel sind oder nicht. Im Prinzip prüft dies übrigens auch ob die Positionsreferenz an der richtigen Stelle eingerastet ist, denn auch dies führt zu einer Veränderung der Abstände zwischen dem Prüfmerkmal und der Positionsreferenz.

Da der Schraubenkopf und der Sensor bereits gefunden wurden, besteht diese Ergebnisverifikation lediglich aus der Berechnung der Abstände der beiden Objekte entlang der Koordinatenachsen und dem Vergleich mit den üblichen Abständen, von daher bedeutet die Verifikation kaum zusätzlichen Aufwand. Da es nur darum geht zu prüfen, dass die richtigen Objekte gefunden wurden, wird man weder an die Vermessung noch an die Toleranzprüfung besonders hohe Genauigkeitsanforderungen stellen. Ein Beispiel für die Ergebnisverifikation ist in Abbil-

dung 9.4 zu sehen, bei der der Abstand in x-Richtung zwischen dem Sensor und dem Schraubenkopf, der in diesem Fall die Positionsreferenz ist, geprüft wird.

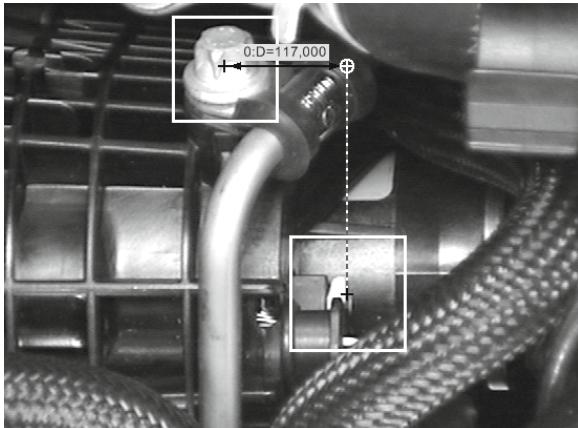


Abbildung 9.4: Verifikation des Ergebnisses durch Vermessung

Tabelle 9.1 zeigt das vollständige Prüfprogramm für ein Prüfmerkmal mit Positions korrektur, Objektsuche und Ergebnisverifikation.

Beispielbeschreibung 9.1. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 9.1 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Motorenpruefung
Beleuchtung	Diffuse, großflächige Beleuchtung.
Funktionen	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> <i>Bildaufnahme</i> <i>Arbeitsbereiche definieren</i>, setzt einen Arbeitsbereich in der Umgebung des Schraubenkopfes. <i>Template Matching</i>, sucht den Schraubenkopf als Referenz für die Positionierung der Suchbereiche. <i>Position bestimmen</i>, benutzt die gefundene Schraube als Positionsreferenz um Verschiebungen aufgrund von Verkippungen des Motors zu kompensieren. <i>Arbeitsbereiche definieren</i>, setzt den Arbeitsbereich für die Suche nach dem Sensor. <i>Arbeitsbereiche positionieren</i>, verschiebt diese Arbeitsbereiche entsprechend der Position des Schraubenkopfes. <i>Template Matching</i>, sucht den Sensor mit Mustern korrekter und fehlerhafter Montage. <i>Klassen auswerten</i>, sortiert das gefundene Objekt aus, falls der Fund mit einem Muster der fehlerhaften Montage gemacht wurde oder die Qualität der Korrelation zwischen Muster und Bild zu gering ist.

-
9. *Arbeitsbereiche Anzahl überprüfen*, prüft, dass ein Arbeitsbereich vorhanden ist, also ein korrekt verbauter Sensor gefunden wurde.
 10. *Arbeitsbereiche kombinieren*, führt die Positionsreferenz und den Sensor in einer Arbeitsbereichsliste zusammen für die Vermessung zur Verifikation des Ergebnisses.
 11. *Maße erzeugen*, misst die Abstände zwischen der Referenz und dem Sensor in x- und y-Richtung.
 12. *Maße Toleranzen überprüfen*, verifiziert, dass die Abstände zwischen Referenz und Prüfobjekt stimmen und keine Fehlsegmentierung statt gefunden hat.
-

9.2 Einfache Vermessung zur Montagekontrolle

Das Beispiel dieses Abschnitts zeigt die in der Einleitung angesprochene Verwendung von Vermessungsfunktionen als Hilfsmittel der Anwesenheitskontrolle. Diese Vorgehensweise ist typisch für Anwendungen, bei denen es, wie in diesem Beispiel, um die Überprüfung der korrekten Montage eines Produktes geht. Bei dem in Abbildung 9.5 gezeigten Prüfteil handelt es sich um eine Tropfkammer, die als Blutfilter in Dialysegeräten verwendet werden. Vor dem Auslass dieser Tropfkammern befindet sich ein Filter, dessen korrekte Montage von kritischer Bedeutung für Funktion und Sicherheit des Produktes ist.



Abbildung 9.5: Tropfkammer für Dialysegeräte

9.2.1 Beleuchtung

Abbildung 9.6 zeigt die Tropfkammer unter verschiedenen Beleuchtungen. Man sieht, dass eine sinnvolle Prüfung nur im Durchlicht möglich ist, wobei die Kanten in polarisiertem Licht deutlicher hervortreten. Für die endgültige Realisierung wurde der Durchtritt des Lichts auf den relevanten Prüfbereich begrenzt, um Stö-

rungen durch Seitenlicht zu minimieren. Das Ergebnis ist in Abbildung 9.8 zu sehen.

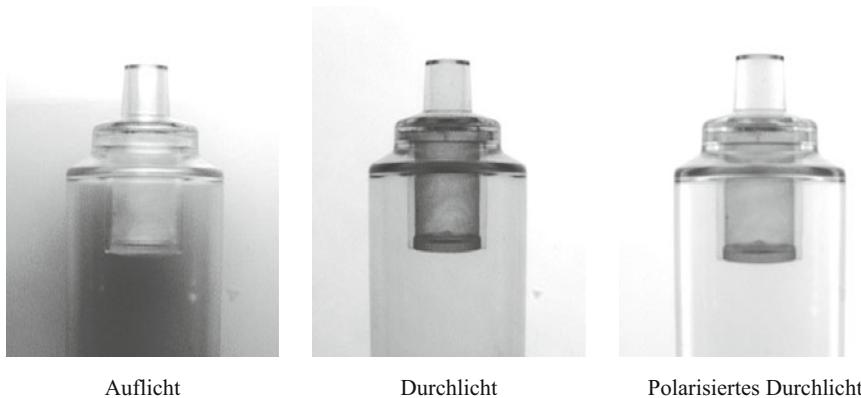


Abbildung 9.6: Tropfkammer in Auflicht, Durchlicht, und polarisiertem Durchlicht

Der prinzipielle Aufbau der Prüfanlage ergibt sich aus den Beleuchtungsanforderungen und der Notwendigkeit, in zwei rechtwinklig zueinander orientierten Blickrichtungen zu prüfen, auf die wir im folgenden Abschnitt gleich noch eingehen. Abbildung 9.7 zeigt diesen Aufbau. Die Tropfakkammern werden auf einem Band stehend zwischen zwei Kombinationen von Beleuchtungseinrichtung und Kameras hindurchgeführt, die im rechten Winkel zueinander angeordnet sind, um Schiefsitz des Filters in jeder Richtung erkennen zu können.

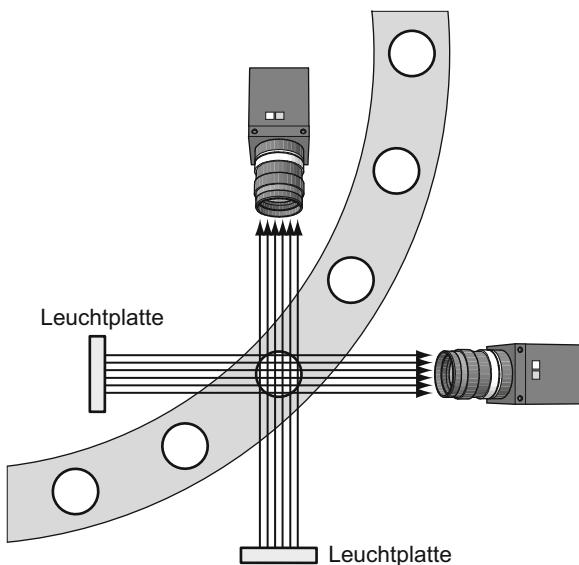


Abbildung 9.7: Prinzipaufbau der Anlage zur Prüfung des Blutfilters

9.2.2 Prüfkriterien

Der schwierigste Schritt bei einer Aufgabe aus der Montagekontrolle ist häufig die Festlegung, anhand welcher Eigenschaften die Korrektheit eines Prüfteils festgestellt werden kann, denn die intuitive Antwort eines Menschen, „das Teil müsse eben richtig aussehen“, hilft hier leider überhaupt nicht weiter. Man muss sich also zunächst einmal Klarheit darüber verschaffen, welche Fehler auftreten können und in welcher *quantifizierbaren* Weise sie die Eigenschaften des Prüfteils verändern.

In unserem Beispiel kann sich ein fehlerhafter Einbau des Filters auf folgende Weise äußern:

- Schiefsitzender Schutz, wodurch am oberen Rand eine Lücke entstehen könnte.
- Deformierter Schutz, in dem durch Materialspannungen Löcher entstehen können.

Abbildung 9.8 zeigt zwei im rechten Winkel zueinander aufgenommene Bilder eines schadhaften Filters. Man sieht, dass der Filter durch eine Deformation verkürzt werden kann, so dass der Abstand zwischen dem Boden des Schutzes und der Mündung der Tropfkammer zu gering wird. Durch den schießen Sitz sind die Abstände von den Außenkanten der Tropfkammer verschieden.

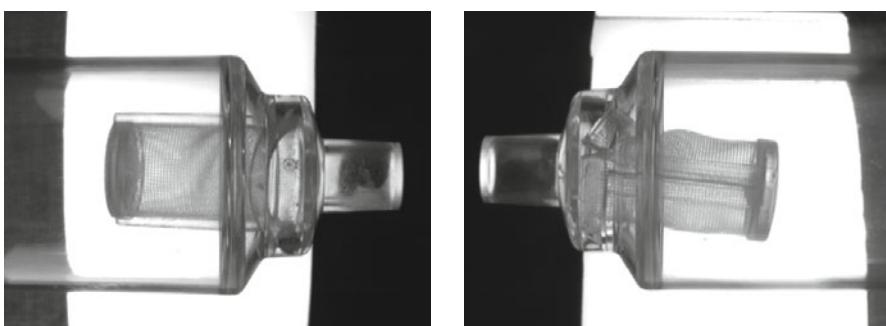


Abbildung 9.8: Schadhafter Filter aus zwei rechtwinklig versetzten Blickrichtungen

Abbildung 9.8 macht auch deutlich, warum zwei Aufnahmen im rechten Winkel zueinander erforderlich sind: Nur so kann man den Fall ausschließen, dass eine Verbiegung des Filters gerade in Richtung der optischen Achse einer Kamera übersehen wird.

Aus den oben aufgeführten möglichen Defekten des Filters ergibt sich, dass die folgenden Aspekte zu prüfen sind:

- Filterlänge
- Zentrierung
- Ansatzdurchmesser

Im Folgenden zeigen wir, wie diese einzelnen Größen festzustellen sind.

Filterlänge: Die Länge des Filters, also der Abstand zwischen Filterboden und Auslass der Tropfkammer lässt sich einfach dadurch messen, dass der untere Bereich des Filters als Objekt erzeugt wird und die obere Kante der Tropfkammermündung. Wie man in Abbildung 9.8 deutlich sieht, ist diese Kante infolge der Reflexionen in dem transparenten Kunststoff vor der dunklen Abdeckung gut erkennbar. Sowohl die Kante als auch der Boden des Filters können leicht binär segmentiert werden, wobei allerdings individuelle Binärschwellen erforderlich sind. Abbildung 9.9 zeigt die verwendeten Arbeitsbereiche für das Bild von der ersten Kamera: Eine Suchlinie wird zur Antastung der Kante verwendet, während der Filterboden in einem rechteckigen Suchbereich erzeugt werden kann.

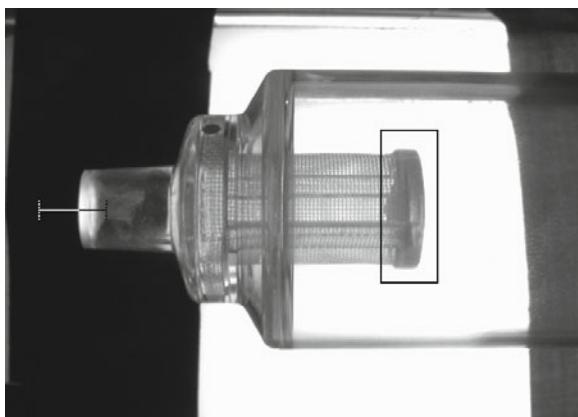


Abbildung 9.9: Suchbereiche für die Erzeugung von Filterboden und Auslasskante

Zentrierung: Die Aufgabe, zu überprüfen, ob der Filter verkippt eingesetzt wurde, könnte nahe legen, die Richtung der Filterkanten zu messen. Da kleine Verformungen aber durchaus zulässig sind, könnte der Winkel durch Unregelmäßigkeiten der Außenkontur verändert werden. Dies ist daher kein geeignetes Maß für den Sitz des Filters. Die zuverlässigste Methode zur Prüfung, ob der Filter schräg montiert oder verbogen ist, besteht darin, die Abstände der Filterkanten zu den Außenkanten der Tropfkammer zu messen. Dabei ist es von Vorteil, nicht jeden Abstand einzeln auf die Einhaltung einer bestimmten Toleranz zu überprüfen, sondern die Differenz der beiden Abstände mit Null zu vergleichen. So erhält man das stabilste, gegenüber Beleuchtungs- und Positionsschwankungen unempfindlichste Maß.

Zur Bestimmung der beiden Abstände kann man vier entsprechend Abbildung 9.10 angeordnete Suchlinien verwenden. Auf jeder wird eine Hell-Dunkel-Kante ermittelt, je zwei dieser Kanten können dann zu einem Abstandsmaß verknüpft werden. Zusätzlich zu diesen Suchlinien enthält Abbildung 9.10 noch zwei Suchbereiche, die zur Messung der Breite des Filters am oberen sichtbaren Ende dienen. Da man ihn dort vom Tropfkammerkorpus optisch kaum trennen kann, ist es am einfachsten, die an den Filter grenzenden Hintergrundbereiche als Objekte zu

erzeugen und deren Abstand zu messen. Man arbeitet also gewissermaßen mit einer „Negativmessung“: Statt der Breite eines realen Objekts misst man die Unterbrechung, die es im Hintergrund verursacht.

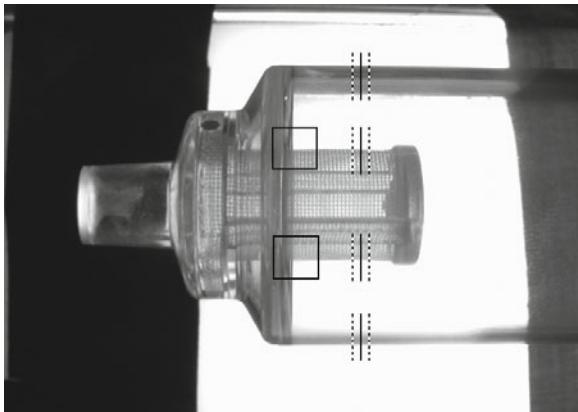


Abbildung 9.10: Suchlinien zur Messung der Ausrichtung des Filters

9.2.3 Objekterzeugung und Maßberechnung

Sämtliche Objekte, die für die Messung der in Abschnitt 9.2.2 beschriebenen Kriterien benötigt werden, können in einem Durchgang segmentiert werden. Wegen der unterschiedlichen Helligkeiten der verschiedenen Kanten sind aber individuelle Binärschwellen in jedem Arbeitsbereich für eine stabile Segmentierung empfehlenswert. Abbildung 9.11 zeigt die separat binarisierten Arbeitsbereiche in dem ansonsten unveränderten Grauwertbild sowie daneben die segmentierten Objekte, schwarz in das Grauwertbild eingezeichnet (die Kantenobjekte wurden zur besseren Erkennbarkeit verdickt). Beachten Sie, dass ober- und unterhalb des vorderen Filterbereiches der helle Hintergrund als Objekt erzeugt wurde.

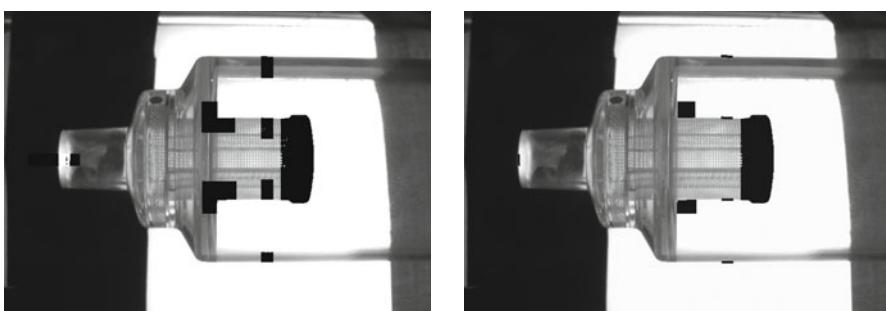


Abbildung 9.11: Individuell binarisierte Arbeitsbereiche und segmentierte Objekte

In Abbildung 9.12 sind die zur Überprüfung verwendeten Maße für ein einwandfreies und ein fehlerhaftes Prüfteil eingezeichnet. Der Abstand zwischen Tropfkammermündung und Filterboden wird in 0° -Richtung gemessen und ist mit null bezeichnet. Die anderen Maße werden alle in 90° -Richtung gemessen. Die Festlegung der Messrichtung ist wichtig, damit keine Verfälschungen durch Unregelmäßigkeiten der Konturen verformter Filter auftreten. Beachten Sie, dass die Abstände an verschiedenen Stellen der Objekte gemessen werden. Das hängt damit zusammen, dass jeweils Minimal- oder Maximalabstände gemessen werden. Bei der Breite des Filters (Maß drei) ist der Minimalwert entscheidend, um Einrisse festzustellen; bei der Gesamtlänge (Maß null) muss der Maximalwert benutzt werden, um sicher bis zum Filterboden zu messen.

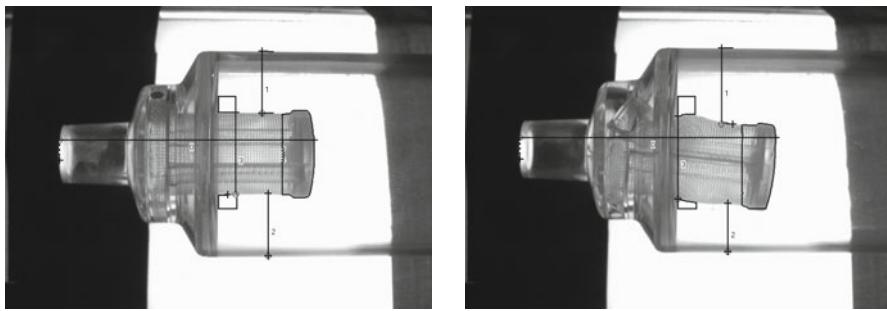


Abbildung 9.12: Zur Überprüfung des Filters benötigte Maße

Zuletzt müssen dann noch die Differenz der in Abbildung 9.12 mit eins und zwei bezeichneten Abstände zwischen den Kanten von Filter und Tropfkammer berechnet und alle Werte mit den entsprechenden Sollvorgaben verglichen werden. Bei dem in Abbildung 9.12 gezeigten fehlerhaften Teil ist es die Differenz zwischen den seitlichen Abständen, die den Ausschlag gibt. Sie ist um etwa einen Faktor 25 größer als beim einwandfreien Prüfteil.

9.2.4 Positionskorrektur

Einen Aspekt haben wir bei der bisherigen Betrachtung dieser Anwendung noch außer Acht gelassen. Die Prüfteile stecken auf Haltern, die auf dem Transportband angebracht sind. Durch die Elastizität des Kunststoffes kann es vorkommen, dass ein Prüfteil nicht ganz gerade auf dem Halter sitzt. Diese Neigungen betragen nur wenige Grad, so dass es nicht nötig ist, die Messrichtungen von 0° bzw. 90° nachzuführen. Sie haben aber durchaus einen Einfluss auf die Positionierung der Arbeitsbereiche. Das gilt besonders für die Suchlinie, mit der die Kante am oberen Abschluss der Tropfkammer erzeugt wird, da bei einer Gesamtlänge des Prüfsteils von nahezu 15 cm eine Neigung von drei Grad immerhin eine Seitenabweichung

von acht Millimeter hervorruft. Da der zu prüfende Ausschnitt im Verhältnis zur Länge des Prüfteils kurz ist, wird keine Rotationskorrektur benötigt, man kann die Arbeitsbereiche einfach aufgrund der Position einer Seitenkante der Tropfkammer nachschieben. Voraussetzung ist nur, dass diese Suchlinie hinreichend lang ist. Abbildung 9.13 zeigt die Suchlinie und das erzeugte Referenzobjekt. Als Referenz wird hier die untere Kante benutzt, da sie sehr schön glatt und stabil ist.

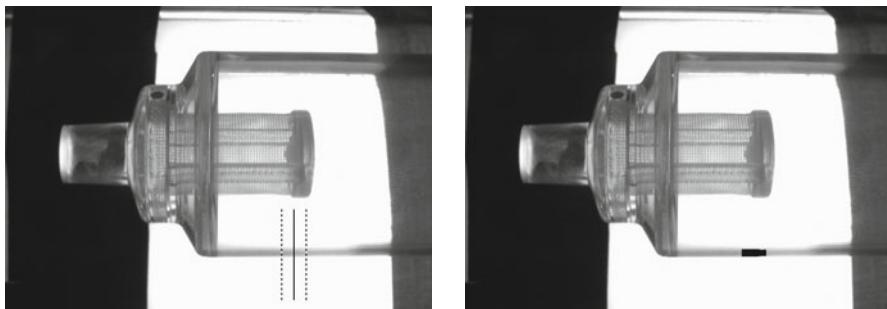


Abbildung 9.13: Suchbereich und Referenzobjekt für Positions korrektur

Tabelle 9.2 gibt eine Zusammenfassung der gesamten Prüfung für eine der beiden Kameras. Abgesehen von der unterschiedlichen Anordnung der Arbeitsbereiche sind die Prüfungen für beide Ansichten genau identisch.

Beispielbeschreibung 9.2. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 9.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Tropfkammer
Beleuchtung	Polarisierte Hintergrundbeleuchtung mit seitlichen Abdeckungen zur Vermeidung von Streulicht
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt eine Suchlinie für die in Bewegungsrichtung vordere Kante des Prüfteils. 3. <i>Binärschwelle erzeugen</i>, optimiert die Schwelle anhand einer Histogrammalyse des Arbeitsbereiches. 4. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert ein Stück der Tropfkammerkante. 5. <i>Arbeitsbereiche Position bestimmen</i>, benutzt die untere (oder obere, je nach Kamera) Kante dieses Ausschnitts als Positionsreferenz. 6. <i>Arbeitsbereiche definieren</i>, setzt die in den Abbildungen 9.13 und 9.14 dargestellten Arbeitsbereiche. 7. <i>Arbeitsbereiche positionieren</i>, verschiebt diese Arbeitsbereiche entsprechend der Position der Tropkammerkante.

-
8. *Binärschwelle erzeugen*, berechnet individuelle Segmentierungsschwellen für jeden der positionierten Arbeitsbereiche.
 9. *Arbeitsbereiche binär erzeugen*, segmentiert die einzelnen Kanten- und Ausschnittsobjekte.
 10. *Maße erzeugen*, berechnet folgende Maße:
 - 0°-Abstand zwischen Kammermündung und Filterboden
 - 90°-Abstand zwischen oberer Kante der Kammer und oberer Kante des Filters
 - 90°-Abstand zwischen unterer Kante der Kammer und unterer Kante des Filters
 - Minimaler 90°-Abstand zwischen den Hintergrundausschnitten ober- und unterhalb des Filters
 11. *Maße verknüpfen*, berechnet die Differenz zwischen den oberen und unteren Abständen zwischen Filter und Kammerwand.
 12. *Maße Toleranzen überprüfen*, vergleicht die berechneten Maße mit den Sollvorgaben.
-

9.3 Anwesenheitskontrolle mithilfe von Klassifikatoren

Das nächste Beispiel stammt aus der Automobilindustrie. Es handelt sich dabei um Befestigungsflansche, die mit einer verstemmten Mutter auf der Welle gesichert sind. Dieses Prüfteil bildet die Basis für das Beispiel dieses Abschnitts, in dem eine Typ- und Montagekontrolle durch das Zusammenwirken von Methoden aus den Bereichen der einfachen Merkmalsüberprüfung, der Vermessung und der Klassifikation durchgeführt wird.



Abbildung 9.14: Befestigungsflansch mit noch unverstemmter Mutter

Mit der Sichtprüfung werden zwei Ziele vorfolgt:

1. Sicherstellung der Montage des richtigen Flanschtyps; die verschiedenen Typen unterscheiden sich nach der Anzahl der Arme des Flansches und deren Länge, bzw. des Abstandes der Befestigungsbohrungen vom Zentrum des Flansches.
2. Überprüfung der Verstemmung der Mutter im Zentrum des Flansches; diese Verstemmung fixiert den Flansch in einer bestimmten Stellung auf der Welle. Wenn die Verstemmung fehlt oder, beispielsweise durch Abrutschen des Verstemmwerkzeuges, nicht korrekt ausgeführt ist, kann diese Fixierung nicht gewährleistet werden.

9.3.1 Beleuchtung

Wie so oft bei der Lösung einer Bildverarbeitungsaufgabe ist auch hier die erste Aufgabe der Entwurf einer geeigneten Beleuchtung. Die entscheidende Frage ist in diesem Fall, wie die Verstemmung sichtbar zu machen ist, die bei weitem schwieriger zu identifizieren ist als die Flanscharme und Gewindebohrungen. In der Draufsicht ist der Unterschied zwischen einer verstemmten, unverstemmten oder fehlverstemmten Mutter für einen Menschen zwar aufgrund der Form deutlich zu erkennen, für eine sichere maschinelle Identifikation müssen aber Kontrastverhältnisse hergestellt werden, unter denen eine vorhandene Verstemmung deutlich hervortritt.

Eine korrekt erfolgte Verstemmung erzeugt eine Fläche, die im Gegensatz zu allen anderen Flächen an der Mutter nicht entweder senkrecht oder waagerecht orientiert ist. Man braucht in diesem Fall eine Beleuchtung, die Unterschiede in der Orientierung von Oberflächen herausarbeitet, also eine gerichtete Beleuchtung. Selbstverständlich lässt sich die Verstemmung nicht gleichzeitig von den senkrechten und waagerechten Oberflächen trennen, aber wenn man erreichen kann, dass sich die waagerechten Flächen einerseits, die senkrechten Flächen und die Verstemmung andererseits deutlich voneinander unterscheiden, hat man hervorragendes Ausgangsmaterial für die weitere Untersuchung.

Eine weitere Forderung an die Beleuchtung ergibt sich daraus, dass das Prüfteil rotationssymmetrisch ist und nicht bekannt ist, an welcher Stelle sich die Verstemmung befinden wird. Man muss also den gesamten Umfang prüfen. Die Beleuchtung muss daher allseitig gleich sein. Die Lösung ist eine große Ringleuchte, die von allen Seiten flach in das Teil hineinstrahlt und alle waagerechten Flächen sehr hell erscheinen lässt, während die senkrechten Kanten der Mutter genau wie die Verstemmung deutliche Schatten erzeugen. Abbildung 9.15 zeigt schematisch den Beleuchtungseffekt. Die Wirkung der Beleuchtung ist stark vereinfacht dargestellt, um das Prinzip zu verdeutlichen, die tatsächlichen Verhältnisse sind aufgrund der Oberflächenstruktur des Materials und der allseitigen Beleuchtung natürlich komplizierter. Der Übersichtlichkeit halber wurden die reflektierten Strahlen nur an drei signifikanten Stellen eingezeichnet, wobei die Auffächerung

in mehrere Strahlen andeutet, dass die Oberfläche kein idealer Spiegel ist, sondern das Licht mit einer gewissen Vorzugsrichtung streut. Man erkennt, dass

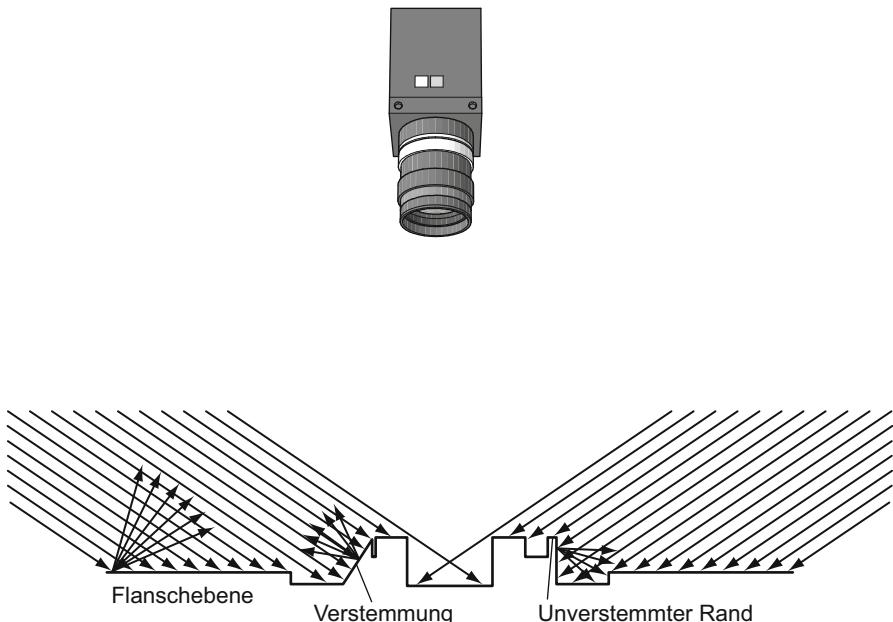


Abbildung 9.15: Beleuchtung der Mutter

- das Licht von waagerechten Flächen vorzugsweise in Richtung der Kamera reflektiert wird. Diese erscheinen also hell;
- das Licht steigender, senkrechter Kanten kaum in Richtung der Kamera reflektiert wird. Diese erscheinen daher dunkel;
- hinter fallenden, senkrechten Kanten Schattenbereiche entstehen;
- das Licht von der schrägen Verstemmungsfläche ebenfalls kaum in Richtung Kamera reflektiert wird, wobei die Neigung der Verstemmungsfläche dem Licht eine größere Angriffsfläche bietet als eine senkrechte Kante. Dieser dunkle Bereich erscheint daher verbreitert.

Die oben beschriebene Wirkung des Beleuchtungsaufbaus lässt sich im Kamerabild gut nachvollziehen. Abbildung 9.16 zeigt eine einwandfrei verstemmte und eine unverstemmte Mutter. Die waagerechten Flächen erscheinen durch die Überstrahlung absolut weiß, die senkrechten Kanten deutlich dunkler. Allerdings sind diese an einigen Stellen durch die Überstrahlung unterbrochen. Im Bild der verstemmten Mutter ist die breite, sich entsprechend der Form der Verstemmung nach oben verjüngende dunkle Fläche gut zu erkennen, während im Bild der unverstemmten Mutter nur eine leichte Verdickung am Ansatzpunkt des Verstemmwerkzeuges vorhanden ist. Ebenfalls deutlich erkennbar ist der breite abgeschatte-

te Ring, der ein sehr stabiles Bildelement darstellt, das sich ausgezeichnet für Positionierungszwecke eignet.

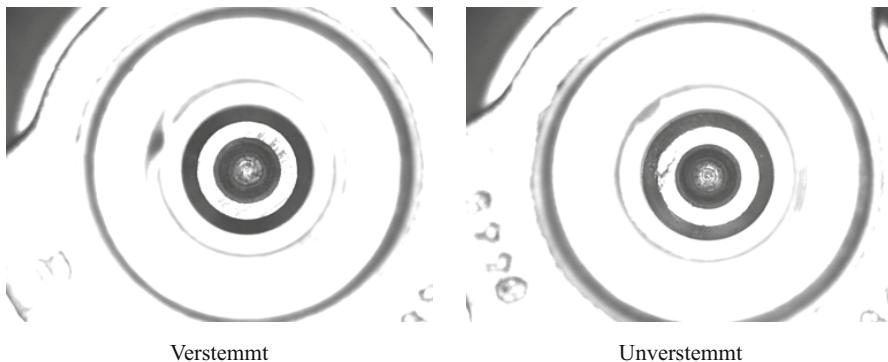


Abbildung 9.16: Mutter unter Ringbeleuchtung, links verstemmt, rechts unverstemmt

Man sieht an dieser Anwendung einmal mehr, dass eine gut konstruierte Beleuchtung nahezu die Hälfte der Lösung ausmacht. Andererseits wird auch deutlich, dass man schon beim Entwurf der Beleuchtung eine gewisse Vorstellung davon haben muss, wie die Prüfung selbst realisiert werden kann, damit die dafür notwendigen Elemente gut sichtbar sind. Wir werden nachher noch sehen, dass eine Positions korrektur für diese Anwendung von großer Wichtigkeit ist, daher liegt ein Vorteil der gewählten Beleuchtungsmethode auch darin, dass sie stabile Referenzen für die Positionierung erzeugt.

Noch ein Wort zur technischen Realisierung der Ringbeleuchtung. Man kann große ringförmige Leuchtstoffröhren erhalten, aber sie bieten durch ihren festgelegten Durchmesser wenig Flexibilität und liefern auch keine völlig gleichmäßige Rundumbeleuchtung, da sie an einer Stelle einen Zünder haben müssen. Aus diesem Grund wurde die Ringbeleuchtung hier aus einzelnen roten LED-Scheinwerfern aufgebaut, die noch dazu den Vorteil hoher Lebensdauer haben und in einem Wellenlängenbereich arbeiten, für den CCD-Sensoren sehr empfindlich sind.

9.3.2 Kontrolle der Verstemmung

Vorverarbeitung. Die als dünne, dunkle Linien im Kamerabild erscheinenden senkrechten Kanten eignen sich nicht für Positionierungszwecke oder sonstige Auswertungen, da sie infolge der Überstrahlung teilweise unterbrochen sein können – wobei durch die nicht ganz exakte Positionierung des Prüfteils im Zentrum der Ringbeleuchtung nicht bestimmt ist, wo diese Unterbrechungen auftreten. Für die Prüfung stellen sie daher einen Störeffekt dar, dessen Einfluss sich durch Vorverarbeitung mit einem Dilatationsfilter verringern lässt, wie Abbildung 9.17 zeigt.

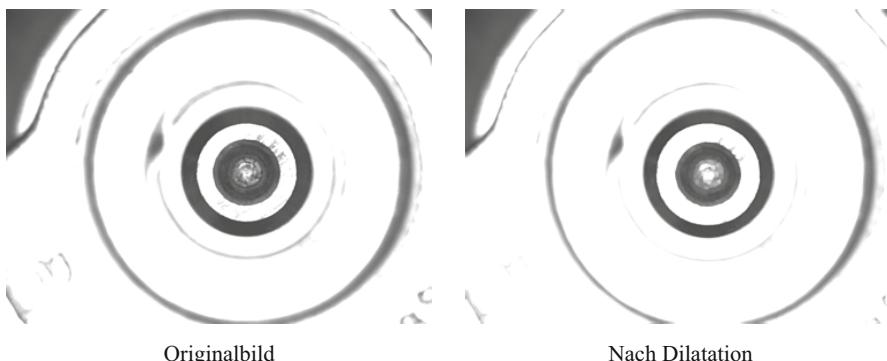


Abbildung 9.17: Mutter vor und nach Grauwertdilatation

Bestimmung der Positionsreferenz. Der nächste Schritt ist die Definition eines Arbeitsbereiches innerhalb der äußeren Begrenzung der Mutter. Darin kann leicht eine Binarisierung auf der Basis einer Histogrammanalyse vorgenommen werden, wenn man berücksichtigt, dass sich durch die Überstrahlung ein erheblicher Teil der Bildpunkte auf dem Grauwert 255 konzentriert. Man darf hier also die in Abschnitt 4.2.3 beschriebene Unterdrückung von Artefakten durch Ausblendung extremer Grauwerte nicht vornehmen. Abbildung 9.18 zeigt den definierten Arbeitsbereich und das zugehörige Histogramm. Wie immer bei Anwendungen mit einer Positionsunsicherheit muss dieser erste Arbeitsbereich groß genug sein, um die relevanten Objekte bei maximaler Positionstoleranz zu enthalten. Man wählt ihn dabei aber so klein wie möglich, einerseits um Suchzeit zu sparen, andererseits um so wenig Zusatzobjekte wie möglich zu erfassen, die hinterher ausgefiltert werden müssten. Der breite Schattenring in der Mitte der Mutter ist jetzt leicht als Referenzobjekt für die Positionierung zu segmentieren. Dabei können Teile der senkrechten Kanten, die nicht vollständig durch die Dilatation unterdrückt wurden, ebenso wie die Verstemmung als Zusatzobjekte auftreten. Die Objektgrößen sind allerdings so unterschiedlich, dass man schon während der Konturerzeugung für eine Vernachlässigung aller Objekte außer dem gesuchten Schattenkreis sorgen kann. Sicherheitshalber bietet sich dennoch eine Überprüfung anhand von Merkmalen wie Umfang, Fläche, Kompaktheit und Höhen-/Breitenverhältnis an, die sich für die lang gestreckte Störobjekte signifikant von den Merkmalswerten des Schattenkreises unterscheiden. Abbildung 9.19 zeigt dies anhand einer unverstemmten Mutter, bei der alle Objekte erzeugt wurden. Umfang und Fläche des Schattenkreises sind deutlich höher als die entsprechenden Werte der Störobjekte, ebenso die Kompaktheit dieses nahezu ideal kreisförmigen Objekts, und die Ratio liegt stets dicht bei eins.

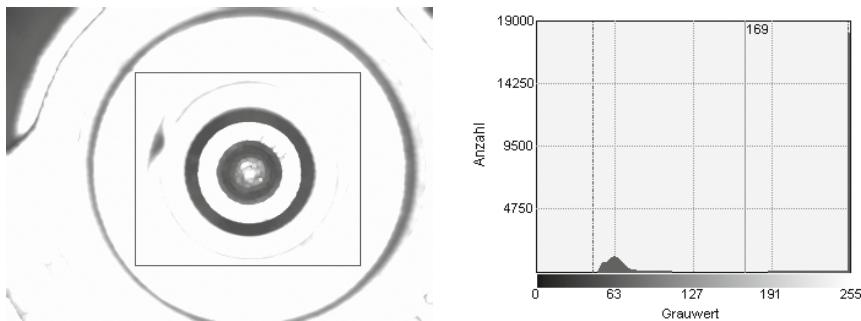


Abbildung 9.18: Arbeitsbereich für Positionierung an der Mutter (schwarzes Rechteck) und zugehöriges Histogramm

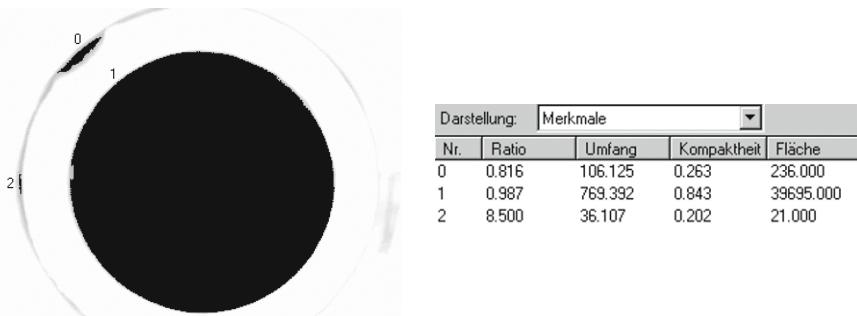


Abbildung 9.19: Schattenkreis mit Störobjekten und zugehörige Merkmale

Es zeigt sich, dass der innere Rand des Schattenkreises die glattere und stabilere Kontur darstellt, daher wird innerhalb des kreisförmigen Objekts noch einmal nach einem hellen Objekt gesucht, das dann – nach einem ähnlichen Auswahlprozess anhand entsprechender Merkmale – letztlich als Positionsreferenz dient.

Kreisabwicklung: Worin liegt jetzt der Sinn dieser komplexen Prozedur zur Bestimmung eines Referenzobjekts? Das Hauptproblem dieser Prüfaufgabe besteht darin, dass nicht feststeht, in welcher Winkelposition die Verstemmung zu finden ist. Es muss also der gesamte Umfang abgesucht werden.

Andererseits können sich, wie man in Abbildung 9.19 schon gesehen hat, im gleichen Abstand vom Zentrum der Mutter eine Reihe von Störobjekten befinden, die anhand einfacher Merkmale nur schwer von der Verstemmung zu unterscheiden sind. Man sollte also gleich ein Objekt der passenden Form suchen, wozu sich das Template-Matching-Verfahren anbietet. Je nach Winkelposition verändert sich aber diese Form, was zwei Möglichkeiten eröffnet:

- Man kann für eine Vielzahl von Winkelpositionen eigene Templates anlegen (oder diese intern berechnen). Der Nachteil ist, dass der Rechenaufwand für die Korrelation mit der Anzahl der Templates erheblich ansteigt. Dabei muss man

auch noch berücksichtigen, dass ohnehin mehrere Templates erforderlich sind, da die Verstemmung nicht immer exakt die gleiche Form aufweist.

- Man kann den Kreisring, innerhalb dessen sich die Verstemmung befinden muss, in ein rechteckiges Bild umwandeln. Durch den großen Radius dieses Kreisrings halten sich die Diskretisierungsprobleme in Grenzen und der Rechenaufwand fällt nur einmal an.

Zur Durchführung einer solchen Abwicklung eines Kreisrings in ein Rechteck wird eine Kreislinie um den Mittelpunkt des Kreisrings definiert und in bestimmten Winkelabständen die Senkrechten auf diesen Kreis berechnet. Dann können die Grauwerte zu beiden Seiten des Kreises parallel zu dieser Normalenrichtung interpoliert und spaltenweise in ein neues, rechteckiges Bild geschrieben werden. Abbildung 9.20 zeigt die Definition des Kreises, die sich an der Schattenkante der Mutter orientiert und das Ergebnis der Abwicklung (zur besseren Erkennbarkeit wurde der Kontrast verstärkt und der überstrahlte Hintergrund grau eingefärbt).

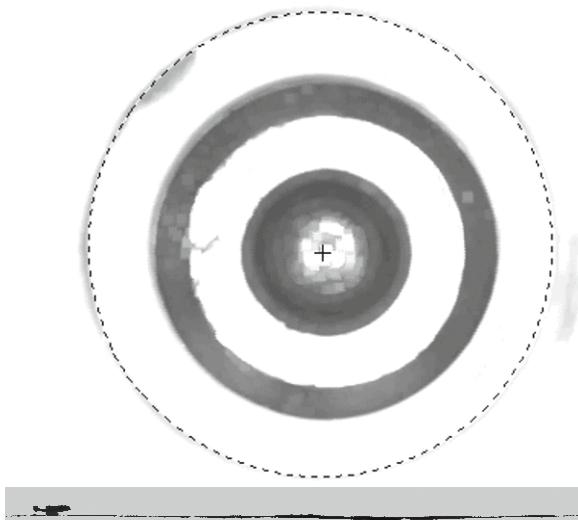


Abbildung 9.20: Kreisdefinition und Ergebnis der Abwicklung

Man kann dieses Verfahren prinzipiell auf völlig beliebige Konturen anwenden, aber es ist von Vorteil, einen manuell definierten Kreis zu benutzen, da dann die Größe und Form exakt festliegen. Besonders wichtig ist aber, dass der abgewickelte Kreis sehr genau positioniert sein muss. Abbildung 9.21 zeigt den Effekt der eintritt, wenn sich Mittelpunkt des Kreises und Mittelpunkt der Mutter nicht decken. Das abgewickelte Bild erhält dann eine deutliche Welle, wodurch sich auch die Form einer korrekten Verstemmung möglicherweise bis zur Unkenntlichkeit verändert.

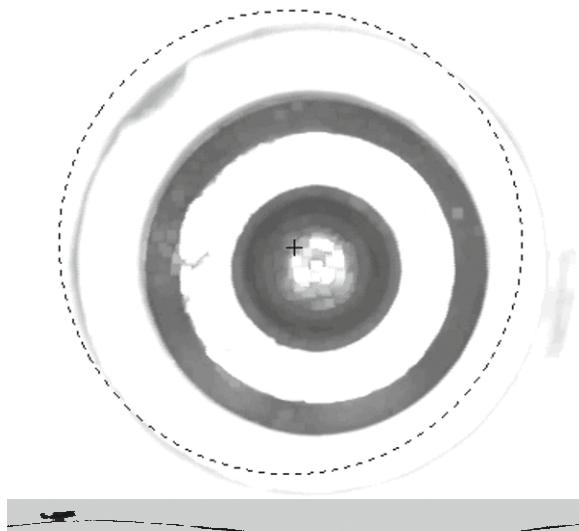


Abbildung 9.21: Kreisdefinition und Ergebnis der Abwicklung bei Fehlpositionierung

Ein Problem stellt sich noch bei der Kreisabwicklung: Die Abwicklung muss, im Gegensatz zu der Kreislinie als solcher, einen definierten Anfangs- und Endpunkt haben. Wenn dieser gerade mitten in der Verstemmung liegt, wird sie beim Abwickeln in zwei Teile zerschnitten, deren Form sich vom gesuchten Objekt so drastisch unterscheidet, dass keine Segmentierung mehr möglich ist. Es ist unmittelbar einleuchtend, dass man dies mit Sicherheit vermeiden kann, wenn man den Kreis 50 % „überwickelt“, also die ersten 180° hinten noch einmal anhängt. Auf diese Weise ist absolut sicher, dass sich die Verstemmung genau einmal unzerschnitten im Bild befinden muss.

Segmentierung und Bewertung der Verstemmung: Aufgrund ihrer kompakten Form ist die Verstemmung problemlos mit Template Matching zu finden. Es ist allerdings empfehlenswert, auch Bilder fehlerhafter Verstemmungen und unverstemmter Teile in den Templatesatz mit aufzunehmen und die Positionen der Korrelationspunkte entsprechend der Teileposition zu optimieren. Abbildung 9.22 zeigt einige solche Templates.



Abbildung 9.22: Templates für die Suche nach der Verstemmung

Die Prüfaufgabe kann jedoch noch nicht als gelöst betrachtet werden. Die Fähigkeiten des Template-Matching-Verfahrens liegen hauptsächlich auf dem Gebiet

der Segmentierung, nicht der Klassifikation. Es ist damit also leicht möglich, eine einwandfreie Verstemmung zu finden, sie aber gegen eine schlechte Verstemmung oder den Werkzeugansatz abzugrenzen ist erheblich schwieriger. Ein eindimensionales Beispiel möge dies verdeutlichen. Abbildung 9.23 zeigt das prinzipielle Aussehen von Schnitten durch die Verstemmung. Der Einfachheit halber ist dabei angenommen, dass nur zwei Grauwerte vorkommen, 255 und 175. Der schmale dunkle Bereich von nur vier Pixeln Breite entspricht einem etwas breit aus gefallenen Werkzeugansatz, hier liegt also keine Verstemmung vor, während der zu breite dunkle Bereich eine fehlerhafte Verstemmung repräsentiert.

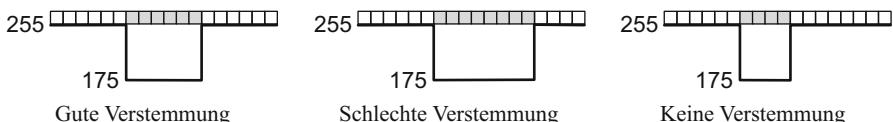


Abbildung 9.23: Abgrenzungsproblem beim Template Matching

Wenn man jetzt die Korrelationskoeffizienten des Beispielprofils für die gute Verstemmung mit den beiden anderen Fällen ausrechnet, findet man eine Korrelation von 0,79 mit der schlechten Verstemmung, von 0,76 mit dem bloßen Werkzeugansatz. Die realen Fälle sind sich zwar nicht ganz so ähnlich, aber dafür erstrecken sie sich in zwei Dimensionen, und man kann wegen der nie ganz gleichen Form der korrekten Verstemmung keine besonders hohen Mindestkorrelationen fordern. Es ist also notwendig, das gefundene Objekt danach noch einmal zu bewerten, wozu man einfach das Bild des Objekts einem Klassifikator vorlegen kann. Abbildung 9.24 zeigt das Ergebnis dieser Klassifikation anhand eines 16×7 Punkte großen Rasterbildes und des mittleren Grauwertes innerhalb des umschreibenden Rechtecks des Objekts, der bei korrekt verstemmten Muttern deutlich niedriger ist als bei unverstemmten.



Abbildung 9.24: Gerasterte und klassifizierte Verstemmungen

Damit ist der erste Teil des Prüfproblems gelöst, die Überprüfung der korrekten Verstemmung. Tabelle 9.3 fasst den Prüfablauf zusammen. Man sieht dort, dass es sich um eine relativ langwierige Prüfung handelt. Ausschlaggebend für die Prüfzeit sind die Prüfschritte *Bild filtern* mit der rechenaufwändigen Dilatation, *Arbeitsbereich abwickeln*, mit der Interpolation entlang des Kreisrings und *Template Matching*, das hier mit einer geringen Schrittweite arbeiten muss, um die schwan-

kende, schlecht bestimmte Form der Verstemmung ausreichend genau detektieren zu können.

9.3.3 Typkontrolle des Flansches

Wir kommen nun zum zweiten Teil dieser Prüfaufgabe, der Bestimmung des Flanschtyps. Die verschiedenen Flanschtypen unterscheiden sich nach der Anzahl der Befestigungsarme und deren Länge. Abbildung 9.25 zeigt einen vollständigen dreiarmigen Flansch. In der Mitte ist die korrekt verstemmte Mutter zu erkennen. Man sieht, dass der Flansch unter derselben Beleuchtung untersucht werden kann wie die Mutter, allerdings muss man dazwischen die Kamera umschalten, da man ein erheblich größeres Sichtfeld benötigt.

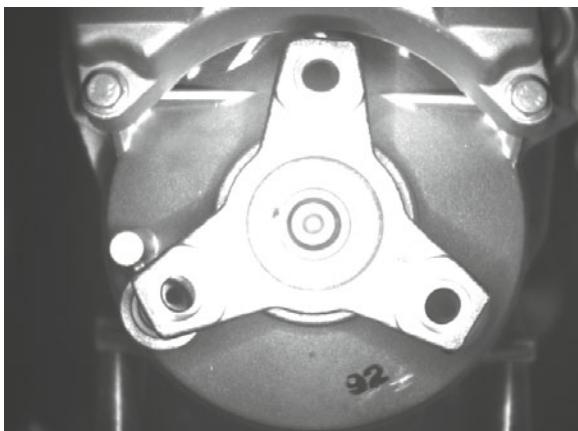


Abbildung 9.25: Dreiarmer Flansch

Beispielbeschreibung 9.3. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 9.3.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Flansch
Beleuchtung	Flach einfallende Ringbeleuchtung, aus LED-Scheinwerfern zusammengesetzt, mit starker Überstrahlung.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> <i>Bildaufnahme</i> <i>Bild filtern</i>, führt eine Dilatation zur Beseitigung dünner Störkanten durch. <i>Arbeitsbereiche definieren</i>, setzt ein Rechteck im ungefähren Zentrum des Flansches. <i>Binärschwelle erzeugen</i>, optimiert die Schwelle anhand einer Histogrammanalyse des Arbeitsbereiches.

-
5. *Arbeitsbereiche binär erzeugen*, segmentiert den Schattenkreis im Zentrum des Flansches. Dieser kann durch Merkmale erzeugen und Arbeitsbereiche filtern noch auf verschiedene Eigenschaften überprüft werden.
 6. *Arbeitsbereiche binär erzeugen*, segmentiert den hellen Innenraum des Schattenkreises, da dieser die stabilere Kontur aufweist. Ebenfalls Überprüfung durch Merkmale erzeugen und Arbeitsbereiche filtern möglich.
 7. *Arbeitsbereiche Position bestimmen*, verwendet den Schwerpunkt des hellen Innenbereichs als Positionsreferenz.
 8. *Arbeitsbereiche definieren*, setzt einen Kreis mit dem Durchmesser der Verstemmkkante.
 9. *Arbeitsbereiche positionieren*, zentriert diesen Kreis auf den Schwerpunkt des hellen Innenbereiches.
 10. *Arbeitsbereich abwickeln*, transformiert einen Streifen beiderseits des positionierten Kreises in ein langes, rechteckiges Bild.
 11. *Arbeitsbereiche definieren*, erklärt das gesamte abgewickelte Bild zum Suchbereich.
 12. *Template Matching*, sucht die Verstemmung innerhalb des abgewickelten Bildes.
 13. *Arbeitsbereiche Merkmale erzeugen*, berechnet den mittleren Grauwert innerhalb des umschreibenden Rechtecks der gefundenen Verstemmung.
 14. *Arbeitsbereiche rastern*, erzeugt aus der gefundenen Verstemmung ein Eingangsbild für den Klassifikator.
 15. *Arbeitsbereiche klassifizieren*, bestimmt anhand des mittleren Grauwertes und des Rasterbildes, ob es sich um eine einwandfreie Verstemmung handelt.
-

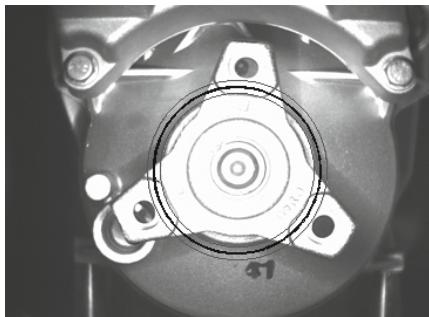
Der gesamte Flansch ist schwierig zu segmentieren, da er je nach Winkelstellung mit hellen Hintergrundbereichen verschmelzen kann. Es ist daher einfacher, Ausschnitte aus den Armen zu erzeugen, um diese zu zählen, und den Abstand zwischen Zentrum der Mutter und den Gewindebohrungen in den Flanscharmen als Maß für die Länge der Arme zu verwenden. Dabei kommt es hier nicht auf hohe Messgenauigkeit an, da nur Typenunterscheidung erforderlich ist.

Die Prüfung läuft zunächst fast genauso ab wie die der Verstemmung in Abschnitt 9.3.2, mit folgenden Unterschieden:

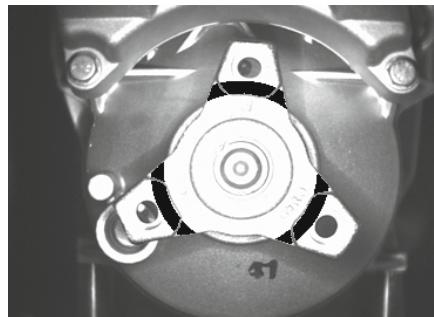
- Kleine Störobjekte, die aus den Kanten der Mutter entstehen, sind hier nicht von Bedeutung, daher kann man auf die aufwändige Dilatationsfilterung verzichten.
- Diese Prüfung ist nicht so positionsempfindlich, daher genügt die etwas weniger saubere Außenkontur des Schattenkreises als Referenz.

Der Endzustand entspricht ziemlich genau dem in Abschnitt 9.3.2 vor der Abbildung. Auch hier wird wieder ein Kreis definiert, jedoch liegt er jetzt, wie Abbildung 9.26 links zeigt, etwas außerhalb der Basis des Flansches und läuft so durch die Ansätze aller Flanscharme. Auf diesem Kreis kann man direkt eine binäre Segmentierung durchführen, wobei man die Konturverfolgung auf einen Aus-

schnitt der Arme begrenzt. Beim richtigen Flansch erhält man so ohne die zeitaufwändige Abwicklung drei Objekte, die in Abbildung 9.26 rechts gezeigt sind.



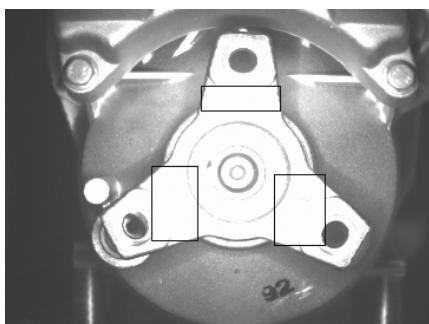
Positionierter Kreis



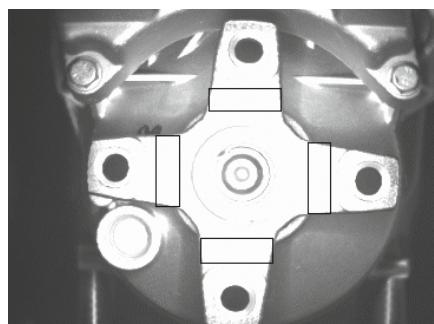
Segmentierte Ausschnitte

Abbildung 9.26: Flansch mit positioniertem Kreis und segmentierten Ausschnitten der Arme

Wie gewöhnlich werden für jedes dieser Objekte einige Merkmale berechnet, um sicherzustellen, dass es sich tatsächlich um Ausschnitte aus den Flanscharmen handelt, nicht etwa um irgendwelche Störobjekte. Die Merkmale dürfen hier nicht von der Winkelstellung des Flansches abhängen. Geeignet sind beispielsweise die Fläche und der maximale Radius, also der größte Abstand zwischen Objektschwerpunkt und Kontur. Nachdem alle Objekte ausgefiltert sind, die den entsprechenden Kriterien für die Ausschnitte aus den Flanscharmen nicht genügen, werden die verbliebenen Objekte gezählt. Wenn dabei nicht genau drei herauskommt, wird die Prüfung an dieser Stelle schon abgebrochen, da es sich nicht um den richtigen Flansch handeln kann. Abbildung 9.27 zeigt beide Fälle, einen korrekten dreiarmligen Flansch und einen Flansch, bei dem die Ausschnitte aus den Armen alle Kriterien erfüllen, nur dass vier vorhanden sind.



Darstellung: Ergebnis			Zoom:
Nr.	Soll Min	Soll Max	
✓ 3	3	3	



Darstellung: Ergebnis			Zoom:
Nr.	Soll Min	Soll Max	
✗ 4	3	3	

Abbildung 9.27: Drei- und vierarmliger Flansch

Wenn die richtige Anzahl Arme vorhanden ist, heißt das aber noch nicht, dass es sich auch um den richtigen Flanschtyp handeln muss. Dazu muss auch die Armlänge korrekt sein. Diese wird anhand des Abstandes zwischen dem Zentrum des Flansches und den Gewindebohrungen beurteilt, da die Gewindebohrungen sicherer zu segmentieren sind als die oberen Kanten der Arme, die je nach Winkelstellung mit hellen Hintergrundobjekten verschmelzen können. Die Gewindebohrungen sind so klar definierte Gebilde, dass sie sich leicht mit Template Matching finden lassen, ohne dass man besonderen Wert auf den Suchbereich legen müsste. Dieser kann einfach als Quadrat definiert werden, in das der komplette Flansch mit voller Armlänge hineinpasst. Es ist also nicht unbedingt erforderlich, hier auf einem mit entsprechendem Radius definierten Kreis um das Zentrum herum zu suchen. Dadurch würde zwar die abzusuchende Fläche eingeschränkt, der Rechenaufwand ist infolge der schwierigeren Adressierung abzusuchender Positionen aber deutlich höher, so dass kein nennenswerter Geschwindigkeitszuwachs zu verzeichnen ist.

Da es bei dieser Anwendung nicht um besonders genaue Vermessung geht, sondern um eine relativ grobe Typunterscheidung, kann man sich darauf beschränken, eine einzelne Gewindebohrung zu suchen und deren Abstand zum Zentrum zu messen. Aus demselben Grund ist keine Kalibrierung auf metrische Maße erforderlich. Der Abbildungsmaßstab ist ja für jeden Flansch gleich, so dass man direkt die Pixelmaße in Beziehung setzen kann. Abbildung 9.28 zeigt wieder zwei Fälle, den richtigen Flansch links, den ebenfalls dreiarmigen – sonst wäre man gar nicht so weit gekommen – mit zu langen Armen rechts.

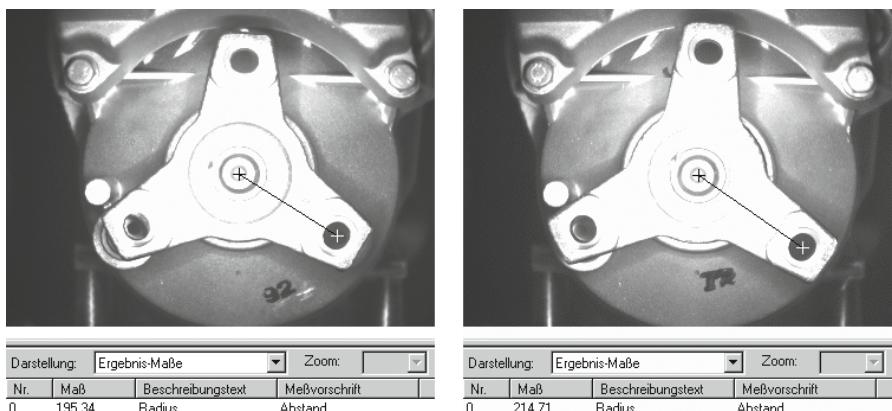


Abbildung 9.28: Flansche mit unterschiedlichen Armlängen

Abschließend sei noch angemerkt, dass man bei einer exakten Vermessung des Flanschdurchmessers etwas anders vorgehen würde. Man sucht dann alle drei Gewindebohrungen mit Template Matching, und erzeugt an den jeweils gefundenen Positionen die tatsächlichen Konturen der Gewindebohrungen durch binäre Segmentierung. Deren Schwerpunkte sind, wie wir aus Abschnitt 7.2.1 wissen,

subpixelgenau. Dann wird ein Ausgleichskreis entsprechend Abschnitt 7.5.1 berechnet – der in diesem Fall, bei drei Punkten, ein exakter Kreis ist, aber dennoch subpixelgenau, da er auf den Objektschwerpunkten beruht. Dessen Radius ist ein sehr genaues und stabiles Maß für den Radius des Lochkreises. Voraussetzung hierfür ist aber, dass die Bohrungen alle als einwandfreie Vollkreise sichtbar sind, also keine Störungen der Konturen auftreten. Tabelle 9.4 fasst den Prüfablauf zusammen.

9.4 Kontrastlose Anwesenheitskontrolle

In diesem Abschnitt geht es um die Prüfung des in Abbildung 9.29 gezeigten Schlauchhalters einer Klimaanlage. Zwischen die einzelnen Streben des Halters werden später Gummischläuche geklemmt. Es kann vorkommen, dass einzelne Halterstreben nicht ausgespritzt werden oder abbrechen. Die besondere Schwierigkeit liegt bei dieser Teilaufgabe darin, dass der Schlauchhalter aus demselben schwarzen Kunststoff besteht wie das gesamte Prüfteil. Es ist daher auch durch diffuse, geschaltete Beleuchtung nicht möglich, einen stabilen Kontrast zu erzeugen.



Abbildung 9.29: Schlauchhalter an Klimaanlage

Beispielbeschreibung 9.4. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 9.3.3 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Flansch
Beleuchtung	Flach einfallende Ringbeleuchtung, aus LED-Scheinwerfern zusammengesetzt, mit starker Überstrahlung.
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt ein Rechteck im ungefähren Zentrum des Flansches. 3. <i>Binärschwelle erzeugen</i>, optimiert die Schwelle anhand einer Histogrammanalyse des Arbeitsbereiches.

-
4. *Arbeitsbereiche binär erzeugen*, segmentiert den Schattenkreis im Zentrum des Gewindeflansches. Dieser kann durch Merkmale erzeugen und Arbeitsbereiche filtern noch auf verschiedene Eigenschaften überprüft werden.
 5. *Arbeitsbereiche Position bestimmen*, verwendet den Schwerpunkt des Schattenkreises als Positionsreferenz
 6. *Arbeitsbereiche definieren*, setzt einen Kreis, der durch die Ansätze der Flanscharme verläuft.
 7. *Arbeitsbereiche positionieren*, zentriert diesen Kreis auf den Schwerpunkt des Schattenkreises.
 8. *Arbeitsbereiche binär erzeugen*, schneidet entlang dieses Kreises Stücke aus den Flanscharmen als neue Objekte aus.
 9. *Merkmale erzeugen*, berechnet Kenndaten (Fläche und maximalen Radius) der Ausschnitte aus den Flanscharmen.
 10. *Arbeitsbereiche filtern*, sortiert alle Objekte mit unzulässigen Merkmalen aus, so dass nur tatsächliche Ausschnitte aus den Flanscharmen übrig bleiben.
 11. *Anzahl überprüfen*, stellt fest, ob die Anzahl der Flanscharme dem geforderten Flanschtyp entspricht.
 12. *Arbeitsbereiche definieren*, setzt einen Suchbereich, in den die Flanscharme vollständig hineinpassen.
 13. *Arbeitsbereiche positionieren*, richtet diesen Suchbereich wieder entsprechend dem Schattenkreis aus.
 14. *Template Matching*, sucht eine einzelne Gewindebohrung.
 15. *Maße erzeugen*, misst den Abstand zwischen den Schwerpunkten von Gewindebohrung und Flansch.
 16. *Maße Toleranzen überprüfen*, stellt fest, ob der gemessene Abstand den vorgeschriebenen Maßen des geforderten Flanschtyps entspricht.
-

Auch eine solche Aufgabe lässt sich jedoch lösen, indem man eine strukturierte Beleuchtung verwendet. Abbildung 9.30 zeigt die Projektion einer Laserlinie auf einen einwandfreien Schlauchhalter und einen Schlauchhalter mit einer fehlenden Strebe. Man sieht deutlich, dass im zweiten Fall einer der erforderlichen Schatten fehlt.

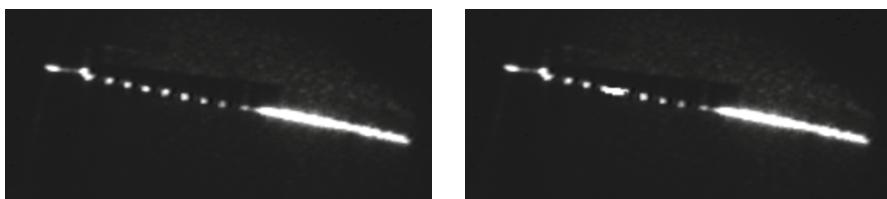


Abbildung 9.30: Einwandfreier und fehlerhafter Schlauchhalter unter Laserlinienbeleuchtung

Zur Überprüfung des Schlauchhalters erzeugt man zunächst durch binäre Segmentation alle hellen Objekte, also die Reflexe der Laserlinie auf dem dunklen Kunststoff. Dann sondert man alle Objekte aus, die nicht dem typischen Reflex des Lasers auf einer der Halterstreben entsprechen. Da diese Reflexe kleine, nahe-

zu kreisförmige Objekte sind, eignen sich hierfür beispielsweise die Merkmale Fläche, maximaler Radius und Kompaktheit, also das normierte Verhältnis von Fläche zu Umfang. Danach kann man die verbliebenen Objekte zählen.

Zusätzlich zu dieser direkten Zählung der Objekte kann man dann noch die Ortsfrequenz überprüfen, indem man die Schwerpunkt der verbliebenen Objekte durch eine Linie verbindet und auf dieser Linie erneut segmentiert, jetzt jedoch nach dunklen Objekten sucht. Bei Begrenzung dieser Segmentierung auf diese Verbindungslinie werden gerade die dunklen Zwischenräume zwischen den hellen Reflexen als Objekt erzeugt. Diese können dann noch einmal auf ihre Länge geprüft und gezählt werden. Beide Prüfungen zusammen bestimmen dann das Vorhandensein der richtigen Zahl von Objekten in vorgeschriebenen Intervallen, womit sich der Fall ausschließen lässt, dass ein weiter entfernt liegender zusätzlicher Reflex das Vorhandensein eines fehlenden Halterclips vortäuscht. Die in dieser Intervallprüfung erzeugten Objekte sind in Abbildung 9.31 hellgrau zwischen den Reflexen eingezeichnet.

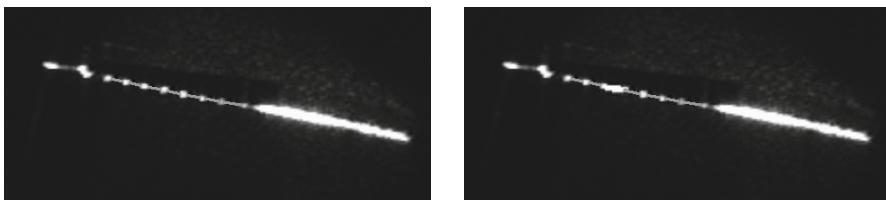


Abbildung 9.31: Zwischenraumobjekte zur Intervallprüfung am Schlauchhalter

Programmhinweis: Eine Verbindungslinie zwischen den Schwerpunkten von Objekten entsteht in **NeuroCheck** automatisch, wenn Sie für die gesamte Objektgruppe mit dem Prüfschritt *Ausgleichsgeometrien erzeugen* eine Gerade berechnen lassen. Der Prüfschritt erzeugt als Zwischenschritt ein virtuelles Objekt aus den Stützpunkten für die Berechnung der Ausgleichsgeraden. Dies sind genau die Schwerpunkt der beteiligten Objekte. Die fertige Prüfung finden Sie im Download im Verzeichnis \Beispiele\Anwesenheit\Klimaanlage.

Abschließend sei noch angemerkt, dass man mit einer solchen strukturierten Beleuchtung durch eine Laserlinie bei entsprechend hoher Auflösung auch echte Vermessungen vornehmen kann, wie in Abschnitt 8.8.2 gezeigt wird.

9.5 Anwesenheitskontrolle mit Zeilenkameras

Alle Beispiele der bisherigen Abschnitte verwendeten für die Prüfung jeweils ein Bild einer Flächenkamera, also ein rechteckiges Bild, das „am Stück“ zu einem bestimmten Zeitpunkt aufgenommen wurde. Diese Aufnahmemethode stößt bei bestimmten Arten von Prüfteilen an Grenzen. Das ist beispielsweise bei sehr großen Objekten der Fall, bei denen die festgelegte Sensorauflösung einer Flächen-

kamera in der Regel keine ausreichend detaillierte Aufnahme mehr zulässt. Ein anderes, sehr verbreitetes Beispiel sind zylindrische Teile, die in allen Industriezweigen in großer Vielfalt vorkommen. Wir wollen hier zunächst einige Bemerkungen dazu machen, wann die Prüfung von Drehteilen mit Flächenkameras möglich ist und welche Schwierigkeiten auftreten können.

9.5.1 Prüfung zylindrischer Teile mit Flächenkameras

In den bisherigen Beispielen haben wir bereits einige Sichtprüfungen an rotations-symmetrischen Teilen behandelt. Abschnitt 7.6 zeigte eine Vermessung der Außenkonturen eines Gewindeflansches, Abschnitt 9.2 eine Montageprüfung an einer Tropfkammer. Die Filmrollen aus 5.4 und die Zündkerze aus 1.6 sind weitere Beispiele für Prüfungen an zylindrischen Teilen.

In allen diesen Fällen bezogen sich die Prüfmerkmale entweder auf den Querschnitt der Prüfteile (beispielsweise bei den Gewindefermessungen) oder waren durch eine einzelne Bildaufnahme eines bestimmten Ausschnitts der Teileoberfläche zu ermitteln (beispielsweise bei der Schrifterkennung auf der Zündkerze oder der Blockcode-Erkennung auf den Filmrollen). Derartige Prüfaufgaben lassen sich auch bei runden Prüfteilen in einem „gewöhnlichen“ Bild einer Flächenkamera lösen. Wenn die Prüfaufgabe jedoch die Erfassung der gesamten Oberfläche eines runden Prüfteils erfordert, dann stößt man bei der Verwendung von Flächenkameras auf verschiedene Probleme.

Verkürzung der Projektion: Die Hauptschwierigkeit liegt darin, dass sich die Oberfläche des Teils vom Objekt fortkrümmt. Bei der Projektion der gekrümmten Oberfläche auf die Sensorebene kommt es außerhalb der Bildmitte daher zu mehr oder weniger starken Verkürzungen. Je breiter der im Bild erfasste Ausschnitt des Teils ist, desto stärker die Verkürzung. In dem Extremfall, dass das gesamte Teil im Bild erfasst wird, ist die Verkürzung an den äußersten Rändern natürlich unendlich groß, d. h. jede Bogenlänge auf der Oberfläche schrumpft im Bild auf eine Länge von null. Abbildung 9.32 zeigt den Effekt dieser Projektion unter der Annahme, dass die Kamera gerade auf die Achse eines Drehteils gerichtet sei. Auf dem Umfang des hier von oben betrachteten Teils sind Objekte im Abstand von 15° angeordnet. Man erkennt, dass bei der Projektion auf einen ebenen Sensor die Abstände der Objekte zueinander nach außen hin immer geringer werden. Die jeweils äußeren beiden Objekte verschmelzen sogar miteinander. Wenn es sich bei den Objekten beispielsweise um Striche eines Barcodes handelte, wäre dieser ganz offensichtlich so nicht mehr lesbar.

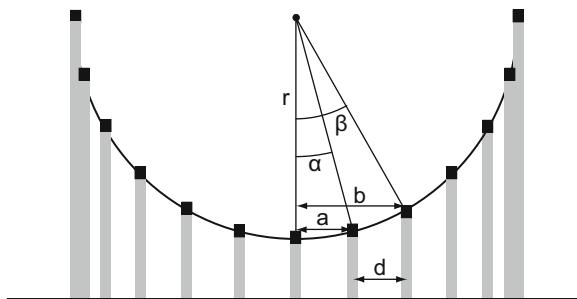


Abbildung 9.32: Verkürzung von Objekten entlang einer Kreislinie bei Projektion auf eine Sensorebene

Wenn sich zwei Objekte auf dem Umfang des zylindrischen Teils an den Winkelpositionen α und β befinden, dann ergibt sich gemäß Abbildung 9.32 ihr Abstand d in der Projektion auf die Sensorebene wie folgt:

$$d = b - a = r(\sin \beta - \sin \alpha). \quad (9.1)$$

Der (nicht dargestellte) Abstand d_k der Objekte auf dem Umfang des kreisförmigen Querschnitts ist gerade gleich der Bogenlänge zwischen den beiden Winkelstellungen:

$$d_k = 2\pi r \frac{\beta - \alpha}{360^\circ}, \quad (9.2)$$

wenn die Winkel in Grad angegeben werden. Tabelle 9.5 gibt eine Vorstellung vom Grad der Verkürzung, der mit wachsendem Abstand von der zentralen Blickrichtung auftritt. Zwischen 90° und 75° bleiben auf einem Werkstück von 100 mm Radius beispielsweise von einer Bogenlänge von 26,18 mm lediglich 3,41 mm in der Projektion übrig.

Tabelle 9.5. Verkürzung von Abständen bei der Projektion eines Kreisumfangs auf einen ebenen Sensor

Radius	Alpha	Beta	Bogen	Projektion	Restlänge
100	0°	15°	26,18	25,88	98,86 %
100	15°	30°	26,18	24,12	92,12 %
100	30°	45°	26,18	20,71	79,11 %
100	45°	60°	26,18	15,89	60,70 %
100	60°	75°	26,18	9,99	38,16 %
100	75°	90°	26,18	3,41	13,02 %

Aufgabentypen: Ein entscheidendes Kriterium dafür, ob eine Prüfaufgabe an einem zylindrischen Teil mit einer Flächenkamera gelöst werden kann, besteht

demnach darin, wie weit sich der tatsächlich zu prüfende Bereich über den Umfang des Teils erstreckt und ob es möglich ist, die relevanten Bereiche möglichst direkt in Blickrichtung der Kamera zu präsentieren. Bei der Schrifterkennung im Einführungsbeispiel in Abschnitt 1.6 erkennt man in Abbildung 1.9 an den Ziffern bereits eine perspektivische Verkürzung, die sich aber so dicht an der Mitte der Projektion noch in Grenzen hält.

Weiterhin ist von Bedeutung, ob das Prüfmerkmal axial oder radial orientiert ist. Der Barcode auf den Filmrollen in Abbildung 5.29 befindet sich zwar weit außerhalb der Mitte der Projektion, gescannt wird er aber senkrecht dazu, so dass die Verkürzung für das Lesen des Barcodes keine Rolle spielt.

Grundsätzlich ist eine Prüfung eines zylindrischen Objekts mit einer Flächenkamera-Aufnahme demnach also möglich, wenn eine der folgenden Bedingungen erfüllt ist:

- Die Prüfmerkmale beziehen sich nur auf Umriss oder Querschnitt des Objekts (so bei der Gewindevermessung in Abschnitt 7.6 und der Montagekontrolle in Abschnitt 9.2.).
- Die Prüfmerkmale liegen in der Projektion auf die Sensorebene so dicht an der Teileachse, dass die Verkürzung aufgrund der Krümmung sehr gering ist (so bei den Ziffern auf der Zündkerze in Abschnitt 1.6).
- Die Prüfmerkmale sind ausgeprägt axial orientiert, so dass die radiale Verkürzung keine wesentliche Rolle spielt (so bei dem Barcode auf Filmrollen in Abschnitt 5.4.).

Außerdem müssen Schwankungen der Orientierung des Prüfteils vor der Kamera (die zu unterschiedlichen optischen Verkürzungen der Prüfmerkmale führen) sich in einem Rahmen bewegen, der für die jeweilige Prüfaufgabe tolerierbar ist. Auch wenn die Zeichen entlang der Achse aufgebracht sind, würde eine Schrifterkennung sicher schwierig oder unmöglich, wenn das Prüfteil sich so weit dreht, dass die Zeichen nur noch am Rand sichtbar sind.

Vermessungsaufgaben: Besondere Probleme stellen sich bei Vermessungen auf der Oberfläche von zylindrischen Teilen, da die Verkürzung je nach Winkellage die geometrische Auflösung des Bildes und damit Kalibrierung und Genauigkeit verändert. Tabelle 9.5 zeigt, dass bereits in einem Bereich von $\pm 15^\circ$ beiderseits der direkten Blickrichtung der Kamera Verzerrungen von mehr als ein Prozent auftreten. Mit steigender Entfernung von der optischen Achse wächst diese Verzerrung durch die Nichtlinearität der Winkelfunktionen rapide an. Vermessungen sind an einem solchen Teil mit einer Flächenkamera-Aufnahme also streng genommen nur am Umriss des Teils oder entlang der Achse möglich, nicht entlang des Umfangs.

Prüfung über den vollen Umfang: Wenn die Prüfmerkmale sich über den vollen Umfang des Prüfteils erstrecken und die Prüfung mit Flächenkameras durchgeführt werden soll, dann müssen mehrere Aufnahmen gemacht werden. Die Anzahl der Aufnahmen richtet sich dabei neben der geforderten Auflösung auch nach der

zulässigen Verzerrung. Wenn beispielsweise eine Verkürzung um höchstens zehn Prozent akzeptabel ist, dann können gemäß Tabelle 9.5 kaum mehr als $\pm 30^\circ$ des Teileumfangs in einem Bild aufgenommen werden. Damit werden sechs Aufnahmen für jedes Teil erforderlich. Diese sechs Aufnahmen können auf verschiedenen Wegen erfasst werden:

- Es werden mehrere Kameras rund um das Teil angeordnet. Diese Variante ist relativ teuer (wegen der erforderlichen Anzahl von Kameras) und wirft auch oft Beleuchtungsprobleme auf, da die Beleuchtung für eine Kamera andere Kameras blendet kann.
- Das Teil wird vor einer Kamera getaktet gedreht, also in jeder der gewünschten Winkelstellungen vor der Kamera angehalten und aufgenommen. Diese Lösung erfordert einen hohen mechanischen Aufwand und eine komplexe Steuerung für den Schrittmotor und die Triggersignale. Da das Teil immer wieder angehalten und in Bewegung versetzt wird, entsteht ein hoher Zeitaufwand für die Bildaufnahme, was noch dadurch verstärkt wird, dass Vibrationen auftreten können, deren Beruhigung vor der Bildaufnahme abgewartet werden muss.
- Das Teil wird vor einer Kamera kontinuierlich gedreht. Dies ist in der Regel mechanisch einfacher als eine getaktete Drehung, erzeugt aber noch größere Synchronisationsprobleme, da die einzelnen Bildaufnahmen an genau spezifizierten Winkelstellungen ausgelöst werden müssen. Da das Teil dann in der Bewegung aufgenommen wird, kann nur sehr kurz belichtet werden, um Verwischungen zu vermeiden. Dies führt wiederum zu dem Problem, im Augenblick der Bildaufnahme hinreichend viel Licht auf die gekrümmte Oberfläche zu bringen.

Man sieht, dass eine Prüfung, die sich über den gesamten Umfang eines zylindrischen Teils erstreckt, mit Flächenkameras nicht befriedigend lösbar ist. Die Prüfung von Drehteilen ist daher eine typische Zeilenkamera-Anwendung.

9.5.2 Prüfung eines Ventilkörpers

Abbildung 9.33 zeigt einen Ausschnitt eines Ventilkörpers, der zur Steuerung des Ölflusses in der Kraftfahrzeughydraulik dient. Man kann an diesem Bild gut die Beleuchtungsproblematik erkennen, die solche Teile aufwerfen. Die gekrümmte, polierte Oberfläche reflektiert unter einem bestimmten Winkel sehr stark, so dass eine homogene Ausleuchtung über eine größere Fläche sehr schwierig ist.

Wir wollen hier zwei der Prüfkriterien für diese Bauteile näher betrachten:

- Die Bohrungen in den gedrehten Einstichen des Teils sind für den einwandfreien Ölfluss verantwortlich. Ihre Anzahl und Größe muss daher überprüft werden.
- Die erhabenen Laufflächen zwischen den gedrehten Einstichen müssen frei von Vertiefungen sein, da sie zur Abdichtung des Ölflusses dienen.

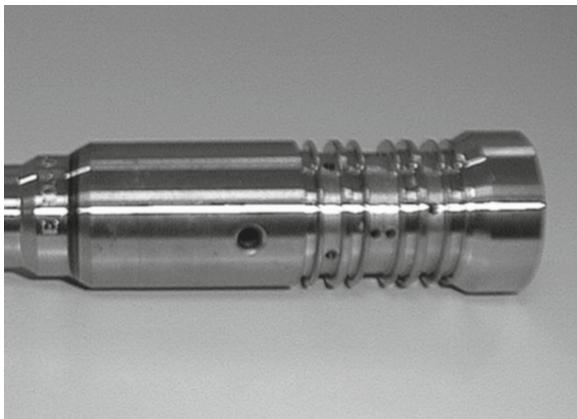


Abbildung 9.33: Ventilkörper aus der Kraftfahrzeughydraulik

Bildaufnahme: Die Prüfung wird mit einer Zeilenkamera vorgenommen. Wie in Abschnitt 8.4 ausführlich erläutert, erfordert eine Bildaufnahme mit einer Zeilenkamera immer eine Relativbewegung zwischen Prüfteil und Kamera. Abbildung 8.14 zeigt die Aufnahmesituation für die Erfassung der Oberfläche eines zylindrischen Bauteils mit einer Zeilenkamera. Die Bauteilloberfläche wird Zeile für Zeile mit der Kamera abgescannt und sozusagen in ein rechteckiges Bild abgewickelt.

Die Aufnahmemethode unterscheidet sich grundlegend von der Aufnahme eines rotierenden Bauteils mit einer Flächenkamera.

Rotation: Da die Zeilenkamera die Bauteilloberfläche kontinuierlich Zeile für Zeile abscannst, wird das Bauteil in jedem Fall kontinuierlich gedreht, nicht angehalten.

Synchronisation: Wenn die Drehung hinreichend gleichmäßig verläuft, benötigt das Bildverarbeitungssystem, bzw. die Kamera, lediglich ein Triggersignal, das den Anfang der Drehung signalisiert. Nur wenn sehr hohe Anforderungen an die Messgenauigkeit und Geometrie gestellt werden, ist eine Synchronisation von Drehbewegung und Bildaufnahme über Drehgeber entsprechend Abschnitt 8.4.4 erforderlich.

Verzerrung: Da die Kamera stets nur eine Zeile aufnimmt, die entlang der Bauteilachse verläuft, sieht sie immer direkt senkrecht auf die Oberfläche des Bauteils, so dass keine Krümmungsverzerrungen entstehen.

Man erhält also auf diese Weise eine verzerrungsfreie Abwicklung der Bauteilloberfläche. Der Zeitbedarf für eine solche Bildaufnahme ergibt sich aus der geforderten geometrischen Auflösung (siehe Abschnitt 8.4.2), der notwendigen Belichtungszeit (siehe Abschnitt 8.4.3) und aus den mechanischen Gegebenheiten, die bestimmen, wie schnell das Bauteil überhaupt gedreht werden kann. In der hier dargestellten Anwendung beanspruchte die Aufnahme eines Prüfteils etwa drei Sekunden.

Beleuchtung: Die beiden oben genannten Prüfaufgaben, Bohrungs- und Oberflächenkontrolle, erfordern unterschiedliche Beleuchtungen:

Bohrungskontrolle: Für die Überprüfung der Ölflussöffnungen wird eine diffuse Hellfeldbeleuchtung verwendet. Die polierte Oberfläche reflektiert das Licht sehr stark, so dass sich die Bohrungen deutlich dunkel abheben.

Oberflächenkontrolle: Störende Vertiefungen in den Laufflächen lassen sich mit einer gerichteten Dunkelfeldbeleuchtung detektieren. Das Prinzip der Beleuchtung entspricht Abbildung 5.26. Die gleichmäßig glatte Oberfläche reflektiert das gerichtet einfallende Licht an der Kamera vorbei, während Vertiefungen in der Oberfläche das Licht in Richtung Kamera streuen. Die Beleuchtung wurde mit einer Kaltlichtquelle mit Linienprojektor realisiert. Aufnahme- und Beleuchtungswinkel müssen dabei sehr präzise aufeinander abgestimmt werden.

Abbildung 9.34 zeigt Zeilenkamera-Aufnahmen des Prüfteils unter den beschriebenen Beleuchtungsbedingungen. Insgesamt erfordert die Prüfung also zwei Bildaufnahmen, zwischen denen die Beleuchtung umgeschaltet werden muss. Es ist dann vorteilhaft, nicht zuerst ein Bild aufzunehmen und vollständig zu verarbeiten, bevor das nächste Bild aufgenommen wird, sondern die Bildaufnahmen unmittelbar hintereinander zu Beginn der Prüfroutine auszuführen. Die Bilder können zwischengespeichert und dann verarbeitet werden, wenn das Bauteil bereits wieder aus der Prüfstation hinaustransportiert wird. So kann ein Teil der Handhabungszeit für die Prüfung verwendet werden.

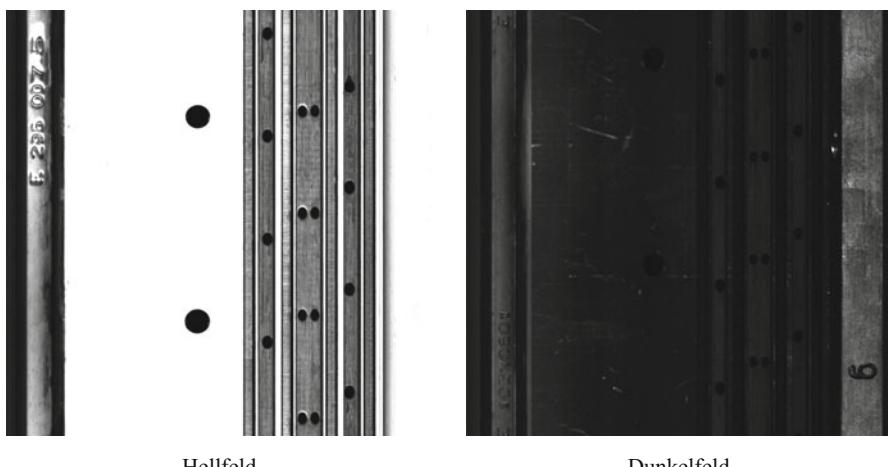


Abbildung 9.34: Zeilenkamera-Aufnahmen des Ventilkörpers unter verschiedenen Beleuchtungsbedingungen

Bohrungskontrolle. Aufgrund des guten Kontrastes in der Hellfeldaufnahme sind die Bohrungen leicht mithilfe einer Binärschwelle detektierbar. Allerdings ist dabei eine präzise Positionierung wichtig, um die Suche nach den Bohrungen auf

eng umschriebene Bereiche begrenzen zu können. Dadurch kann die Gefahr vermindert werden, außer den Bohrungen noch die Kanten der Einstiche zu finden, die im Bild ebenfalls dunkel erscheinen. Auch entstehen weniger Störobjekte, die herausgefiltert werden müssten, und je mehr solche Artefakte vorhanden sind, desto schwieriger kann es sein, gute Filterkriterien zu finden. Zudem ist eine kleinere Fläche auch schneller abzusuchen. Als Referenzobjekt für die Positionsbestimmung bietet sich in diesem Bild die charakteristische Dreieranordnung von Bohrungen an, die in Abbildung 9.35 gezeigt ist.

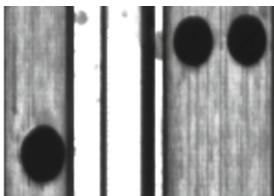


Abbildung 9.35: Referenzmuster für die Positionsbestimmung im Hellfeld

Die weitere Prüfung vollzieht sich dann in folgenden Schritten:

Suchbereiche: Aufgrund der Position der ersten gefundenen Bohrungsgruppe werden dann Suchbereiche für die Bohrungen gesetzt. Man beachte, dass es hierbei nur um die horizontale Position geht. Die vertikale Position ist ohnehin unbestimmt, da keine Information darüber vorliegt, bei welcher Winkelstellung des Bauteils die Zeilenkamera-Aufnahme begonnen hat. Es wird einfach ab der Stellung, in der das Teil in die Station hineingelangt, eine volle Umdrehung aufgenommen¹.

Segmentierung: Die Bohrungen werden innerhalb dieser Suchbereiche dann aufgrund einer Helligkeitsschwelle segmentiert. Eine Mustersuche, wie für die Positionierung, genügt hier nicht. Für die nachfolgende Überprüfung der Größe der einzelnen Bohrungen ist es erforderlich, die tatsächliche sichtbare Kontur zu bestimmen.

Beseitigung von Artefakten: Nach der Binärsegmentierung werden zunächst Störobjekte beseitigt, die sich hier unter anderem aus Oberflächenstrukturen und aus den Kanten der Einstiche ergeben können. Zu diesem Zweck werden alle Objekte beseitigt, die nicht hinreichend kreisähnlich sind (gemessen am Höhen-/Breitenverhältnis und der Kompaktheit; siehe hierzu Abschnitt 10.1.2) und deren Fläche nicht innerhalb bestimmter Grenzen liegt.

Überprüfung der Bohrungsgröße: Danach wird geprüft, ob in jedem der Einstiche die geforderte Anzahl von Objekten der passenden Größe vorhanden ist. Da das Bauteil mehr als 360° vor der Kamera gedreht wird, wird dabei ein Objekt mehr zugelassen, als Bohrungen vorhanden sein sollen.

¹ Tatsächlich werden etwas mehr als 360° abgewickelt, um zu vermeiden, dass eine Bohrung im Bild zufällig zerschnitten wird, weil die Drehung gerade dort anfängt. Es ist ein typisches Vorgehen bei Zeilenkamera-Anwendungen, etwas mehr als eine volle Umdrehung aufzunehmen.

Überprüfung der geometrischen Anordnung: Zuletzt wird der Abstand zwischen je zwei im gleichen Einstich benachbarten Bohrungen geprüft, um sicherzustellen, dass nicht möglicherweise eine Bohrung an falscher Stelle eingebracht wurde – was sich ja nicht in einer fehlerhaften Anzahl äußern würde.

Tabelle 9.6 fasst den Prüfablauf für die Bohrungskontrolle zusammen.

Beispielbeschreibung 9.6. Die folgende Tabelle fasst die Bohrungskontrolle aus Abschnitt 9.5.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Ventil
Beleuchtung	Diffuses Auflicht, Hellfeld
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt einen Suchbereich für die Detektierung der Referenzposition. 3. <i>Template Matching</i>, detektiert charakteristische Bohrungsgruppe als Positionsreferenz. 4. <i>Arbeitsbereiche Position bestimmen</i>, ermittelt die Abweichung von der Referenzposition. 5. <i>Arbeitsbereiche definieren</i>, setzt die Suchbereiche für die Segmentierung der Bohrungen. 6. <i>Arbeitsbereiche positionieren</i>, führt die Suchbereiche gemäß der ermittelten Positionsabweichung nach. 7. <i>Binärschwelle erzeugen</i>, optimiert die Hell-/Dunkelschwelle anhand einer Histogrammanalyse der Suchbereiche. 8. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert dunkle Objekte innerhalb der Suchbereiche. 9. <i>Arbeitsbereiche Merkmale erzeugen</i>, berechnet charakteristische Größen der gefundenen Objekte. 10. <i>Arbeitsbereiche filtern</i>, eliminiert Störobjekte. 11. <i>Arbeitsbereiche filtern</i>, prüft Übereinstimmung der verbliebenen Objekte mit den charakteristischen Merkmalen der Bohrlöcher. 12. <i>Arbeitsbereiche Anzahl überprüfen</i>, prüft, ob die korrekte Anzahl von Bohrungen vorhanden ist. 13. <i>Maße erzeugen</i>, berechnet Abstände benachbarter Bohrungen innerhalb eines Einstichs. 14. <i>Maße kalibrieren</i>, rechnet die Pixelmaße in metrische um. 15. <i>Maße Toleranzen überprüfen</i>, prüft die Abstände zwischen benachbarten Bohrungen.

Kontrolle der Laufflächen: Wir möchten ausdrücklich darauf hinweisen, dass das Beispiel in diesem Abschnitt stark vereinfacht ist. Gerade bei Anwendungen vom Typ der Oberflächenkontrolle sind in der Regel langwierige und aufwändige Versuchsreihen erforderlich, um die Vielfalt der möglichen Fehlertypen zu erfassen – und ebenso die Vielfalt der Erscheinungsbilder einwandfreier Oberflächen.

Nicht selten muss der Applikationsingenieur hier „tief in die Trickkiste“ greifen, um durch Vorverarbeitung, Ausblendung bekannter Oberflächeneffekte und so weiter eine prozesssichere Abgrenzung zwischen Gut- und Schlechteilen zu erreichen.

Abbildung 9.36 zeigt Ausschnitte der Oberflächen dreier verschiedener Ventilkörper. Das Prüfteil im linken Bild hat eine einwandfreie Lauffläche; das mittlere Prüfteil zeigt eine Reihe von kleineren Defekten, während beim rechten Prüfteil deutlich eine großflächige Beschädigung erkennbar ist. Eine Oberflächenprüfung dieses Typs, bei der nach bestimmten einzelnen Defekten gesucht wird, ist direkt mit einer Anwesenheitskontrolle vergleichbar – nur dass man auf Abwesenheit prüft, auf das Nichtvorhandensein der Defekte.

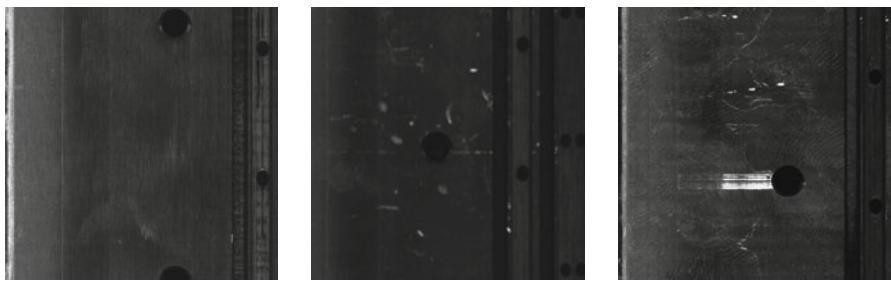


Abbildung 9.36: Verschiedene Ventilkörper unter Dunkelfeldbeleuchtung

Hier müssen zunächst die Defekte verstärkt werden, damit sie sicher detektiert werden können. Dazu wird das Bild dreimal gefiltert:

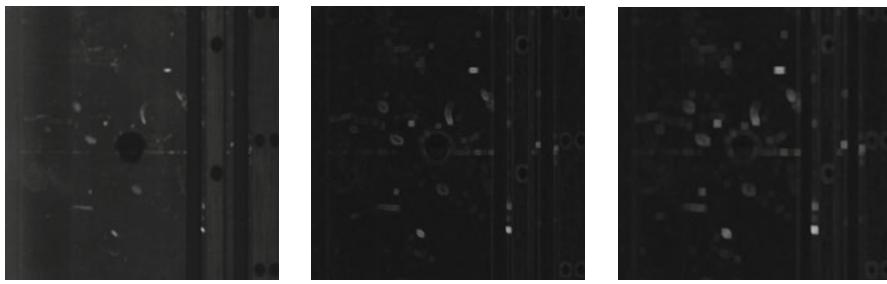
Mittelwertfilter: Dieser Tiefpassfilter dient dazu, feine Bearbeitungsstrukturen zu glätten, die sonst in der nachfolgenden Filterstufe störend hervortreten könnten.

Kontrastfilter: Mit diesem nichtlinearen Filter werden die Inhomogenitäten der Grauwertverteilung, durch die sich die Defekte äußern, stark hervorgehoben.

Dilatation: Zur Verbesserung der Detektierung werden die hell hervortretenden Fehlstellen verdickt.

Abbildung 9.37 zeigt das mittlere Bild aus Abbildung 9.36 nach den drei einzelnen Filterstufen.

Nach Abschluss der Vorverarbeitung wird im Bereich der Laufflächen nach den hellen Fehlstellen gesucht. Diese werden nach Fläche und Helligkeit bewertet. Zuletzt werden die verbliebenen hellen Objekte gezählt. Wenn keine Objekte übrig sind, die als Fehlstellen erkannt wurden, wird das gesamte Prüfteil als einwandfrei bewertet. Tabelle 9.7 fasst den Prüfablauf für die Kontrolle der Laufflächen zusammen.



Mittelwertfilter

Kontrastfilter

Dilatation

Abbildung 9.37: Ventilkörper mit beschädigter Lauffläche nach Vorverarbeitung

Beispielbeschreibung 9.7. Die folgende Tabelle fasst die Laufflächenkontrolle aus Abschnitt 9.5.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Anwesenheit\Ventil
Beleuchtung	Gerichtetes Auflicht, Dunkelfeld
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt einen Suchbereich über die gesamte Lauffläche. 3. <i>Bild filtern</i>, Mittelwertfilter zur Glättung von Bearbeitungsstrukturen. 4. <i>Bild filtern</i>, Kontrastfilter zur Verstärkung von Fehlstellen. 5. <i>Bild filtern</i>, Dilatation zur Verdickung heller Objekte. 6. <i>Binärschwelle erzeugen</i>, optimiert die Hell-/Dunkelschwelle anhand einer Histogrammanalyse der Suchbereiche. 7. <i>Arbeitsbereiche binär erzeugen</i>, segmentiert helle Objekte innerhalb des Suchbereiches. 8. <i>Arbeitsbereiche Merkmale erzeugen</i>, berechnet charakteristische Größen der gefundenen Objekte, in diesem Fall die maximale Helligkeit und die Fläche. 9. <i>Arbeitsbereiche filtern</i>, lässt Objekte übrig, die nach Größe und Helligkeit die Kriterien für eine Fehlstelle erfüllen. 10. <i>Arbeitsbereiche Anzahl überprüfen</i>, bewertet das Prüfteil als „schlecht“, wenn nach der Objektfilterung noch ein Objekt, also eine Fehlstelle, vorhanden ist.

9.5.3 Bemerkungen

Zum Abschluss wollen wir noch auf einige grundsätzliche Eigenschaften von Zeilenkamera-Anwendungen hinweisen.

Bildgröße: In der Regel hat man es bei Zeilenkamera-Anwendungen mit sehr großen Bildern zu tun. Im obigen Beispiel sind es Bilder von nur 1000×1000 Bildpunkten, wie man sie auch mit einer Flächenkamera aufnehmen könnte. Bildgrößen über 50 Millionen Bildpunkte sind jedoch keineswegs ungewöhnlich.

Zeitbedarf: Zeilenkamera-Prüfungen von Einzelbauteilen haben meist einen hohen Zeitbedarf. Dieser ergibt sich einmal aus der Verarbeitungszeit für die großen Zeilenkamerabilder, zum Anderen aber auch aus der Notwendigkeit, Kamera oder Prüfteil zu bewegen, um überhaupt zu einem zusammenhängenden Bild zu kommen. Dem Zeitbedarf kann man reduzieren, indem man während der Bildauswertung bereits das nächste Bild aufnimmt und damit diese beiden zeitintensiven Schritte parallel ausführt.

Oberflächenkontrolle: Dies ist ein sehr typisches Anwendungsgebiet für Zeilenkameras. Im vorliegenden Beispiel wurde die Oberfläche mittels einer Anwesenheitskontrolle nach entsprechender Vorverarbeitung geprüft. Das ist ein Prinzip, das sich häufig anwenden lässt, vor allem wenn es darauf ankommt, dass eine Oberfläche frei von bestimmten Effekten sein muss.

Wenn die Oberfläche dagegen eine bestimmte vorgeschriebene Struktur aufweisen soll, müssen Verfahren aus der Texturanalyse verwendet werden, die noch weitaus rechenintensiver sind als die hier eingesetzten Vorverarbeitungsfilter. Typisch für Anwendungen vom Typ der Oberflächenkontrolle sind langwierige Optimierungsprozesse an einer Vielzahl von Musterteilen – auch und gerade aus der Serienfertigung – da das Erscheinungsbild technischer Oberflächen und der darauf vorkommenden Defekte sehr vielfältig ist. Es ist daher von großem Vorteil, wenn das eingesetzte Bildverarbeitungssystem eine schnelle, interaktive Optimierung der laufenden Anwendung zulässt.

Beleuchtung: Wie Flächenkamera-Anwendungen sind auch Zeilenkamera-Prüfungen auf eine angepasste Beleuchtung angewiesen. Da die Kamera aber nur einen sehr kleinen Ausschnitt der Bildszene wahrnimmt, eine ganz bestimmte Zeile, reagieren Zeilenkamera-Anwendungen sehr empfindlich auf die Justierung von Beleuchtung und Kamera. Andererseits sind die zu beleuchtenden Flächen in der Regel klein, so dass eine homogene Ausleuchtung leichter zu erreichen ist.

Geometrie: Bei Zeilenkamera-Anwendungen, die auf der Abwicklung der Oberfläche eines zylindrischen Teils beruhen, ist die vertikale Position von Objekten im Bild in der Regel unbestimmt. Sie hängt davon ab, bei welcher Winkelstellung des Prüfteils mit der Bildaufnahme begonnen wurde. Man muss also entsprechende Positionscompensationen vorsehen.

Programmhinweis: In **NeuroCheck** kann der Prüfschritt *Zeilenkamerabild justieren* dazu verwendet werden, anhand einer festgestellten Position im Bild dafür zu sorgen, dass der Bildaufbau immer gleich ist, unabhängig von der Startposition der Drehung.

9.6 Zusammenfassung

In der folgenden Liste sind einige allgemeine Eigenschaften von Aufgaben aus der Anwesenheitskontrolle zusammengefasst. Da dieser Aufgabenbereich sehr heterogen ist, lassen sich die Prüfungen kaum nach einem vorgegebenen Ablauf aufbauen. Ebenso verschieden sind auch die Hardwareaufbauten (Beleuchtung, Anzahl der Kameras ...). Es erschien uns daher nicht sinnvoll, diese speziellen Aspekte der oben beschriebenen Prüfaufgaben in die Zusammenfassung aufzunehmen.

Aufgaben aus der Anwesenheitskontrolle: Es handelt sich hier um ein sehr heterogenes Aufgabengebiet, das vom bloßen Zählen von Objekten über die Prüfung einfacher Eigenschaften und die Erkennung von Objekten mit Klassifikatoren bis hin zur Montagekontrolle mit Vermessungsalgorithmen reicht.

Einfache Anwesenheitskontrolle: Bei diesem Teilgebiet ist die Gültigkeit der zu prüfenden Objekte mittels einfacher Merkmalsfilterungen feststellbar. Darunter verstehen wir die Überprüfung von Merkmalen für einzelne Objekte anhand zahlenmäßig einstellbarer Grenzwerte. Man geht daher folgendermaßen vor: Nach der Segmentierung berechnet man einfache Eigenschaften, wie Größe, Helligkeit, sortiert die ungültigen Objekte aus und zählt die verbleibenden. Wenn die Segmentierung mit Template Matching erfolgen kann, ist durch den Korrelationskoeffizient bereits ein erstes Maß für die Gültigkeit der gefundenen Objekte gegeben.

Montagekontrolle mit Vermessung: In diesem Teilgebiet folgt die Gültigkeit der Objekte aus Vermessungsergebnissen, das heißt es spielen auch Beziehungen zwischen Objekten, wie Abstände, Winkel, Kreisähnlichkeit und so weiter eine Rolle. In der Regel kommt es dabei nicht auf besonders hohe absolute Genauigkeit an, sondern um die Relation zu Referenzteilen.

Anwesenheitskontrolle mit Klassifikatoren: Wenn zur Feststellung der Korrektheit eines Objekts komplexe Wechselwirkungen zwischen Merkmalen geprüft werden müssen oder das Gesamterscheinungsbild des Objekts, bietet sich die Verwendung von Klassifikatoren an, die auf das Aussehen und die Eigenschaften gültiger und ungültiger Objekte trainiert werden.

Kontrastlose Anwesenheitskontrolle: Wenn die Form von Objekten zu prüfen ist, die aus einheitlichem Material gefertigt sind und daher kaum Kontrast ergeben, kann man strukturierte Beleuchtungen, wie etwa Laserlinien, einsetzen, um die relevanten Objekte zu segmentieren.

Relevante Eigenschaften: Das Hauptproblem bei vielen Aufgaben aus der Anwesenheitskontrolle ist die Übersetzung der Qualitätsforderung in berechenbare Werte. Beispiel: Die Forderung, dass ein Filter nicht schief eingesetzt werden darf, kann man am stabilsten durch die Messung symmetrischer Abstände überprüfen.

10. Querschnitt: Objektmerkmale

In den meisten Beispielen der vorangegangenen Kapitel haben wir die Gültigkeit von Objekten anhand verschiedener Merkmale überprüft. Es ist daher an der Zeit, eine Übersicht solcher Merkmale zu geben. Aus der Vielzahl in der Literatur beschriebener Merkmale haben wir eine Auswahl zusammengestellt, die sich in vielen praktischen Anwendungsfällen bewährt hat. Dabei wollen wir gleichzeitig noch einmal einen Blick auf einige der Schwierigkeiten werfen, die die Umsetzung alltäglich vertrauter Begriffe in die diskrete Welt digitaler Bilder aufwerfen kann.

10.1 Einfache geometrische Merkmale

Unter einfachen geometrischen Merkmalen fassen wir hier Werte wie Koordinaten, einfache Dimensionen, Fläche usw. zusammen. Die Abgrenzung zu formbeschreibenden Merkmalen, die wir hier vornehmen, ergibt sich hauptsächlich aus der algorithmischen Komplexität und ist natürlich teilweise willkürlich.

10.1.1 Umschreibendes Rechteck

Jedes segmentierte Objekt kann von einem Rechteck umschlossen werden. Im einfachsten Fall orientiert man dieses Rechteck entlang der Koordinatenachsen. Daraus ergeben sich die folgenden einfachen Merkmale:

Ursprungskoordinaten: Die Koordinaten der linken oberen Ecke des umschreibenden Rechtecks entsprechen der x-Koordinate des am weitesten links liegenden Objektpunkts und der y-Koordinate des am weitesten oben liegenden Objektpunkts.

Dimensionen: Die *Breite* oder x-Dimension des umschreibenden Rechtecks entspricht der Differenz der x-Koordinaten des am weitesten rechts und des am weitesten links liegenden Objektpunkts. Entsprechend ist die *Höhe* oder y-Dimension die Differenz der y-Koordinaten des am weitesten oben und des am weitesten unten liegenden Punkts.

Ratio: Unter der *Ratio* versteht man das Verhältnis von Höhe zu Breite des umschreibenden Rechtecks. Damit hat man bereits einen sehr einfachen formbeschreibenden Parameter.

So einfach diese Merkmale auch sein mögen, sie können in der Praxis dennoch wertvolle Dienste leisten. Wie wir bei den Anwendungen zur Schrifterkennung in Kapitel 5 gesehen haben, finden Segmentierungsalgorithmen Objekte nicht unbedingt in einer Reihenfolge, die dem Menschen sinnvoll erscheint. Wenn Zeichen zeilenweise gelesen werden sollen, müssen sie daher nach x- und y-Koordinaten sortiert werden. In aller Regel genügen hierfür die Ursprungskoordinaten des umschreibenden Rechtecks, die bedeutend schneller zu berechnen sind als die Schwerpunktskoordinaten, bzw. bei der Segmentierung gleich mit anfallen.

Man beachte übrigens, dass beim Template Matching das umschreibende Rechteck mit der Objektbegrenzung gleichbedeutend ist. Da ein mit Template Matching gefundenes Objekt nicht unbedingt eine geschlossene, visuell identifizierbare Kontur besitzen muss, können die Objektgrenzen nur anhand der Größe des Templates festgelegt werden.

Orientiertes umschreibendes Rechteck: Eine etwas aufwändigere Definition für das umschreibende Rechteck verwendet die Richtung der Hauptachse des Objekts als neue Koordinatenrichtung. In diesem neuen Koordinatensystem kann das umschreibende Rechteck dann genauso definiert werden wie oben.

Rechteckigkeit: (Russ 2007) und (Sonka 2008) definieren die *Rechteckigkeit* (*Rectangularity*) bzw. *Ausdehnung* (*Extent*) eines Objekts als das Verhältnis der Objktfläche A zur Fläche A_{kur} des kleinsten umschreibenden Rechtecks:

$$R = \frac{A}{A_{kur}}. \quad (10.1)$$

Dieses Merkmal ist eins für ein exaktes Rechteck, sonst kleiner. Eine genaue Berechnung des Merkmals setzt allerdings die Verwendung des orientierten umschreibenden Rechtecks voraus, da sich sonst für ein exaktes, aber gedrehtes Rechteck ein zu kleiner Wert ergibt.

10.1.2 Fläche und Umfang

Da ein durch Template Matching gefundenes Objekt keiner visuell geschlossenen Bildstruktur entsprechen muss, sind Fläche, Umfang und ähnliche Werte im Allgemeinen nur für Objekte sinnvoll, die anhand von Binärschwellen oder anderen Helligkeitseigenschaften segmentiert wurden. Wir können daher im Folgenden davon ausgehen, dass das Objekt in einer Konturdarstellung vorliegt.

Fläche: Die *Fläche* ist eine der am einfachsten zu ermittelnden Eigenschaften eines segmentierten Objekts. Sie entspricht der Anzahl der Pixel innerhalb der Objektkontur. Bei binär segmentierten Objekten muss man dabei allerdings zwei Definitionen unterscheiden:

Geschlossene Fläche: Die geschlossene Fläche umfasst alle Punkte, die sich innerhalb der Objektkontur befinden, auch wenn diese gemäß der Binärschwelle nicht zum Objekt gehören würden.

Fläche ohne Löcher: In diesem Fall werden alle Bereiche, die nach der Binärschwelle nicht zum Objekt gehören, aus der Berechnung ausgenommen.

Zur Berechnung der Fläche bedient man sich häufig einer Lauflängendarstellung (englisch *Run Length Encoding*) des Objekts. Die Punkte in jeder Zeile des Objekts werden dabei in einer Folge von Segmenten, so genannten *Runs*, zusammengefasst. Jedes Segment gehört vollständig zum Objekt, enthält also keine objektfremden Punkte, und wird durch einen Startpunkt und eine Länge beschrieben. Die Fläche des Objekts ergibt sich dann einfach aus der Summe der Segmente. Abbildung 10.1 zeigt die Lauflängendarstellung eines Objekts, das sowohl eine Einbuchtung als auch ein „Loch“ besitzt. Nach jeder Zeile folgen die entsprechenden Segmente, beschrieben durch die Startpunktkoordinaten in Klammern und die Länge des Segmentes.

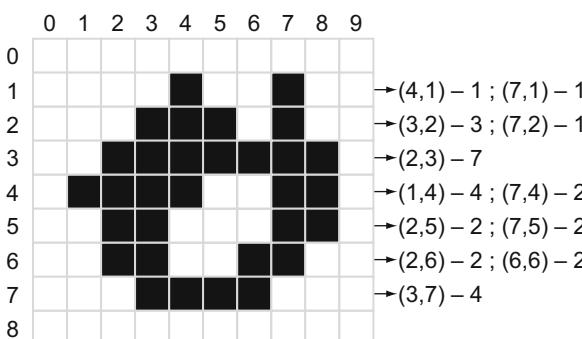
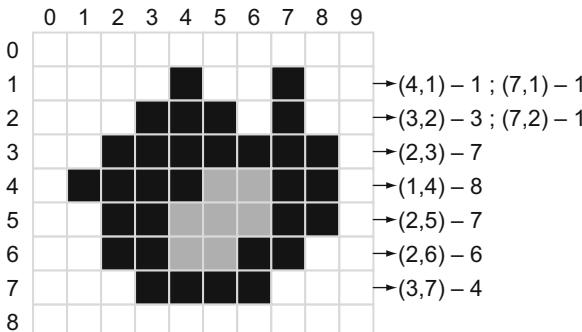


Abbildung 10.1: Lauflängencodes zur Flächenberechnung

Programmhinweis: Der NeuroCheck-Prüfschritt *Arbeitsbereiche binär erzeugen* besitzt die Option „Regionen füllen“. Wenn diese Option abgeschaltet wird, werden Objekte ohne Löcher erzeugt, andernfalls wird die Darstellung für die Berechnung der geschlossenen Fläche benutzt.

Umfang: Der *Umfang* eines segmentierten Objekts lässt sich am einfachsten direkt aus dem Kettencode berechnen. Dabei muss man allerdings eine gewisse Vorsicht walten lassen. Zur Berechnung des Umfangs einer Kontur kann man nicht einfach die Pixel zählen, aus denen die Kontur besteht, denn:

1. Die wahre Kontur verläuft weder genau an der Außenkante der Pixel noch an der Innenkante, sondern an einem aufgrund der Kombination aus Ortsdiskretisierung und Graustufenquantisierung nicht eindeutig bestimmten Ort dazwischen, den man üblicherweise in der Mitte eines Pixels annimmt.
2. Die Entfernung zwischen den Mittelpunkten zweier Pixel ist abhängig vom Verlauf der Kontur an dieser Stelle, wie Abbildung 10.2 zeigt:

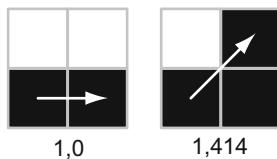


Abbildung 10.2: Umfangsbeitrag von Konturstücken

- Eine vertikale oder horizontale Verbindung zwischen zwei Punkten trägt mit einer Länge von 1,0 zum Umfang bei.
- Diagonale Verbindungen tragen mit einer Länge von $\sqrt{2} \approx 1,414$ zum Umfang bei.

Mit dieser Methode ist die Berechnung des Umfangs aus dem Kettencode sehr einfach. Gemäß der Definition des Kettencodes in Abschnitt 4.3.3 werden alle diagonalen Konturstücke durch ungerade Werte des Kettencodes repräsentiert, alle vertikalen und horizontalen Konturstücke durch gerade Werte. Der Umfang des in Abbildung 10.3 angenäherten Kreises ergibt sich damit zu:

$$U = 8 \times 1,0 + 8 \times 1,414 = 19,312 .$$

Eine echte Kreislinie, die in der Mitte der Konturpunkte verläuft, hätte einen Radius von drei Bildpunkten, folglich einen Umfang von

$$U = 6 \times \pi = 18,85$$

Bildpunkten. Die Abweichung beträgt also trotz der geringen Anzahl von Bildpunkten nur etwa 2,5 %. Dennoch ist bei der Verwendung von Umfangswerten Vorsicht geboten. Der Umfang der meisten realen Objekte ist stark von der Bild-

auflösung abhängig. Wenn die Vergrößerung anwächst, sind mehr Unregelmäßigkeiten in der Kontur zu erkennen, wodurch sich größere Umfangswerte ergeben¹. Des weiteren sind Umfangswerte sehr rauschempfindlich. Abbildung 10.4 zeigt die Ergebnisse von Umfangs- und Flächenberechnungen an Kreisen mit und ohne Rauschen. Durch das Rauschen werden die Konturlinien unregelmäßiger, wodurch der Umfang um bis zu 1/3 ansteigt, während die Fläche mit Abweichungen von unter drei Prozent nahezu unberührt bleibt. In industriellen Bildszenen kommen allerdings glatte Kanten sehr oft vor, so dass der Umfang dort durchaus ein brauchbares Merkmal ist.

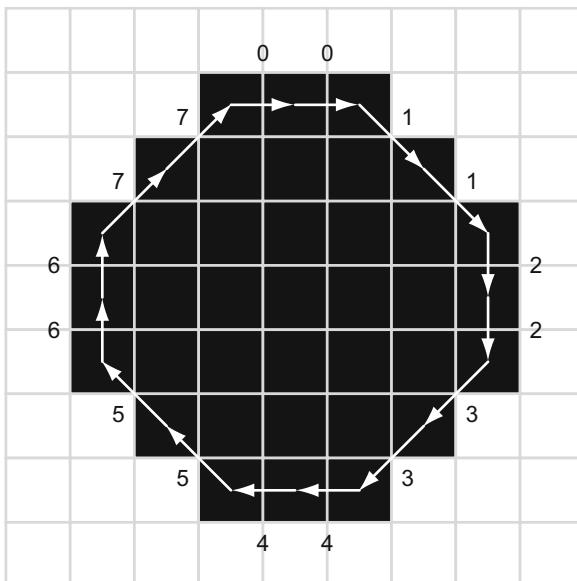
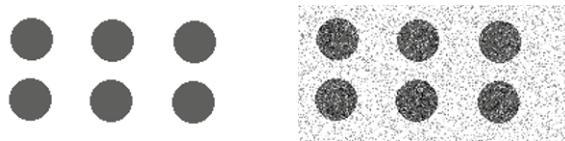


Abbildung 10.3: Umfangsberechnung an diskretisiertem Kreis

Kompaktheit: Aus Fläche und Umfang lässt sich noch ein sehr interessantes Merkmal berechnen. Wegen des Zusammenhangs mit Umfang und Fläche wollen wir es hier vorstellen, man könnte es allerdings auch in die Gruppe der formbeschreibenden Merkmale einordnen, da es für gleiche geometrische Formen immer denselben Wert ergibt – wie üblich abgesehen von den Diskretisierungseffekten digitaler Bilder.

¹ Hierin liegt die Wurzel des Küstenlängenparadoxons nach (Mandelbrot 1987): Eine Küstenlinie wird umso länger, je kleiner die verwendete Meßplatte ist. Das Verhalten von Objekteigenschaften unter verschiedenen Vergrößerungen ist außerdem die Wurzel eines eigenen Zweigs der Bildanalyse, der mit fraktalen Dimensionen arbeitet.



Nr.	Umfang	Fläche	Nr.	Umfang	Fläche
0	98.704	749.000	0	120.503	730.000
1	98.704	749.000	1	132.125	735.000
2	98.704	749.000	2	111.018	742.000
3	98.704	749.000	3	119.296	737.000
4	98.704	749.000	4	110.018	735.000
5	98.704	749.000	5	126.610	732.000

Originale Kreise

Kreise unter Rauschen

Abbildung 10.4: Einfluss von Rauschen auf Flächen- und Umfangsberechnungen

(Nischwitz et al. 2007) definiert die *Kompaktheit* als Verhältnis von Umfangsquadrat zu Fläche. Dieses Maß nimmt für einen Kreis den minimal möglichen Wert an, da ein Kreis bei gegebenen Umfang die maximale Fläche aller geometrischen Formen einschließt. Dem Begriff Kompaktheit entspricht daher eigentlich die umgekehrte Definition besser. Da der Kreis die kompakteste aller Formen ist, sollte er auch den höchsten Wert erhalten. Wir definieren daher die Kompaktheit analog zum *Formfactor* nach (Russ 2007) als Verhältnis von Fläche zu Quadrat des Umfangs:

$$K = \frac{4\pi A}{U^2}. \quad (10.2)$$

Die Skalierung mit 4π sorgt dafür, dass beim Kreis (mit Fläche πr^2 und Umfang $2\pi r$) dieser Wert genau eins wird. Für jede andere geometrische Form ergibt sich dann ein kleinerer Wert.

10.1.3 Schwerpunktskoordinaten

Der *Schwerpunkt* eines Objekts ist definiert als der Punkt, in dem man das Objekt unterstützen müsste, um es im Gleichgewicht zu halten. Die Koordinaten (x_s, y_s) des Schwerpunkts lassen sich durch Mittelung der Koordinaten aller Punkte eines Objekts über seine Fläche A berechnen:

$$x_s = \frac{\sum_i x_i}{A}, \quad (10.3)$$

$$y_s = \frac{\sum_i y_i}{A}, \quad (10.4)$$

Man kann den Schwerpunkt auch direkt aus der Kontur berechnen. Der Vorteil ist dabei, dass weniger Punkte verwendet werden müssen. Andererseits haben stark zerklüftete Bereiche der Kontur dann einen zu großen Einfluss, da sie überproportional viele Punkte enthalten. Nach (Russ 2007) lässt sich dieses Problem durch folgende Berechnungsmethode korrigieren:

$$x_s = \frac{\sum_i (x_i + x_{i-1})^2 \cdot (y_i - y_{i-1})}{A}, \quad (10.5)$$

$$y_s = \frac{\sum_i (y_i + y_{i-1})^2 \cdot (x_i - x_{i-1})}{A}, \quad (10.6)$$

$$A = \frac{\sum_i (x_i + x_{i-1}) \cdot (y_i - y_{i-1})}{2}. \quad (10.7)$$

Den Vorteil der geringeren Punkteanzahl erkauft man also durch erhöhten Rechenaufwand für jeden Punkt.

In Abschnitt 7.2.1 wurde bereits darauf hingewiesen, dass Schwerpunktkoordinaten infolge der Mittelung über das gesamte Objekt subpixelgenau sind. Tatsächlich gehören sie zu den stabilsten Maßen, die man für ein Objekt berechnen kann.

10.1.4 Achsen und Radien

Hauptträgheitsachse: Auf Definition und Berechnung der Hauptachse eines Objekts sind wir schon in Abschnitt 3.2.1 eingegangen, daher werden wir hier nur kurz die wichtigsten Aspekte wiederholen.

Die Hauptträgheitsachse eines Objekts ist definiert als diejenige Achse, um die das Objekt mit dem geringsten Trägheitsmoment rotiert. Dies ist gleichbedeutend mit der Forderung, dass die Summe der Abstandsquadrate aller Punkte des Objekts von dieser Achse minimal sein muss. Die formale Berechnung nach Gleichung (3.1) bringt einen sehr hohen Rechenaufwand mit sich. Daher wurden in der Bildverarbeitung verschiedene Näherungsverfahren zur Bestimmung der Achse entwickelt.

(Parker 1994) gibt ein Näherungsverfahren an, das auf der Annahme beruht, die Achse gehe genau durch einen Bildpunkt auf der Objektkontur. Da die Achse au-

Berdem notwendigerweise durch den Schwerpunkt verlaufen muss, kann man folgendermaßen vorgehen:

1. Man markiert alle Bildpunkte auf der Objektkontur, die oberhalb des Schwerpunkts liegen.
2. Für jede Linie durch den Schwerpunkt und einen dieser Bildpunkte berechnet man den mittleren Abstand aller Punkte des Objekts zu dieser Linie.
3. Die Linie mit dem geringsten mittleren Punktabstand kann als Näherung der Hauptachse verwendet werden.

Umschreibendes Polygon: Wenn man die Koordinatenachsen schrittweise dreht, erhält man für jede neue Koordinatenrichtung ein neues umschreibendes Rechteck. Die Punkte, an denen diese umschreibenden Rechtecke die Kontur berühren, können nach (Russ 2007) als Eckpunkte eines umschreibenden Polygons verwendet werden. Bei einem Winkelinkrement von zehn Grad erhält man neun unterschiedliche Richtungen der Achsen, folglich 36 Eckpunkte für das Polygon. Die Achse dieses Polygons kann wieder als Näherung für die Hauptachse des Objekts verwendet werden.

Maximaler Feret-Durchmesser: Unter einem *Feret-Durchmesser* versteht man nach (Russ 2007) die Länge der Projektion eines Objekts auf eine Koordinatenachse. Die Höhe und Breite des umschreibenden Rechtecks sind also die Feret-Durchmesser des Objekts im Standardkoordinatensystem. Der maximale Feret-Durchmesser ist entsprechend der maximale Abstand zweier Punkte der Objektkontur. Die Richtung dieser längsten Sehne durch das Objekt ist zwar keine Näherung der Hauptachse, aber ebenfalls ein Parameter der Orientierung des Objekts, da diese Richtung in einer festen Beziehung zur Hauptachse steht – wenn man von Diskretisierungseffekten durch die endliche Bildauflösung absieht. Außerdem kann die Länge der Sehne selbst als einfaches Maß für die Objektform benutzt werden.

Radien: Unter einem *Radius* verstehen wir hier den Abstand vom Objektschwerpunkt zu irgendeinem Konturpunkt. Durch Abscannen der gesamten Kontur kann man einen minimalen und einen maximalen Radius und deren Richtungen finden. Darüber hinaus kann man dabei auch den mittleren Radius und andere statistische Werte der Radienverteilung berechnen. Alle diese Werte können als Merkmale für die Objektform und die Orientierung benutzt werden. Für kreisförmige Objekte erhält man auch einen maximalen und einen minimalen Radius. Diese kommen jedoch nur aufgrund der Bilddiskretisierung zustande und haben daher keine systematische Beziehung zueinander. Klar definierte Objektformen mit bekannten Vorzugsrichtungen können jedoch auf diese Weise durchaus voneinander unterschieden werden.

Wenn man jeden einzelnen Radiuswert in einem Vektor speichert, erhält man übrigens wieder genau die Polarabstandsprojektion, die wir in Abschnitt 3.2.2 für die Positionsbestimmung eingeführt haben. Sie diente dort als sehr stabiles Maß für die Orientierung von Objekten, kann aber auch zu deren Unterscheidung mittels Korrelation verwendet werden. Hiermit befinden wir uns schon an der Grenze

zu den formbeschreibenden Merkmalen, die im folgenden Abschnitt behandelt werden.

10.2 Formbeschreibende Merkmale

10.2.1 Konturkrümmung

Eine Kontur kann als ebene Kurve betrachtet werden. Die Krümmung einer ebenen Kurve ist definiert durch die Richtungsänderung der Tangente an die Kurve. Nach (Bronstein et al. 2005) gilt formal mit den Tangentenrichtungen α und α_0 in den Punkten P und P_0 und der Bogenlänge $\overline{PP_0}$ zwischen diesen Punkten für die Krümmung:

$$K = \lim_{P_0 \rightarrow P} \frac{\alpha - \alpha_0}{\overline{PP_0}}. \quad (10.8)$$

Diese Definition veranschaulicht Abbildung 10.5.

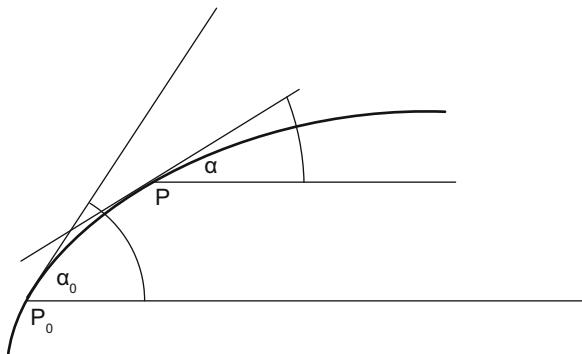


Abbildung 10.5: Formale Definition der Krümmung

Man kann zeigen, dass für eine in der Form $y = f(x)$ gegebene Kurve die Krümmung dann die Form

$$K = \frac{y''(x)}{(1 + y'^2)^{3/2}} \quad (10.9)$$

nimmt. Man benötigt also für eine formale Berechnung der Krümmung die erste und zweite Ableitung der Kontur. Digitale Konturen zeigen aufgrund von Diskrethisierungseffekten normalerweise einen sehr „unruhigen“ Verlauf. Da Differenzie-

rungen zur Verstärkung solcher Variationen führen, ist es meist sehr schwierig, auf diese Weise zuverlässige Maße für die Krümmung einer Kontur zu erhalten.

Wenn man nun die Tangente im Punkt P so verschiebt, dass sie Punkt P_0 schneidet, so ist unmittelbar klar, dass der durch den Punkt P verlaufende lotrechte Abstand zwischen dieser Linie und der Tangente an den Punkt P_0 ein Maß für die Winkeldifferenz $\alpha - \alpha_0$ darstellt, wie Abbildung 10.6 zeigt. Je stärker die Kurve gekrümmmt ist, desto weiter öffnet sich dieser Winkel und desto größer wird folglich dieser Abstand.

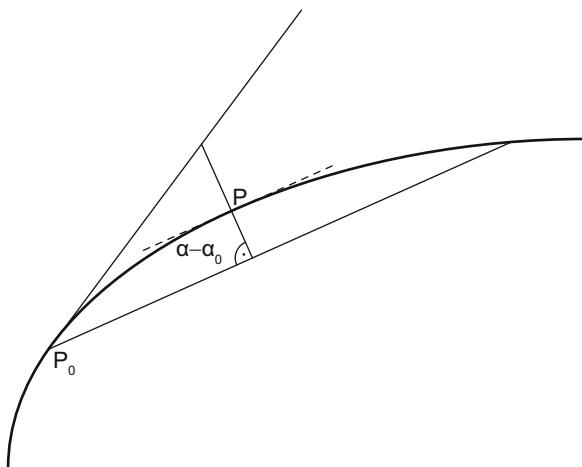


Abbildung 10.6: Annäherung der Krümmung

Wegen der Schwierigkeiten bei der numerischen Ableitung digitaler Kurven berechnet man zur Ermittlung der Krümmung normalerweise weder die Tangenten und noch den eingezeichneten Abstand. Stattdessen definiert man eine bestimmte Länge der Sehne, die in die Kurve gelegt wird, und benutzt den Abstand zwischen dieser Sehne und der Kurve als Maß für die Krümmung. Abbildung 10.7 zeigt dies für konvexe und konkave Kurvenstücke. Formal gibt es dabei überhaupt keinen Unterschied, nur dass die Kontur einmal links von der Verbindung zwischen P_0 und P_1 liegt, einmal rechts davon. Diese beiden Fälle lassen sich durch das Vorzeichen unterscheiden, wobei die Definition willkürlich ist. Man kann beispielsweise definieren, dass konkave Bereiche als negativ, konvexe als positiv betrachtet werden.

Es ist klar, dass der absolute Wert dieses Abstandes von der Länge der Sehne abhängt. Wegen des Grenzwertes in Gleichung (10.8) entspricht dieses Maß der wahren Krümmung umso genauer, je kürzer die Sehne wird. Andererseits schwankt es dann auch wegen der unvermeidlichen Diskretisierungseffekte einer digitalen Kontur wieder stärker. In der Praxis wird es selten darum gehen, den Krümmungswert wirklich absolut zu messen. Vielmehr wird man die Krümmung eines Prüfteils mit Referenzwerten vergleichen, die unter den gleichen Voraussetzungen berechnet wurden.

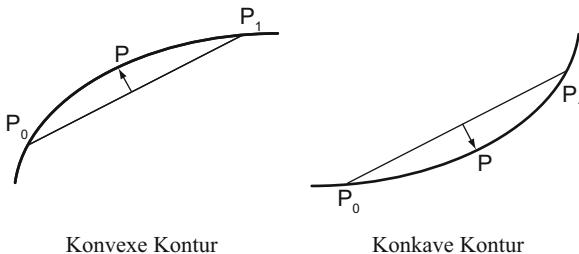


Abbildung 10.7: Krümmung in konvexen und konkaven Konturen (Sehne wird von links nach rechts durchlaufen)

Eine typische Anwendung der Krümmungsberechnung zeigt Abbildung 10.8. Es handelt sich dabei um eine Draufsicht auf handelsübliche Rollenlager. Der Abschlussring der inneren Lagerschale eines der beiden Lager ist im linken Bereich durch einen Bearbeitungsfehler an der Oberfläche beschädigt. Man kann den hell reflektierenden Teil des Rings leicht als Objekt segmentieren. Der Durchmesser des Rings beträgt etwa 440 Bildpunkte. Eine Krümmungsberechnung mit einer Sehnenlänge von 40 Bildpunkten ergibt bei dem einwandfreien Lager eine mittlere Krümmung von 0,571 bei einer Standardabweichung von 0,547. Das beschädigte Lager zeigt demgegenüber eine Standardabweichung von 1,206. Noch wesentlich krasser ist der Unterschied bei den Maximalwerten. Während beim einwandfreien Lager die Diskretisierungsfehler der kreisförmigen Kontur zu einer minimalen Krümmung von -1,27 und einer maximalen von 1,85 führen, liegen diese Werte für das beschädigte Teil bei -7,7 und 7,13 (negative Werte zeigen konkave Bereiche an).

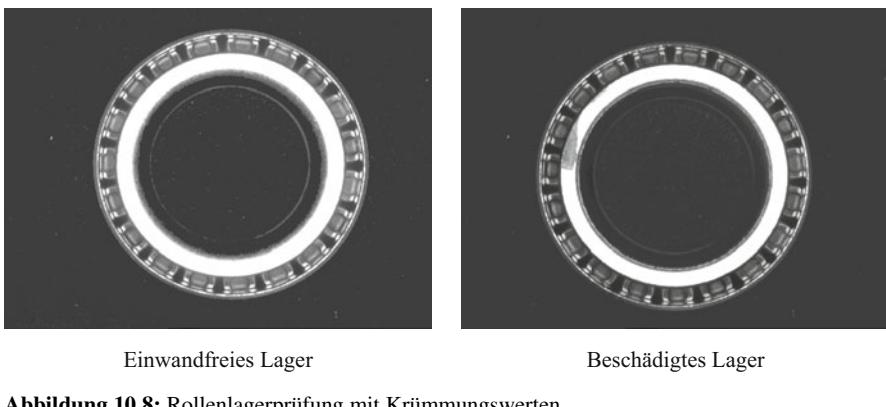


Abbildung 10.8: Rollenlagerprüfung mit Krümmungswerten

Krümmungsvektor: Analog zur Polarabstandsprojektion, die im Abschnitt 3.2.2 erklärt wird, kann man auch die Krümmungswerte entlang der gesamten Kontur in einem Vektor zusammenfassen. Dieser hat für bestimmte Objekte ebenfalls eine charakteristische Form und kann daher zur Objekterkennung benutzt werden.

10.2.2 Fasermerkmale

Unter der *Faserlänge* eines Objekts wollen wir hier die Länge einer Linie verstehen, die überall den maximal möglichen Abstand von der Objektkontur aufweist. Bei einem Objekt wie in Abbildung 10.9 ist dies anschaulich die Länge der Mittellinie des Objekts. Die *Faserbreite* ist dann entsprechend der Abstand gegenüberliegender Konturpunkte senkrecht zu dieser Mittellinie.

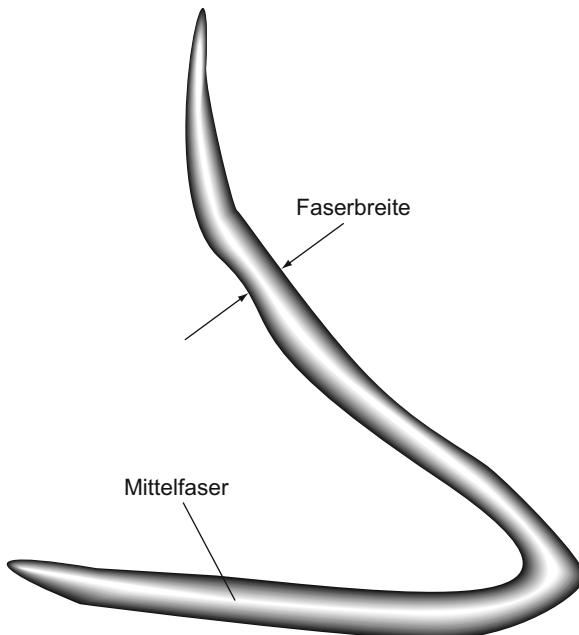


Abbildung 10.9: Faserlänge und Faserbreite eines Objekts

Die Lage der Mittelfaser kann nach (Russ 2007) mithilfe einer Karte des euklidischen Abstandes von der Kontur bestimmt werden. Alle Punkte innerhalb eines Objekts werden entsprechend ihrem Abstand von der Kontur eingefärbt. Konturpunkte selbst sind schwarz, die am weitesten entfernten Punkte weiß. Dabei entsteht eine Darstellung wie in Abbildung 10.9. Die Berechnung des euklidischen Abstands von der Kontur für jeden Punkt des Objekts ist natürlich ziemlich aufwändig, man kann jedoch ein iteratives Verfahren verwenden:

1. Jeder Punkt außerhalb eines Objekts erhält in der Distanzkarte die Helligkeit null, jeder Punkt einer Objektkontur die Helligkeit eins.
2. Jeder Punkt innerhalb eines Objekts, der einen Punkt der Helligkeit g berührt, erhält die Helligkeit $g + 1$. Dies wird fortgesetzt, bis jedem Punkt ein Helligkeitswert zugewiesen wurde.

Auf diese Weise entsteht eine Darstellung analog zu Abbildung 10.9, bei der die Mittelfaser des Objekts hell hervortritt. Die Länge einer solchen Linie kann nach (Russ 2007) mit einem Fehler von höchstens 2,5 % nach folgender Formel bestimmt werden:

$$L = 0,948 \times \text{Anzahl orthogonaler Nachbarn} + 1,340 \times \text{Anzahl diagonaler Nachbarn}. \quad (10.10)$$

Für Linien ist dieses Verfahren genauer als die bei der Umfangsberechnung für geschlossene Konturen in Abschnitt 10.1.2 benutzte Methode.

10.2.3 Euler-Zahl

Die Euler-Zahl zur Beschreibung von Objekten beruht auf der Anzahl von „Löchern“, die ein Objekt enthält. Bei einem binär segmentierten schwarzen Objekt wäre das also die Anzahl heller Objekte, die sich innerhalb seiner Konturen befinden. Trotz dieser einfach anmutenden Definition ist dieses Merkmal in technischen Anwendungen sehr leistungsfähig. Viele Werkstücke lassen sich anhand der in ihnen enthaltenen Objekte unterscheiden, beispielsweise verschiedene Typen von Befestigungsblechen, Fahrzeugfelgen oder Rohrflanschen.

10.2.4 Momente und Fourierdeskriptoren

Momente: Mithilfe von *Momenten* kann man Objekte rotations-, translations- und größeren invariant beschreiben. Die Momente der Ordnung $p + q$ sind nach (Nischwitz et al. 2007) definiert als:

$$m_{pq} = \sum_x \sum_y x^p y^q g(x, y). \quad (10.11)$$

Darin ist $g(x, y)$ der Grauwert des Punkts (x, y) . Die Summen laufen jeweils über alle Punkte des Objekts. Diese Definition ist übrigens völlig analog zu der Definition der Momente in der beschreibenden Statistik. Dasselbe gilt für die zentrierten Momente, die mit den Schwerpunktskoordinaten (x_s, y_s) definiert sind als:

$$\mu_{pq} = \sum_x \sum_y (x - x_s)^p (y - y_s)^q g(x, y). \quad (10.12)$$

Diese Momente können dann zusätzlich noch normiert werden:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^2}. \quad (10.13)$$

Mithilfe dieser normierten, zentrierten Momente lassen sich nach (Hu 1962) invariante Merkmale berechnen. Dabei werden Momente bis zur dritten Ordnung benötigt. Diese Methode der Objektbeschreibung für eine nachfolgende Klassifikation ist sehr leistungsfähig, aber wie man an den Formeln für die Momente bereits erkennt, auch sehr rechenaufwändig und wird daher in industriellen Anwendungen eher selten eingesetzt. (Sonka et al. 2008) diskutiert die mathematischen Eigenschaften der Momente ausführlicher und führt eine Vielzahl von Literaturstellen hierzu auf.

Fourierdeskriptoren: Eine weitere aufwändige aber leistungsfähige Methode zur Objektbeschreibung beruht auf der Fouriertransformation von Konturdarstellungen. Nach (Gonzalez und Woods 2008) betrachtet man dabei jeden Punkt der Kontur als komplexe Zahl, deren Realteil durch die x-Koordinate und deren Imaginärteil durch die y-Koordinate gegeben ist. Die gesamte Kontur stellt dann eine Folge komplexer Zahlen dar. Die Fourierkoeffizienten dieser Folge können als Beschreibung der Kontur benutzt werden. (Sonka et al. 2008) führt verschiedene Methoden zur Berechnung invarianter Merkmale aus Fourierdeskriptoren an. Dieses Verfahren zur Objektbeschreibung wird häufig für die Handschrifterkennung eingesetzt.

10.3 Grauwertmerkmale

Grauwertmerkmale beschreiben die Helligkeitsverteilung innerhalb eines Objekts. In dieser Gruppe finden wir einige der einfachsten Merkmale überhaupt, aber auch sehr rechenaufwändige Beschreibungsmöglichkeiten für die Oberflächenstruktur.

10.3.1 Statistik erster Ordnung

In den Formeln für die Berechnung statistischer Werte erster Ordnung kommt jeweils nur der Grauwert eines einzelnen Punkts vor – daher der Name. Typische derartige Merkmale sind:

- Mittlerer Grauwert
- Minimaler Grauwert
- Maximaler Grauwert
- Standardabweichung der Grauwerte

Es ist klar, dass Objekte sehr verschiedenen Aussehens die gleichen statistischen Eigenschaften erster Ordnung haben können. Abbildung 10.10 zeigt drei verschiedene Gebilde mit dem gleichen mittleren Grauwert von 127. Das homogen graue Bild unterscheidet sich natürlich in allen anderen Merkmalen der Grauwertstatistik erster Ordnung drastisch von den beiden Schachbrettmuster. Das einfache Schachbrett hat z. B. eine Standardabweichung von 127, während die homogene graue Fläche eine Standardabweichung von null aufweist. Die beiden unterschiedlich großen Schachbrettmuster sind jedoch auf diese Weise nicht auseinander zu halten. Die Ursache dieses Problems liegt darin, dass keine räumlichen Beziehungen zwischen Bildpunkten betrachtet werden.

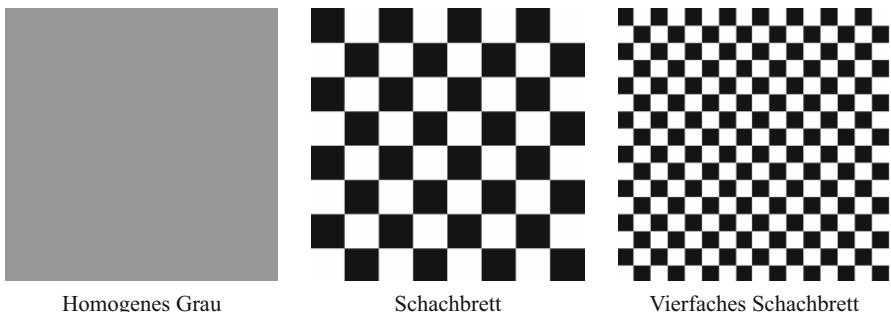


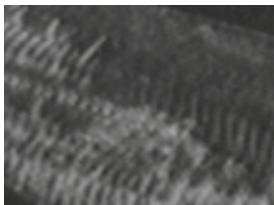
Abbildung 10.10: Objekte mit gleichem mittleren Grauwert

10.3.2 Texturmerkmale

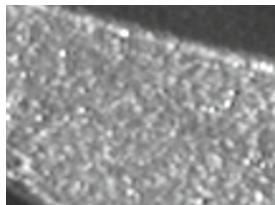
Unter *Texturen* versteht man Oberflächenstrukturen, wie etwa das typische Aussehen von Gusseisen, gebürstetem Stahl, Granit, Waschbeton usw. Textur ist einer der Begriffe in der Bildverarbeitung, die am schwierigsten exakt zu definieren sind. Entsprechend gibt es sehr viele, teils sehr aufwändige Methoden, Texturen zahlenmäßig zu beschreiben. Viele dieser Methoden beruhen auf statistischen Ansätzen. In (Julesz 1975) wurde gezeigt, dass Menschen keine Texturen unterscheiden können, deren statistische Eigenschaften zweiter Ordnung identisch sind. Zweiter Ordnung bedeutet, dass die Beziehungen zwischen Paaren von Bildpunkten ausgewertet werden. Diese Paare können sich in praktisch beliebigen Abständen in beliebigen Lagen zueinander befinden.

Die einfachsten statistischen Merkmale zweiter Ordnung beruhen auf Gradien-ten, d. h. Grauwertübergängen zwischen genau benachbarten Pixeln. Die beiden Schachbrettmuster in Abbildung 10.10 lassen sich mithilfe von Gradienten leicht auseinander halten. Das kleine Schachbrettmuster enthält zwar exakt ebenso viele schwarze und weiße Bildpunkte wie das große, aber es gibt doppelt so viele Übergänge zwischen Weiß und Schwarz. Entsprechend liegt der mittlere Gradient auch doppelt so hoch. Bereits diese einfachen Merkmale können zur Unterscheidung vieler technischer Oberflächen und zur Detektierung von Fehlstellen auf diesen

Oberflächen verwendet werden. Abbildung 10.11 zeigt den Unterschied zwischen einer polierten und einer rohen Aluminiumgussoberfläche bei sehr flacher Beleuchtung. Diese beiden Oberflächen lassen sich anhand des mittleren Gradienten leicht auseinander halten.



Poliert, mittlerer Gradient 15,3



Roh, mittlerer Gradient 25,1

Abbildung 10.11: Polierte und rohe Aluminiumgussoberfläche

Das Gebiet der Texturanalyse ist, wie gesagt, sehr vielfältig und verlangt je nach Oberfläche sehr hohen mathematischen Aufwand, bis hin zur Simulation ganz bestimmter Oberflächen mit speziellen stochastischen Modellen. Diese Vielfalt der Oberflächen und Analyse-Methoden hat dazu geführt, dass die meisten Lösungen im Bereich der Texturanalyse sehr spezifisch für die jeweilige Anwendung sind, weswegen es unmöglich ist, typische Lösungen, die auf etablierten Methoden basieren, anzugeben. Für weitere Informationen verweisen wir daher auf die Literatur, z. B. (Davis 1981), (van Gool et al. 1983), (Waszkewitz 1999), (Tönnies 2005), (Jähne 2005), (Nischwitz et al. 2007), (Sonka et al. 2008), (Gonzalez und Woods 2008).

10.4 Zusammenfassung

Objektmerkmale: Segmentierte Bildobjekte können durch verschiedene Merkmale beschrieben werden, um sie zu erkennen, zu klassifizieren und ihre Gültigkeit zu überprüfen. Die meisten geometrischen Merkmale sind nur für Objekte sinnvoll, die mit Schwellwert- oder Kantenmethoden gefunden wurden, da durch Template Matching segmentierte Objekte keinen visuell zusammenhängenden Bildstrukturen entsprechen müssen.

Umschreibendes Rechteck: Koordinaten und Größe des umschreibenden Rechtecks gehören zu den einfachsten geometrischen Objektmerkmalen. Das Rechteck kann entlang der Koordinatenachsen oder zur genaueren Beschreibung entlang der Hauptachse des Objekts ausgerichtet werden.

Rechteckigkeit: Dieses Merkmal ist definiert als das Verhältnis zwischen Fläche des Objekts und Fläche des umschreibenden Rechtecks, das hier stets nach der Hauptachse des Objekts ausgerichtet sein sollte.

Fläche: Die Fläche A eines Objekts ist definiert als die Anzahl der Bildpunkte innerhalb des Objekts.

Umfang: Der Umfang U eines Objekts kann direkt aus der Konturbeschreibung im Kettencode ermittelt werden, wenn man die berücksichtigt, dass ein Pixel in diagonaler Richtung um $\sqrt{2}$ länger ist als in horizontaler oder vertikaler. Der Umfang reagiert empfindlich auf Bildrauschen und auf die Bildauflösung, lässt sich aber bei den in industriellen Szenen häufigen vorkommenden glatten Bildkanten dennoch gut verwenden.

Kompaktheit: Dieses Merkmal ist definiert als $4\pi A/U^2$. Es nimmt für einen idealen Kreis seinen Maximalwert von eins an. Für jedes andere Objekt ist es kleiner.

Schwerpunktskoordinaten: Der Ort des Objektschwerpunkts kann aus allen Punkten des Objekts oder mit entsprechenden Korrekturfaktoren auch nur aus den Punkten der Objektkontur ermittelt werden. Die Schwerpunktskoordinaten sind durch die Mittelung über das gesamte Objekt subpixelgenau und ein sehr stabiles Merkmal.

Hauptträgheitsachse: Die Hauptachse eines Objekts wird meist mithilfe von Näherungsverfahren berechnet, da die formale Bestimmung sehr rechenaufwändig ist. Gängige Verfahren nehmen z. B. an, dass die Hauptachse genau durch einen Konturpunkt verlaufen muss oder berechnen sie aus einem umschreibenden Polygon.

Maximaler Feret-Durchmesser: Unter einem Feret-Durchmesser versteht man die Länge der Projektion eines Objekts auf eine beliebig orientierte Koordinatenachse. Der maximale Feret-Durchmesser ist der maximale Abstand zwischen zwei Punkten auf der Kontur des Objekts.

Radien: Unter einem Objektradius versteht man den Abstand zwischen einem Konturpunkt und dem Objektschwerpunkt. Mithilfe des minimalen und des maximalen Radius sowie ihrer relativen Lage lassen sich Objektformen beschreiben.

Konturkrümmung: Formal berechnet sich die Konturkrümmung aus der zweiten Ableitung der Kontur. Wegen der Störempfindlichkeit der numerischen Differenzierung verwendet man meist den Abstand eines Konturpunkts von einer Sehne fester Länge als Maß für die Krümmung. Man kann dabei konkave und konvexe Bereiche anhand des Vorzeichens unterscheiden. Krümmungswerte eignen sich z. B. zur Feststellung von Beschädigungen der Außenkanten eines Objekts.

Krümmungsvektor: Im Krümmungsvektor werden die Krümmungswerte für alle Punkte einer Kontur eingetragen. Wegen seines charakteristischen Verlaufs kann er zur Objekterkennung eingesetzt werden.

Faserlänge und -breite: Die Mittelfaser eines Objekts ist eine Linie innerhalb des Objekts, die den größtmöglichen Abstand von allen Konturpunkten hat. Sie kann mittels einer euklidischen Distanzkarre bestimmt werden. Ihre Länge bezeichnet

man als Faserlänge, den mittleren Abstand der Konturpunkt von dieser Linie als Faserbreite.

Euler-Zahl: Dieses Merkmal gibt die Anzahl der „Löcher“ in einem Objekt an. Bei einem durch Binärsegmentierung gefundenen schwarzen Objekt wäre dies die Anzahl aller weißen Bereiche innerhalb des Objekts. Viele technische Objekte, z. B. Stanzteile, lassen sich anhand dieses Merkmals unterscheiden.

Momente: Diese Merkmale sind analog zu Momenten in der beschreibenden Statistik definiert. Sie können zur rotations-, translations- und größeninvarianten Objektbeschreibung benutzt werden, sind aber sehr rechenaufwändig.

Fourierdeskriptoren: Hierunter versteht man die Koeffizienten einer Fouriertransformation der Objektkontur. Diese ebenfalls rechenaufwändigen Merkmale werden häufig in der Handschrifterkennung benutzt.

Grauwertstatistik erster Ordnung: In statistische Merkmale erster Ordnung gehen nur die Grauwerte einzelner Punkte ein. Räumliche Beziehungen lassen sich dadurch nicht beschreiben. Das einfachste Beispiel ist die mittlere Helligkeit.

Grauwertstatistik zweiter Ordnung: In statistische Merkmale zweiter Ordnung gehen die Grauwerte von Punktpaaren in definierten Abständen ein. Bei direkt benachbarten Punkten spricht man von Gradientenstatistik. Damit lassen sich bereits einfache Texturen beschreiben.

Texturmerkmale: Es gibt eine Vielzahl von Merkmalen für die Texturanalyse. Die meisten beruhen auf Grauwertstatistiken zweiter Ordnung auf der Grundlage teils sehr komplexer stochastischer Modelle.

11. Farbbildverarbeitung

Dieses Kapitel gibt eine Einführung in die Farbbildverarbeitung. Unter Farbbildverarbeitung verstehen wir die Auswertung der Farbinformation in Bildern für Anwendungen in der industriellen Bildverarbeitung. Obwohl Farbfernsehen schon seit vielen Jahrzehnten normal ist und im Consumer-Markt praktisch ausschließlich Farbkameras verfügbar sind, werden in der industriellen Bildverarbeitung nach wie vor die meisten Anlagen mit Graustufenkameras ausgerüstet und die Prüfaufgaben dem entsprechend nur mit der Auswertung der Helligkeitsinformationen gelöst. Das ist umso erstaunlicher, wenn man bedenkt, dass Farbe einen wesentlichen Teil der vom Menschen wahrgenommenen visuellen Information darstellt. Der Mensch kann laut (Russ 2007) lediglich 20–30 Graustufen unterscheiden, kann aber rund 1000 Farben und Farbnuancen unterscheiden, woraus offensichtlich wird, wie wichtig Farbinformationen für uns sind.

Es gibt aber einige Gründe, die gegen die Verwendung von Farbbildern in der Bildverarbeitung sprechen. Zunächst enthalten Farbbilder mehr Informationen über die aufgenommene Szene als Graustufenbilder, und dieses Mehr an Information bedeutet natürlich auch ein Mehr an Speicherplatzbedarf und Rechenaufwand. Einen Mehraufwand, den man wenn möglich vermeiden wird. Ein zweiter Grund ist, dass die meisten Bildverarbeitungsalgorithmen für Grauwertbilder entwickelt wurden und sich für die Auswertung von Grauwertbildern bewährt haben. Die Übertragung dieser etablierten Verfahren auf Farbbilder ist nicht immer ohne Weiteres möglich. Des Weiteren ist die Beschreibung von Farbe weitaus komplexer als die der Helligkeit. Während man die Helligkeit an einem Bildpunkt mit einem Wert beschreiben kann, benötigt man für die Beschreibung von Farbe im Allgemeinen drei voneinander unabhängige Werte, und dabei ist die Bedeutung der drei einzelnen Werte keineswegs festgelegt. Wie wir im ersten Anwendungsbeispiel sehen werden, gibt es verschiedene Modelle, mit denen Farben quantitativ beschrieben werden können, die jeweils ihre Vor- und Nachteile haben.

Trotzdem ist der Einsatz von Farbbildverarbeitung für manche Anwendungen notwendig. Dabei gibt es zwei typische Fragestellungen. Die eine Frage ist die nach der aktuellen Farbe eines bereits gefundenen Objekts. Dabei handelt es sich um eine Identifikationsaufgabe, die als Merkmal für die Klassifikation die Farbe des Objekts zugrunde legt. Dieser Fragestellung werden wir im ersten Anwendungsbeispiel begegnen.

Die andere typische Fragestellung ist die nach der Position oder Anwesenheit eines Objekts mit einer spezifischen Farbe. Dabei handelt es sich also um eine Segmentierung auf der Basis von Farbinformationen, bei der sich das Objekt durch die Farbe vom Hintergrund abhebt. Nach der Segmentierung können die Objekte natürlich genauso für Lageerkennung, Vermessung und Anwesenheits-

kontrolle verwendet werden wie die in Grauwertbildern segmentierten Objekte. Ein Beispiel für eine Farbsegmentierung zeigt die zweite Anwendung, die in diesem Kapitel präsentiert wird.

11.1 Farbidentifikation

Unser erstes Beispiel für die Farbbildverarbeitung kommt aus dem Bereich des Textileinzelhandels. Wahrscheinlich kennen wir alle die kleinen, bunten Größenringe aus Plastik, die an den Bügeln in Kaufhäusern und Bekleidungsgeschäften hängen, um die Größe der Kleidungsstücke zu bezeichnen. Wenn ein Kleidungsstück verkauft wird, wird der Größenring vom Bügel gelöst. Die dabei in großen Mengen anfallenden Größenringe sind natürlich völlig vermischt, wovon die Abbildung 11.1 einen Eindruck vermittelt. Um die Größenringe wieder verwenden zu können, müssen sie nach der von ihnen bezeichneten Größe sortiert werden, was die Aufgabe dieser Anwendung ist.



Abbildung 11.1: Größenringe vor der Sortierung (mit freundlicher Genehmigung der RNA Automation Ltd., Birmingham, UK)

11.1.1 Auswertestrategie

Das entscheidende Kriterium für die Sortierung ist die auf dem Größenring aufgeprägte und weiß lackierte Zahl oder Buchstabenkombination¹. Es wäre natürlich denkbar, diese Aufgabe nur mit Klarschrifterkennung im Grauwertbild wie in Kapitel 5 beschrieben zu lösen, allerdings gibt es einige Gründe, die gegen dieses Vorgehen sprechen:

- Da die Objekte rund sind, kommt es immer zu perspektivischen Verzerrungen der eingeprägten Zahlen, wie man in Abbildung 11.2 sehen kann, was die Klassifikation weniger robust macht. Dies kann, da insgesamt rund 70 verschiedene Kleidergrößen identifiziert werden müssen, durchaus problematisch sein.

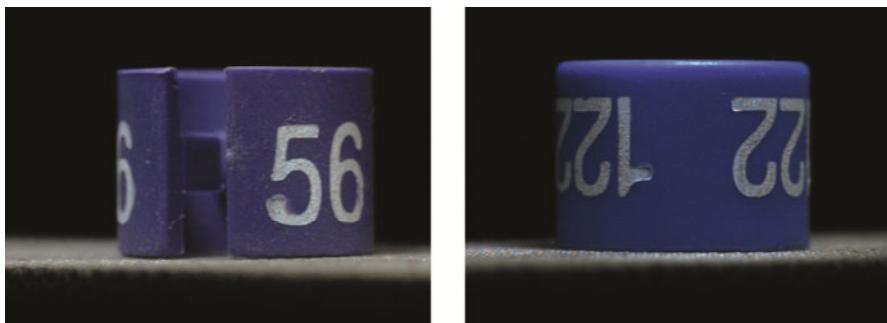


Abbildung 11.2: Perspektivische Verzerrung durch runden Objektkörper

- Die Zuführung der Größenringe stellt sicher, dass sich immer nur ein Objekt im Aufnahmebereich der Kamera befindet, allerdings ist die Orientierung des Rings zufällig, so dass in der Hälfte der Fälle die Beschriftung des Größenrings auf dem Kopf steht, wie man in Abbildung 11.2 ebenfalls sehen kann. Die Schrifterkennung muss also entweder auch auf dem Kopf stehende Zeichen lesen können, was die Anzahl der Klassen praktisch verdoppelt, oder aber man liest die Zahlen zweimal und dreht zwischen den Lesevorgängen das Bild auf den Kopf, wobei der zweimalige Lesevorgang natürlich auch doppelt so viel Rechenzeit in Anspruch nimmt.
- Wenn man Template Matching für die Identifikation der Größen einsetzt, bedeutet die große Anzahl verschiedener Klassen einen erheblichen Rechenaufwand, der bei der in dieser Anwendung geforderten hohen Taktrate von 300 Größenringen pro Minute durchaus kritisch werden kann.

Aus den genannten Gründen scheidet der naheliegende Ansatz, die Größe per Klarschrifterkennung zu identifizieren, aus. Das zweite hervorstechende Merkmal

¹ Da alle Größen außer S, M, L, und so weiter mit Zahlen bezeichnet werden, werden wir im Folgenden nur von Zahlen sprechen, was immer auch die mit Buchstaben bezeichneten Größen einschließt.

der Größenringe ist ihre Farbe, und so liegt es nahe, diese als Kriterium für die Identifizierung heranzuziehen. Die unterschiedlichen Konfektionsgrößen haben auch unterschiedliche Farben, woran man die einzelnen Größen beim Einkauf noch an der Stange hängend unterscheiden kann. Tatsächlich ist die Farbe der Größenringe keineswegs eindeutig. Zwar hat jede Größe nur eine Farbe, aber eine Farbe wird durchaus für unterschiedliche Konfektionsgrößen verwendet. Meistens kommen die Konfektionsgrößen, die die gleiche Farbe haben, nicht auf derselben Kleiderstange vor, da es sich bei der einen zum Beispiel um die englische Damengröße 8 (Abbildung 11.3 links) und bei der anderen um die Kindergröße 128 (Abbildung 11.3 rechts) handelt. Für unsere Identifikationsaufgabe bedeutet das aber, dass die Farbe der Größenringe keineswegs als alleiniges Kriterium ausreicht.

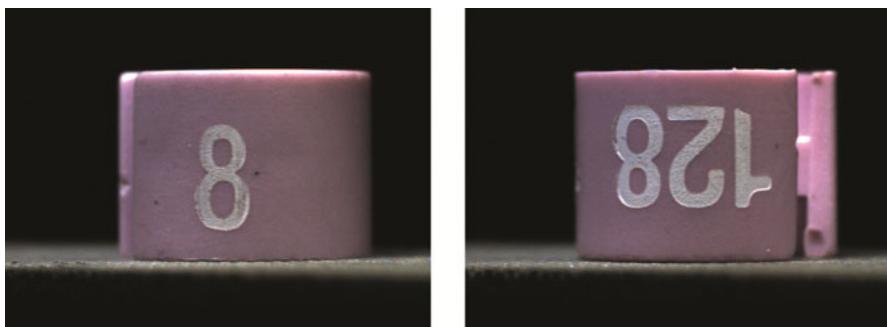


Abbildung 11.3: Verschiedene Größen auf Ringen identischer Farbe

Tatsächlich bietet es sich an, beide möglichen Kriterien kombiniert zu verwenden. In einem ersten Schritt kann mit der Identifizierung der Farbe beziehungsweise Farbkasse eine grobe Klassifizierung der Größenringe erfolgen. In einem zweiten Schritt muss bei der Klarschrifterkennung nur noch zwischen wenigen möglichen Größen unterscheiden werden. Durch die Kombination beider Identifikationsmethoden ist es möglich, die Größenringe mit ausreichender Zuverlässigkeit und hoher Geschwindigkeit zu sortieren.

11.1.2 Beleuchtung und Bildaufnahme

Die Größenringe werden auf einem Förderband liegend durch die Bildverarbeitungsstation transportiert. Um die Beschriftung möglichst gut erfassen zu können, sollten die Kameras waagerecht auf den Ring gerichtet sein. Da auf einem Größenring die Zahl dreimal aufgeprägt ist, schaut die Kamera im ungünstigsten Fall unter einem Winkel von 60° auf die Größe. Wie man in Abbildung 11.4 links gut sehen kann, ist unter einem Winkel von 60° die Beschriftung praktisch nicht mehr zu identifizieren. Aus diesem Grund werden die Größenringe von einer zweiten Kamera aufgenommen, deren Blickrichtung gerade um 60° gegenüber der ersten Kamera gedreht ist, wie in Abbildung 11.5 skizziert. Wie man auf der rechten Sei-

te von Abbildung 11.4 sehen kann, ist die Kamera damit direkt auf die Schrift gerichtet, wenn die andere Kamera mitten zwischen zwei Beschriftungen blickt. Bei dieser Konfiguration mit zwei Kameras ist die Beschriftung im ungünstigsten Fall um 30° gegenüber der Kameraachse verdreht, was kein Problem für die Identifikation darstellt.

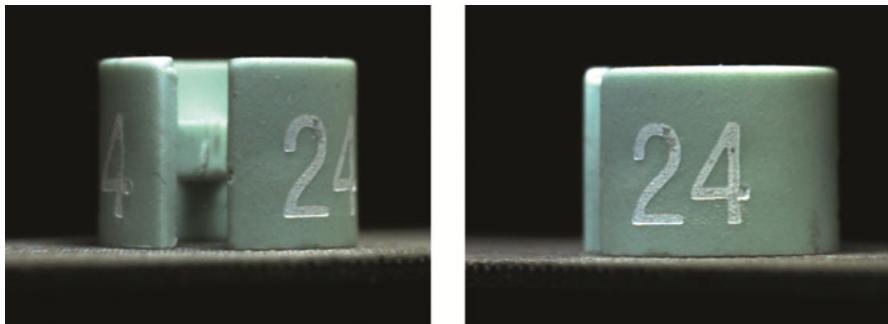


Abbildung 11.4: Größenring aus Sicht zweier um 60° verdrehter Kameras

Die Bildaufnahme wird durch eine Lichtschranke getriggert, so dass die Lage des Objekts im Bildfeld der Kamera sehr konstant ist. Da die Schrift im Vergleich zum Objekt sehr groß ist, sind Farbkameras mit der sehr geringen Auflösung von 640×480 Pixel völlig ausreichend.

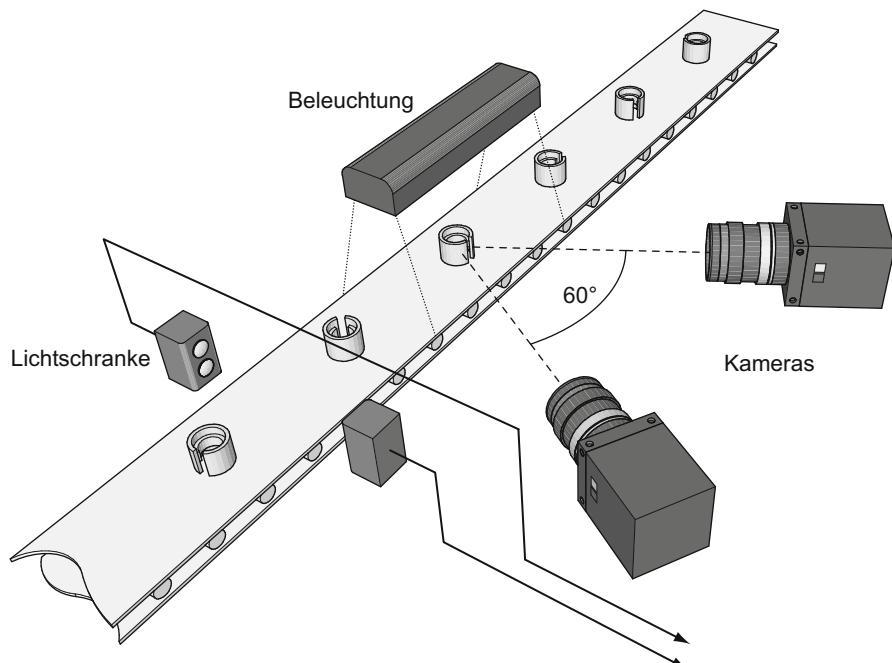


Abbildung 11.5: Aufbau der Station zur Erkennung von Größenringen

Um eine Farberkennung zu ermöglichen, ist eine Auflichtbeleuchtung natürlich unerlässlich. Allerdings scheidet eine waagerechte Beleuchtung aus, denn durch die zylindrische Form der Größenringe würde das Licht immer irgendwo direkt in die Kameras reflektiert werden, was sowohl die Klarschriftlesung als auch die Farberkennung stören würde. Von daher bleibt praktisch nur noch eine Beleuchtung von oben. Diese sollte möglichst diffus sein, um eine Farberkennung auf der gesamten Ringbreite zu ermöglichen, und wird durch ein linienförmiges Leuchtelement aus Weißlicht-LEDs realisiert. Glücklicherweise ist der Kontrast zwischen der weißen Beschriftung und dem farbigen Ringkörper bei allen Farben groß genug, um eine Schrifterkennung bei der diffusen Auflichtbeleuchtung zu erlauben. Wenn für die Schrifterkennung eine Dunkelfeldbeleuchtung erforderlich wäre, wäre es kaum möglich, mit derselben Kamera und Beleuchtung auch die Farbe des Rings zu erfassen. Das intensive Licht der LEDs erzeugt am oberen Rand der Ringe bei manchen Farben einen helleren Kranz, der allerdings bei der Schrifterkennung nicht stört, da er weit genug von der Beschriftung entfernt ist, und bei der Farberkennung dieser Bereich von der Auswertung ausgenommen werden kann.

11.1.3 Farbklassifizierung

Der erste Schritt unserer Auswertung ist die Einteilung der Größenringe in Farbklassen. Dazu ist eine Farbbezeichnung notwendig, anhand der der Computer diese Trennung durchführen kann. Wir kennen wohl alle Fälle, in denen zwei Personen ein und dieselbe Farbe unterschiedlich bezeichnen; so ist beispielsweise für den einen ein Objekt gelb, das für den anderen orange ist. Für die Klassifizierung brauchen wir deshalb eine quantitative Beschreibung der Farbe.

Die einfachste Möglichkeit der Farbbezeichnung ist es, die RGB-Werte als Grundlage für die Klassifizierung zu nehmen. Der RGB-Farbraum ist sehr gut geeignet, um die Farbdaten zu beschreiben, die von einer Kamera aufgenommen werden. Es gibt allerdings zwei Gründe, die gegen die Verwendung des RGB-Farbraums sprechen: Zum einen entspricht die Darstellung nicht der menschlichen Wahrnehmung von Farbe, was die Interpretation der Werte sehr erschwert und zum anderen kann man die Beurteilung der Farbe nicht von der Helligkeit trennen. Der Mensch trennt im Allgemeinen die Farbe von der Helligkeit, so sind zum Beispiel ein helles Blau und ein dunkles Blau für uns die gleiche Farbe mit unterschiedlicher Helligkeit, während beispielsweise ein helles Blau und ein helles Grün zwar dieselbe Helligkeit, aber unterschiedliche Farbtöne haben. Mit dem Farbton und der Helligkeit lässt sich eine Farbe noch nicht vollständig beschreiben: Es fehlt noch eine Möglichkeit, zwischen einer kräftigen Farbe und einer Pastellfarbe zu unterscheiden. Diese Farbeigenschaft wird als Sättigung bezeichnet wobei kräftige, „satte“ Farben eine hohe Sättigung haben, während blasse oder „schmutzige“ Farben eine niedrige Sättigung haben.

Der sogenannte HSI-Farbraum beschreibt Farbe auf genau diese Weise, wobei der Name sich aus dem Farbton (englisch Hue), der Sättigung (englisch Saturation)

on) und der als Intensität (englisch Intensity) bezeichneten Helligkeit herleitet. Beim Farbton greift der HSI-Farbraum die traditionelle Vorstellung des Farbkreises auf und beschreibt den Farbton durch einen Winkel. Als Farbton für null Grad wurde Rot festgelegt, und von dort aus steigt der Winkel über Gelb, Grün, Blau und Violett an, um bei 360° wieder zu Rot zu gelangen. Der Farbkreis wird zur Farbscheibe, indem man die Sättigung der Farbe als Radius einführt. Im Mittelpunkt, also bei einer Sättigung von null, ist die Farbscheibe grau, während am Rand der Farbscheibe die vollständig gesättigten, eben „satten“ Farben sind. Derartige Farbscheiben sind für unterschiedliche Helligkeiten in Abbildung 11.6 dargestellt. Farbton als Winkel, Sättigung als Radius und Intensität als Höhe bilden zusammen ein Zylinderkoordinatensystem, mit dem der Farbraum beschrieben wird. Wie man in Abbildung 11.6 sehen kann, ist die Farbscheibe bei mittlerer Helligkeit am größten. Tatsächlich können Farben nur bei mittlerer Helligkeit voll gesättigt sein. In den Extremfällen der maximalen und minimalen Helligkeit ist gar keine Sättigung möglich, denn Weiß und Schwarz haben keinerlei Farbanteile.

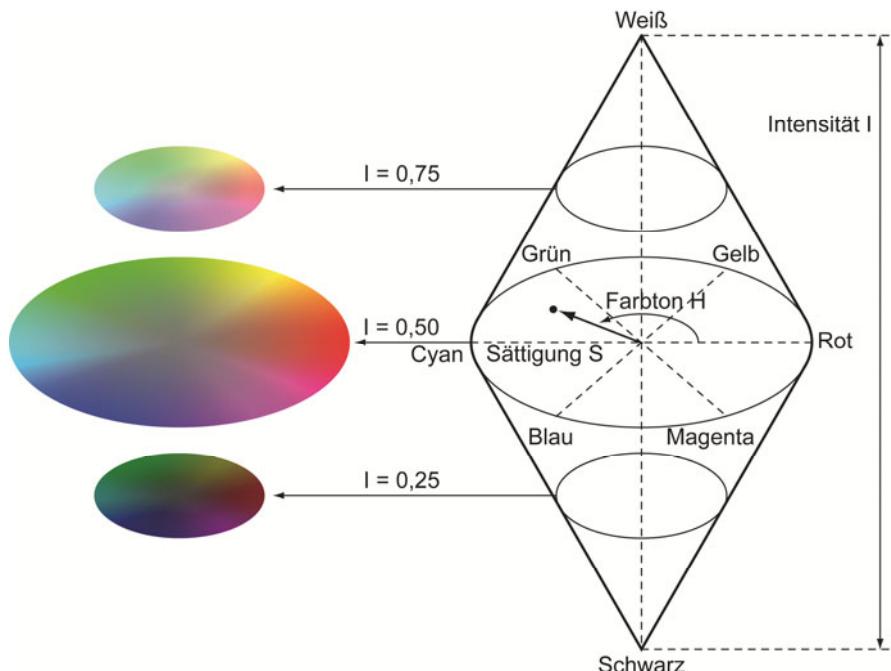


Abbildung 11.6: Der HSI-Farbraum

Da die Bilddaten von Kameras meistens im RGB-Farbraum vorliegen, müssen die Farbdaten in den HSI-Farbraum transformiert werden. Laut (Gonzalez und Woods 2008) kann man RGB-Farbdaten mit

$$\begin{aligned}
 H &= \begin{cases} \theta & , \text{falls } B \leq G \\ 360^\circ - \theta & , \text{falls } G < B \end{cases}, \\
 S &= 1 - \frac{\text{Min}\{R, G, B\}}{R + G + B} \quad \text{und} \\
 I &= \frac{1}{3}(R + G + B)
 \end{aligned} \tag{11.1}$$

in den HSI-Farbraum transformieren, wobei

$$\theta = \arccos \left(\frac{2R - G - B}{2\sqrt{(R - G)^2 + (R - B)(G - B)}} \right).$$

Nachdem wir den HSI-Farbraum jetzt kennen, können wir uns der praktischen Durchführung der Prüfung zuwenden. Zunächst müssen wir den Bereich festlegen, in dem die Farbe bestimmt werden soll. Man könnte die Farbe natürlich über das gesamte Objekt hinweg messen, aber dabei geht die weiße Beschriftung mit in die Bewertung ein. Aus diesem Grund ist es besser, nur einen Teilbereich für die Farbbestimmung heranzuziehen. Im Bereich über der Beschriftung kann es durch die Beleuchtung zu Reflexen kommen, die die Farbbestimmung verfälschen würden, deshalb ist der Bereich unterhalb der Größenangabe am besten geeignet. Aufgrund der runden Form des Objekts erscheinen die Ränder des Größenrings dunkler als die Mitte des Rings. Zwar ist im HSI-Farbraum die Beschreibung der Farbe von der Helligkeit entkoppelt, allerdings können in sehr dunklen Bereichen Farben nicht mehr erkannt werden, so dass man mit dem Suchbereich auch zum linken und rechten Rand etwas Platz lassen wird. Als Auswertebereich bleibt somit ein Band unterhalb der Beschriftung.

Eine Positionierung des Arbeitsbereichs ist bei dieser Anwendung unnötig. Aufgrund der Triggerung der Bildaufnahme durch die Lichtschranke und die konstante Geschwindigkeit des Transportbands ist die horizontale Lage des Objekts ausreichend stabil. In vertikaler Richtung ist die Lage durch das Transportband völlig festgelegt. Dabei ist lediglich zu beachten, dass der Abstand zwischen der Beschriftung und dem oberen und unteren Rand des Größenrings unterschiedlich ist. Deshalb hängt die Höhe der Unterkante der Beschriftung davon ab, ob der Ring auf dem Kopf liegt oder nicht.

Die eigentliche Farbbestimmung besteht darin, das Bild in den HSI-Farbraum zu konvertieren und dann für jeden Kanal den mittleren Grauwert im Suchbereich zu bestimmen. Um die Auswirkung von Kamerarauschen und Verunreinigungen des Größenrings zu reduzieren, sollte man den Messbereich vor der Transformation in den HSI-Farbraum glätten.

Für jede Größenringfarbe muss der Bereich für die drei Farbkoordinaten hinterlegt werden, in dem sich die gemessenen Werte befinden dürfen. Bei 70 möglichen Farben bedeutet dies, dass die gemessenen Werte im ungünstigsten Fall 70-

mal mit vorgegebenen Werten verglichen werden müssen. Um die Anzahl der Vergleiche zu reduzieren, empfiehlt es sich, eine Vorklassifizierung durchzuführen. Diese teilt die Farben anhand des Farbtöns in acht Klassen, die den Farbtonkreis in jeweils 45° breite Bereiche einteilen. Je nach Bereich sind dadurch für die Farbklassifizierung nur fünf bis zwanzig Vergleiche erforderlich.

Ein Problem bei der Farbklassifikation ist noch offen: Die Ringe haben eine Öffnung, um sie auf die Bügel stecken zu können. Diese Bereiche werden nicht gut beleuchtet, so dass sie erheblich dunkler sind als die sonstige Oberfläche des Größenrings. Diese Bereiche müssen von der Farbauswertung ausgenommen werden, daher werden statt eines großen Auswertebereichs mehrere kleine Auswertebereiche verwendet, wie in Abbildung 11.7 dargestellt. Damit können Auswertebereiche, die sich über der Ringöffnung befinden, vor der Farbbestimmung aussortiert werden. Die Auswertebereiche, die ausschließlich im Schatten liegen, sind anhand der geringen Helligkeit zu erkennen, während die Übergänge zwischen abgeschatteten Bereichen und der farbigen Oberfläche anhand des großen Helligkeitskontrasts zu erkennen sind. Vor der Bestimmung der Farbe sortiert man somit die Auswertebereiche aus, die entweder sehr dunkel sind oder einen sehr großen Kontrast haben. Damit gehen die abgeschatteten Bereiche der Öffnung nicht in die Farbauswertung ein.

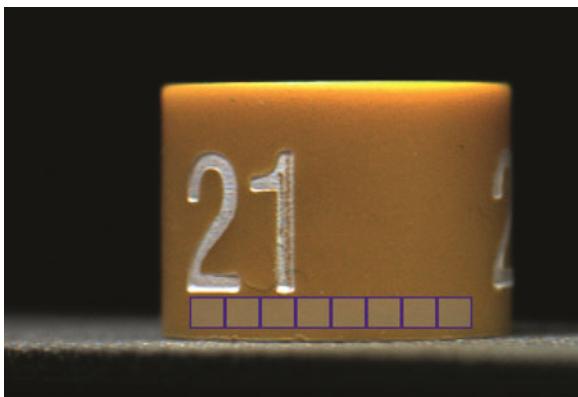


Abbildung 11.7: Arbeitsbereich für die Bestimmung der Farbe

Die Verwendung von mehreren Auswertebereichen für die Farberkennung erlaubt es auch, zur Unterdrückung von Ausreißern die Auswertebereiche mit den größten und kleinsten Farbwerten vor der Auswertung auszuschließen. Die Farbauswertung wird in Tabelle 11.1 zusammengefasst.

Beispielbeschreibung 11.1. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 11.1.3 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Farbe
Beleuchtung	Auflichtbeleuchtung mit Weißlicht-LEDs
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt mehrere kleine Arbeitsbereiche unterhalb der Beschriftung mit ausreichend Abstand zu den seitlichen Rändern. 3. <i>Farbbild konvertieren</i>, konvertiert das Bild in ein Intensitätsbild. 4. <i>Arbeitsbereiche Merkmale erzeugen</i>, ermittelt Helligkeit und Kontrast. 5. <i>Arbeitsbereiche filtern</i>, schließt die Öffnung des Größenrings anhand der Helligkeit und des Kontrasts von der Farbauswertung aus. 6. <i>Farbbild konvertieren</i>, konvertiert das Bild in ein Farbtonbild. 7. <i>Arbeitsbereiche kopieren</i>, kopiert die Arbeitsbereiche für die Ermittlung des Farbtoms. 8. <i>Arbeitsbereiche Merkmale erzeugen</i>, ermittelt den Farbton als mittleren Grauwert. 9. <i>Arbeitsbereiche sortieren</i>, sortiert den Arbeitsbereich mit dem höchsten Farbton aus, um Ausreißer zu unterdrücken. 10. <i>Arbeitsbereiche sortieren</i>, sortiert den Arbeitsbereich mit dem niedrigsten Farbton aus, um Ausreißer zu unterdrücken. 11. <i>Farbbild konvertieren</i>, konvertiert das Bild in ein Sättigungsbild. 12. <i>Arbeitsbereiche Merkmale erzeugen</i>, ermittelt die Sättigung als mittleren Grauwert. 13. <i>Arbeitsbereiche sortieren</i>, sortiert den Arbeitsbereich mit der höchsten Sättigung aus, um Ausreißer zu unterdrücken. 14. <i>Arbeitsbereiche sortieren</i>, sortiert den Arbeitsbereich mit der niedrigsten Sättigung aus, um Ausreißer zu unterdrücken. 15. <i>Arbeitsbereiche kombinieren</i>, fasst die Arbeitsbereiche mit Farbton und Sättigungswerten zusammen.

11.1.4 Auswahl des Kamerabildes für die Beschriftungserkennung

Die Identifizierung der Farbe des Größenrings ist wie gesagt nur der erste Schritt der Auswertung. Der zweite Schritt ist die Erkennung der Beschriftung. Es werden zwei Kameras verwendet, um sicherzustellen, dass die Schrift in mindestens einem Kamerabild gut erkennbar ist. Da man zunächst nicht weiß, in welchem, muss im nächsten Schritt bestimmt werden, in welchem Kamerabild die Beschriftung besser zu lesen ist. Die Beschriftung ist umso besser zu identifizieren, je direkter die Kamera darauf gerichtet ist. Wenn die Kamera direkt auf die Beschriftung gerichtet ist, ist diese wesentlich breiter, als wenn die Kamera die Beschriftung unter einem großen Winkel aufnimmt. Dies macht man sich zunutze

und wählt das Kamerabild für die Beschriftungsidentifizierung, in dem die Breite der Beschriftung am größten ist. Der nächste Schritt ist somit, die Breite der Beschriftung in beiden Kamerabildern zu messen und das Bild mit der größeren Breite für das Lesen der Konfektionsgröße auszuwählen. Um die Breite der Beschriftung zu messen, muss die Beschriftung als ein Objekt segmentiert werden, was bei Größenbezeichnungen, die aus mehreren Ziffern bestehen, bedeutet, die Ziffern zusammenzufassen.

Da die Beschriftung immer weiß ist, ist sie immer heller als die Farbe des Rings. Deshalb kann man die Messung der Beschriftungsbreite in einem Grauwertbild, das die Helligkeitsverhältnisse im Farbbild wiedergibt, durchführen. Um dieses Helligkeitsbild zu erhalten, könnte man das Bild in den HSI-Raum transformieren und dann nur den I-Kanal verwenden. Im Allgemeinen wird man aber den Aufwand dieser Transformation vermeiden und das Helligkeitsbild direkt aus dem RGB-Bild berechnen, indem man einen Mittelwert aus Rot-, Grün- und Blauwert bildet. Wenn man die drei Farbwerte gleich gewichtet addiert, entspricht das resultierende Helligkeitsbild nicht unserem Helligkeitsempfinden. Tatsächlich hat das grüne Licht für den Menschen einen wesentlich stärkeren Einfluss auf die empfundene Helligkeit als das rote oder blaue Licht. Aus diesem Grund lässt man die einzelnen Farbkanäle gewichtet in den Helligkeitswert eingehen. (Russ 2007) gibt folgende Gewichte für die Umwandlung an:

$$H(x, y) = 0,299 \cdot R(x, y) + 0,587 \cdot G(x, y) + 0,114 \cdot B(x, y) \quad (11.2)$$

Mit dieser Formel kann für jedes Pixel des Farbbildes direkt ein Helligkeitswert berechnet werden und damit das Farbbild in ein Grauwertbild umgewandelt werden. Aus der Formel 11.2 kann man entnehmen, dass der Grünwert mehr als die Hälfte zum Helligkeitswert beiträgt.

Aufgrund der unterschiedlichen Helligkeiten des Hintergrunds ist es empfehlenswert, die Schrift anhand der Kanten zum Hintergrund zu segmentieren. Dazu wird ein Kontrastfilter mit einer großen Filtermaske auf das Bild angewandt, dessen Ergebnis in Abbildung 11.8 links dargestellt ist. Damit erscheinen die Ränder der Beschriftung hell im Bild. Aber auch die Kanten des Rings werden hell dargestellt, da sich der Ring je nach Farbe mehr oder weniger stark vom Hintergrund unterscheidet. Auch die Stärke des Kontrastes zwischen Beschriftung und Ring hängt natürlich von der Ringfarbe ab. Um diese Unterschiede auszugleichen, wird das Bild normiert, was in Abbildung 11.8 rechts zu sehen ist.

Nach der Kontrastfilterung und der Normierung sind die Ränder der Ziffern vollständig hell im Bild zu sehen, während das Innere der Ziffern und der Rest des Größenrings dunkel sind. In diesem Kantenbild werden die Ziffern binär segmentiert, den Suchbereich kann man dabei so wählen, dass der obere und untere Rand des Größenrings außerhalb liegt. Bei der Segmentierung muss man darauf achten, dass die gefundenen Objekte gefüllt werden, da anderenfalls das dunkle Innere der Ziffern als Loch im Objekt verbleiben würde. Neben den gesuchten Ziffern werden auch der linke und rechte Rand des Rings als Objekt erzeugt. Da dieser Rand

höher und schmäler als die Ziffern ist, kann man ihn anhand dieser Merkmale herausfiltern.

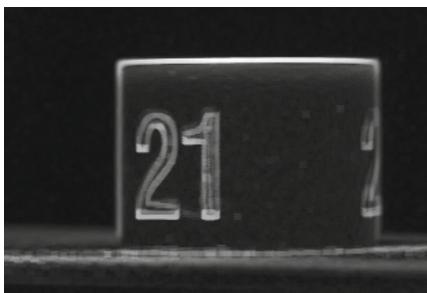
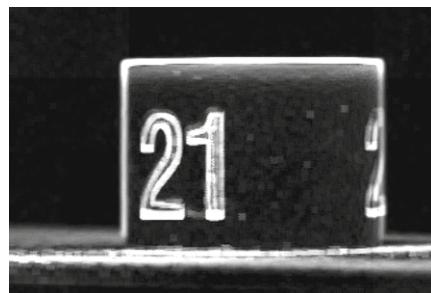


Bild nach Kontrastfilterung



Normiertes Bild

Abbildung 11.8: Mit Kontrastfilter bearbeitetes und normiertes Bild

Um die Ziffern einer Zahl zusammenzufassen, erzeugen wir ein künstliches Bild, in dem die gefundenen Objekte weiß eingezeichnet sind und der Rest schwarz ist, was in Abbildung 11.9 links dargestellt ist. Durch mehrfache Dilatation mit einer speziell angepassten Filtermaske werden die Ziffern breiter gemacht, so dass die Ziffern einer Zahl einen zusammenhängenden weißen Bereich bilden, was in der Abbildung 11.9 rechts zu sehen ist. Als Filtermaske wird ein waagrechter Strich verwendet, der ein Pixel hoch ist und elf Pixel breit ist. Bei der Dilatation mit dieser Maske werden Objekte breiter, aber ihre Höhe ändert sich nicht. Bei den relativ nah beieinander stehenden Ziffern einer Konfektionsgröße bewirkt das, dass die einzelnen Ziffern zu einem hellen Bereich verschmelzen.



Binarisierte Ziffern



Zusammengefasste Ziffern

Abbildung 11.9: Gezeichnete und zusammengefasste Objekte

Die so zu einem Bereich verbundenen Ziffern können dann durch Binarisierung als ein Objekt segmentiert werden. Bei einem ungünstigen Blickwinkel der Kamera können zwei Beschriftungen im Bild sichtbar sein. In diesem Fall gilt natürlich das Gleiche für die beiden Beschriftungen in einem Bild wie für die Beschriftungen in den beiden verschiedenen Kamerabildern: Die breitere Beschriftung ist für

die Erkennung der Konfektionsgröße besser geeignet. Deshalb verwendet man für die Analyse nur das breiteste Objekt und kann das andere aussortieren.

Die Breite der verbliebenen Beschriftung kann dann vermessen werden und das Kamerabild mit der breiteren Beschriftung ausgewählt werden, was in dem Beispiel in Abbildung 11.10 das rechte Kamerabild ist. Durch die Dilatation ist die gemessene Breite natürlich wesentlich größer als die tatsächliche Breite der Beschriftung, was man in der Abbildung 11.10 ebenfalls gut sieht. Wenn man die tatsächliche Breite der Beschriftung messen müsste, wäre das hier beschriebene Verfahren ungeeignet. Für die Auswahl des richtigen Kamerabilds ist aber nur der Vergleich zwischen den beiden Breiten relevant. Die Bestimmung der Breite der Beschriftung für die Auswahl des besser geeigneten Kamerabilds ist in Tabelle 11.2 zusammengefasst.

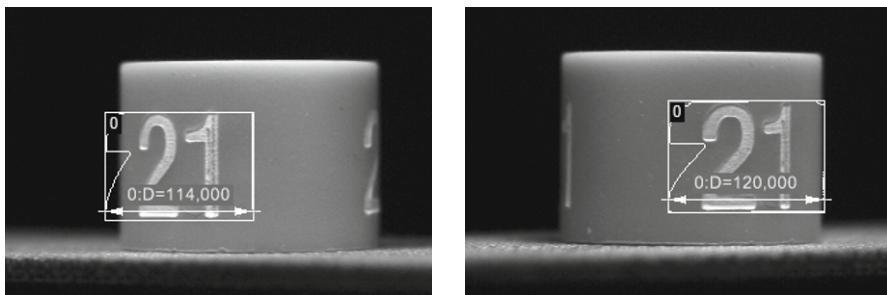


Abbildung 11.10: Breiten der erweiterten Beschriftung in beiden Kamerabildern

Beispielbeschreibung 11.2. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 11.1.4 zusammen. Die Messung wird jeweils für beide Kamerabilder durchgeführt.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Farbe
Beleuchtung	Auflichtbeleuchtung mit Weißlicht-LEDs
Prüfschritte	Das Beispiel verwendet folgenden Prüfschrittablauf: <ol style="list-style-type: none"> <i>Bildaufnahme</i> <i>Farbbild konvertieren</i>, wandelt das Bild in ein Helligkeitsbild um. <i>Bildgröße verändern</i>, halbiert die Größe des Bildes, um eine schnellere Auswertung zu ermöglichen. <i>Bild filtern</i>, hebt durch Filterung mit einem Kontrastfilter die Helligkeitsübergänge hervor. <i>Bild normieren</i>, verstärkt den Kontrast im Bild. <i>Arbeitsbereiche definieren</i>, definiert einen Suchbereich in Höhe der Beschriftung über die gesamte Bildbreite. <i>Binärschwelle erzeugen</i>, berechnet eine Schwelle für die Segmentierung der Schrift.

8. *Arbeitsbereiche binär erzeugen*, segmentiert die Zahlen.
 9. *Arbeitsbereiche Merkmale erzeugen*, berechnet die Höhe, Breite und Fläche der Objekte, um den seitlichen Rand des Größenrings auszusortieren.
 10. *Arbeitsbereiche filtern*, filtert den seitlichen Rand des Größenrings und weitere Störungen aus.
 11. *Arbeitsbereiche zeichnen*, zeichnet ein künstliches Bild, bei dem die gefundenen Ziffern weiß sind und der Rest des Bilds schwarz ist.
 12. *Bild auf Arbeitsbereichen filtern*, dilatiert die künstlichen Ziffern, um sie zusammenwachsen zu lassen.
 13. *Arbeitsbereiche binär erzeugen*, erzeugt die zusammengewachsenen Ziffern als ein Objekt.
 14. *Arbeitsbereiche Merkmale erzeugen*, berechnet die Breite der Objekte, um die breiteste Zifferngruppe im Bild zu bestimmen.
 15. *Arbeitsbereiche sortieren*, sortiert alle Objekte außer dem Breitesten aus.
 16. *Maße erzeugen*, bestimmt die Breite der am besten erkennbaren Beschriftung im Bild.
 17. *Maßliste in Ablage kopieren*, kopiert die Breite der Beschriftung in die Ablage, um sie mit der Breite im anderen Kamerabild zu vergleichen.
-

11.1.5 Erkennung der Beschriftung

Für die Erkennung der Beschriftung kann man entweder die Zeichen rastern und dann mit einem Klassifikator identifizieren oder die Zeichen direkt mit Template Matching suchen. Bei dieser Anwendung hat man sich für Template Matching entschieden. Gegen einen Klassifikator sprach, dass die Rasterung der Zeichen sehr empfindlich auf die Verzerrung durch die runde Objektform reagiert hat. Ein weiterer Grund war, dass aufgrund der vielen verschiedenen Konfektionsgrößen das Erzeugen von Trainingsdaten einen erheblichen Aufwand darstellt. Es zeigte sich, dass für Template Matching im Schnitt drei bis vier Muster für eine Konfektionsgröße völlig ausreichend sind. Um ein neuronales Netz zu trainieren, sind aber wesentlich mehr Muster notwendig, was den Aufwand noch weiter gesteigert hätte. Schließlich führen die sehr unterschiedlichen Breiten zwischen Konfektionsgrößen mit einer Ziffer und solchen mit drei Ziffern dazu, dass es schwierig ist, eine geeignete Rasterung der Zeichen zu finden. Natürlich hätte man jede Ziffer einzeln lesen können, allerdings hätte dies die Segmentierung verkompliziert.

Kritisch hätte bei dieser Anwendung die Laufzeit von Template Matching sein können. Allerdings ist die Anzahl der Templates, nach denen gesucht wird, gering genug, so dass die hohe Taktrate gehalten werden konnte. Die geringe Zahl von erforderlichen Templates hat zwei Gründe: Der erste Grund ist, dass es wie gesagt ausreicht, nach drei oder vier Templates pro Konfektionsgröße zu suchen. Der zweite Grund ist, dass durch die vorhergehende Farbklassifizierung die Zahl der möglichen Konfektionsgrößen mit meistens zwei bis drei sehr gering ist. Insgesamt muss pro Farbklasse im Schnitt nach zehn Templates gesucht werden.

Zwei weitere Aspekte wirken sich ebenfalls positiv auf die Laufzeit von Template Matching aus: Zur Bestimmung, welche Kamera die Konfektionsgröße besser lesen kann, haben wir in Abschnitt 11.1.4 bereits die Beschriftung binär erzeugt. Dieses Objekt eignet sich sehr gut als Positionsreferenz, um den Suchbereich für Template Matching zu positionieren. Dadurch kann der Suchbereich für Template Matching verhältnismäßig klein gewählt werden. Der andere Aspekt ist, dass die Beschriftung ein sehr großflächiges Objekt darstellt, was die Verwendung einer sehr großen Schrittweite für die Grobsuche erlaubt. Bei dieser Anwendung konnte man sogar noch einen Schritt weitergehen: Um den Rechenaufwand zu reduzieren, wurde bereits vor der Bestimmung der besseren Kameraperspektive die Bildgröße halbiert, d. h. jede zweite Spalte und Zeile des Bildes wurde weggelassen. Die übrig gebliebenen Pixel reichen sowohl für die Auswahl des Bildes als auch für die Erkennung der Beschriftung mit Template Matching völlig aus. Aufgrund dieser Rahmenbedingungen ist es möglich, bei dieser Anwendung mit Template Matching die Prüfung schnell genug durchzuführen.

Wir schrieben bereits, dass die Größenringe auch auf dem Kopf liegen können. Auf dem Kopf liegende Beschriftungen werden von Template Matching natürlich nicht erkannt. Es gibt verschiedene Möglichkeiten, mit diesem Problem umzugehen:

- Man kann gedrehte Templates hinterlegen, was die Zahl der Templates erhöht.
- Man kann die Templates zur Laufzeit drehen, was für jedes Template zu zusätzlichem Rechenaufwand führt.
- Man kann das Bild drehen und die Suche erneut durchführen, was für ein Bild nur einen einmaligen Rechenaufwand bedeutet.

Bei dieser Anwendung wird das Bild gedreht, um einerseits die Anzahl der Templates zu begrenzen, und weil sich andererseits das Bild schneller drehen lässt als zehn Templates.

Man kann Template Matching im Farbbild ausführen und dabei die Korrelation bezüglich des Farbwertes berechnen. In unserer Anwendung haben wir die Farbinformation aber schon für die Einteilung der Ringe in Klassen verwendet. Somit wird das Template Matching nur noch für Konfektionsgrößen gleicher oder sehr ähnlicher Farben ausgeführt, so dass die Korrelation im Farbbild keine zusätzlichen Informationen mehr gegenüber dem Helligkeitsbild enthält. Zudem bedeutet die Berechnung der Korrelation im Farbbild natürlich auch den dreifachen Rechenaufwand gegenüber der Korrelation im Grauwertbild. Aus diesen Gründen wird auch für die Erkennung der Konfektionsgröße das Helligkeitsbild verwendet, in dem die Breite der Beschriftung bestimmt wurde. In Tabelle 11.3 ist die Identifikation der Beschriftung zusammengefasst.

Beispielbeschreibung 11.3. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 11.1.5 zusammen. Wenn keine Konfektionsgröße gefunden wird, wird das Bild gedreht und die Suche erneut ausgeführt.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Farbe
Beleuchtung	Auflichtbeleuchtung mit Weißlicht-LEDs
Prüfschritte	Das Beispiel verwendet folgenden Prüfschrittablauf: <ol style="list-style-type: none"> <i>Bild aus Ablage kopieren</i>, das bei der Bestimmung der Beschriftungsbreite verkleinerte Bild wird hier weiter verwendet. <i>Arbeitsbereiche definieren</i>, definiert einen Suchbereich für die Schrifterkennung. Die Position wird von der Bestimmung der Beschriftungsbreite übernommen. <i>Template Matching</i>, sucht die für diese Farbe möglichen Konfektionsgrößen. <i>Arbeitsbereiche Anzahl überprüfen</i>, prüft ob eine passende Konfektionsgröße gefunden wurde.

11.2 Farbsegmentierung

Dieses Beispiel stammt aus der Elektroindustrie. Es handelt sich um ein Relais, bei dem die Verzinkung der Spulendrahtenden geprüft werden muss. Der Draht besteht aus lackiertem Kupfer. Um diesen leitend mit dem Kontakt zu verbinden, wird das Ende verzinkt. Bei diesem Prozess verdampft der isolierende Lack, so dass die Oberfläche des Drahtendes elektrisch leitfähig wird. In Abbildung 11.11 sieht man eine Windung der Kupferspule und das verzinkte Drahtende, das auf dem Messingkontakt festgeklemmt ist.

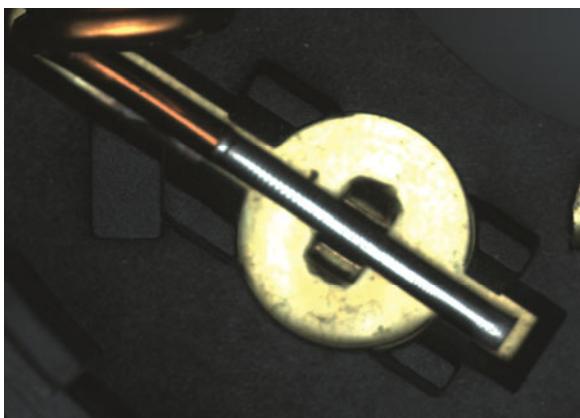


Abbildung 11.11: Verzinktes Ende eines Kupferdrahtes auf Messingkontakt

Bei dieser Prüfaufgabe handelt es sich im wahrsten Sinne des Wortes um eine Anwesenheitskontrolle, denn es muss geprüft werden, ob ein verzinktes Drahtende auf dem Kontakt anwesend ist oder nicht. Das Besondere an dieser Prüfaufgabe ist, dass sich die Helligkeit zwischen verzinktem Draht, unverzinktem Draht und Kontakt für eine stabile Binarisierung zu wenig unterscheidet. Die drei Metalle unterscheiden sich aber deutlich in ihrer Farbe, und somit kann man die Aufgabe mittels Farbsegmentierung lösen.

11.2.1 Beleuchtung

Metallische Oberflächen zeichnen sich dadurch aus, dass sie Licht sehr stark reflektieren. Deshalb sind sie eher schwierig zu beleuchten, insbesondere wenn sie, wie bei einem Draht, gekrümmmt sind. Dann wird das einfallende Licht fast immer von irgendeinem Punkt aus direkt in die Kamera reflektiert. Man kann versuchen, die Reflexe durch eine sehr flache Dunkelfeldbeleuchtung zu minimieren. Dies ist allerdings im vorliegenden Fall nicht möglich, da dafür der Rand des Relais zu weit über den Kontakt hinausragt. Damit liegt der Kontakt bei sehr flachem Licht im Schatten.

Da aber auch das reflektierte Licht den typischen Farbcharakter des jeweiligen Metalls hat, kann man eine Farbsegmentierung auch mit dem reflektierten Licht durchführen. Dazu muss die Belichtungszeit der Kamera entsprechend kurz eingestellt werden bzw. mit einer entsprechend großen Blende gearbeitet werden, damit das Bild nicht überstrahlt wird. Die Beleuchtung erfolgt damit mit einem Ringlicht mit Weißlicht-LEDs aus Richtung der Kamera.

11.2.2 Farbklassifikation

Bei der in Kapitel 4.3.1 vorgestellten Segmentierung mit Binarisierung war der erste Schritt, die Pixel in weiße und schwarze einzuteilen, also zu klassifizieren. Im zweiten Schritt, der eigentlichen Segmentierung, wurden die Pixel gleicher logischer Farbe zu Objekten zusammengefasst. Bei der Segmentierung im Farbbild ist das prinzipielle Vorgehen identisch: Im ersten Schritt wird jedes einzelne Pixel anhand seiner Farbe in eine Farbklaasse eingeteilt, also klassifiziert. Im zweiten Schritt werden benachbarte Pixel einer Farbklaasse zu Objekten zusammengefasst, was wir in Abschnitt 11.2.3 beschreiben.

Im vorherigen Beispiel haben wir bereits eine Farbklassifizierung vorgenommen. Dabei konnten wir durch die Mittelung über viele Pixel die Farbe sehr genau und stabil bestimmen, wodurch die Grenzen der zahlreichen Farbklassen sehr eng gefasst werden konnten. In der jetzt vorliegenden Anwendung hingegen liegen andere Rahmenbedingungen vor: Hier muss für jedes einzelne Pixel entschieden werden, ob es zu einem der drei Metalle, also Farbklassen, gehört oder zum Hintergrund. Die Farbwerte der einzelnen Pixel einer Farbklaasse unterscheiden sich aber deutlich, im Gegensatz zu den mittleren Farbwerten der Arbeitsbereiche aus

dem vorherigen Beispiel. Die Streuung der Farbwerte ist bei dem Draht teilweise auf die Krümmung und den damit veränderten Winkeln zwischen Beleuchtung und Oberfläche sowie Oberfläche und Kamera an den Rändern zurückzuführen. Die raue Struktur der Kontaktplatte sorgt ebenfalls für sehr unterschiedliche Farben, die alle zum Messing gehören. Aus diesem Grunde ist eine Klassifikation basierend auf weitgehend frei definierbaren Klassengrenzen erforderlich.

Für solche Klassifikationaufgaben haben wir in Kapitel 6.3 den Nächster-Nachbar-Klassifikator vorgestellt. Durch die Wahl der Referenzfarben – dies entspricht dem Prototypenmuster in der Nomenklatur von Kapitel 6.3 – lassen sich die Grenzen zwischen Klassen beliebig genau beschreiben, ohne dass eine analytische Beschreibung der Grenze existieren muss. Als Merkmalsvektor können direkt die RGB-Werte des Pixels verwendet werden. Der Wechsel in den HSI-Farbraum bringt in diesem Fall keinen Vorteil, da sich durch die gewählten Referenzfarben beliebige Klassengrenzen definieren lassen, unabhängig vom Farbraum. Im Gegenteil: Die für den Nächsten-Nachbar-Klassifikator erforderliche Abstandsrechnung zwischen zwei Merkmalsvektoren ist in einem Zylinderkoordinatensystem wie dem HSI-Farbraum aufwändiger als im kartesischen Koordinatensystem des RGB-Farbraums. Der Abstand d_j zwischen der Farbe eines Pixels i und einer Referenz p_j berechnet sich in Anlehnung an Gleichung 6.6 mit:

$$d_j = \sqrt{(i_R - p_{jR})^2 + (i_G - p_{jG})^2 + (i_B - p_{jB})^2} \quad (11.3)$$

Die in diesem Beispiel relevanten Farben – Zink, Messing und Kupfer – nehmen nur einen kleinen Teil des gesamten Farbraums ein. Der große Rest ist Hintergrund, also vor allem Gehäusekunststoff, Schatten, usw. In Kapitel 6.3 haben wir bereits darauf hingewiesen, dass sich die Klassen an den Rändern weit über den durch die Referenzen abgedeckten Bereich hinaus ausdehnen, was zu völlig willkürlichen Ergebnissen führen kann. Aufgrund dieser Problematik wurde der Nächster-Nachbar-Klassifikator um eine Rückweisungsschwelle erweitert, wie in Kapitel 6.3.2 dargestellt. Diese Erweiterung wird auch für die Farbklassifikation verwendet.

Die Bestimmung der Farbklasse eines Pixels beginnt damit, den Abstand zur nächstgelegenen Referenzfarbe gemäß Formel 11.3 zu berechnen. Wenn dieser Abstand kleiner ist als die Rückweisungsschwelle, wird das Pixel der Klasse zugeordnet, zu der die Referenzfarbe gehört. Wenn der Abstand größer als die Rückweisungsschwelle ist, wird das Pixel als Hintergrund klassifiziert.

Der Bereich des Farbraums, der zu einer Klasse gehört, wird somit durch die ausgewählten Referenzfarben und die Rückweisungsschwelle definiert. Dabei ist die Rückweisungsschwelle nur relevant für die Trennung zwischen den Farben und dem Hintergrund. Die Trennung der Farben untereinander wird ausschließlich von den Referenzfarben bestimmt. Eine kleine Rückweisungsschwelle lässt die Grenze zwischen der Farbe und dem Hintergrund sehr nah an den Referenzpunkten entlang verlaufen, wodurch der Grenzverlauf sehr genau und detailliert vorgegeben werden kann. Allerdings ist damit auch eine hohe Dichte der Referenzpunk-

te erforderlich, was zu vielen Musterfarben führt, wenn zur Klasse ein größerer Farbbereich gehört. Wie in Kapitel 6.3 ausgeführt, steigt bei Nächster-Nachbar-Klassifikatoren die Rechenzeit mit der Zahl der Prototypen an, weshalb für eine schnelle Klassifizierung möglichst wenige Referenzfarben verwendet werden sollten.

Eine hohe Rückweisungsschwelle erlaubt es, die Referenzfarben mit größerem Abstand untereinander zu definieren. Damit kann also ein größerer Bereich des Farbraums mit weniger Referenzfarben einer Farbe zugeordnet werden. Dafür können aber die Grenzen des Bereichs weniger stark beeinflusst werden. Die Auswahl der Referenzfarben und die Einstellung der Rückweisungsschwelle ist also ein Kompromiss zwischen einer stabilen Klassifizierung durch möglichst exakte Bereichsgrenzen und einer schnellen Klassifizierung durch möglichst wenige Referenzfarben. Die Stabilität hängt natürlich nicht nur von der Anzahl der Referenzfarben ab, sondern auch davon, welche ausgewählt werden. Man kann unter Umständen durch Optimierung der Referenzfarbenauswahl die Geschwindigkeit steigern, ohne die Stabilität zu beeinträchtigen, was allerdings gegebenenfalls längeres Experimentieren erfordert. In Abbildung 11.12 sieht man Segmentierungsresultate mit acht Referenzfarben für Zink, zwei für Kupfer und achtzehn für Messing. Die hohe Anzahl von Referenzfarben für Messing ist aufgrund der strukturierten Oberfläche erforderlich. Die linke Abbildung zeigt das geglättete Bild. In der Mitte wird eine für die Referenzfarbenauswahl gut geeignete Rückweisungsschwelle verwendet. In der rechten Abbildung wird eine höhere Rückweisungsschwelle mit denselben Referenzfarben verwendet, wodurch viele Pixel, die bei der niedrigeren Schwelle als Hintergrund klassifiziert werden, fälschlich als Objekte segmentiert werden.

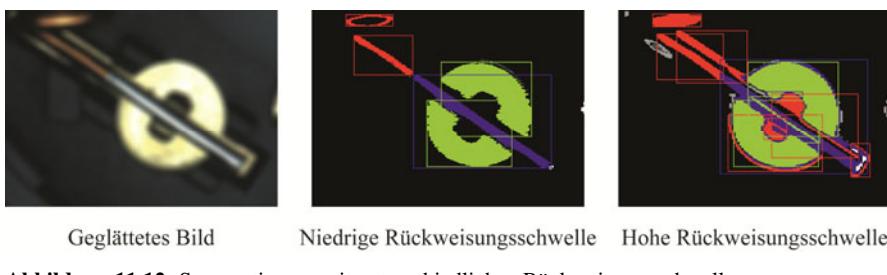


Abbildung 11.12: Segmentierung mit unterschiedlichen Rückweisungsschwellen.

In Abbildung 11.13 sind die Ergebnisse der Segmentierung mit den gleichen Rückweisungsschwellen wie in Abbildung 11.12 dargestellt. Allerdings wurde die Anzahl der Referenzfarben deutlich reduziert auf vier für Zink, eine für Kupfer und acht für Messing. Wie man in der mittleren Abbildung sehen kann, passt die niedrige Schwelle nicht zu den Referenzfarben, da große Teile der Objekte nicht segmentiert werden und Löcher in den Objekten entstehen, wohingegen die hohe Schwelle sehr gut zu dieser Auswahl passt, wie in der rechten Abbildung zu sehen ist. Gegenüber der vorherigen Abbildung wurde die Anzahl der Referenzfarben ungefähr halbiert, und damit ist wie erwartet auch die Rechenzeit nur noch halb so

groß. An den Abbildungen 11.12 und 11.13 kann man gut sehen, dass die Referenzfarbenauswahl und die Rückweisungsschwelle zusammenpassen müssen und nicht isoliert optimiert werden können.

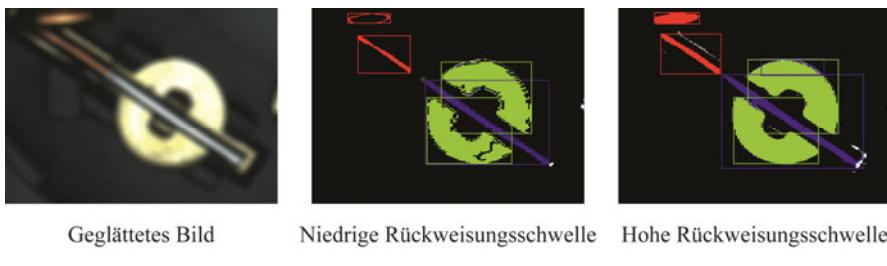


Abbildung 11.13: Segmentierung mit wenigen Referenzfarben

Wie bereits geschrieben, gibt es deutliche Farbunterschiede zwischen den einzelnen Pixeln einer Farbe. Dies lässt sich reduzieren, indem man das Bild glättet. Allerdings werden damit die Farbkanten ebenfalls verschmiert, so dass sich an den Grenzen Mischfarben bilden. Diese werden meistens als Hintergrund klassifiziert, so dass zwischen zwei Objekten unterschiedlicher Farbe bei starker Glättung ein schmaler Streifen als Hintergrund klassifiziert wird. Die Objekte werden also nicht bis zu den von uns klar erkennbaren Grenzen segmentiert, sondern sind etwas kleiner. Wenn die Objektgrenzen also sehr genau bestimmt werden sollen, muss man mit einer schwachen Glättung, einer kleinen Rückweisungsschwelle und vielen Referenzfarben klassifizieren. Abbildung 11.14 zeigt die Wirkung einer schwachen Glättung. Beide Bilder wurden mit der hohen Rückweisungsschwelle von Abbildung 11.12 und 11.13 segmentiert. In der Mitte wurde die Farbauswahl aus Abbildung 11.12 verwendet. Man sieht, dass im mittleren Bild von den Drähten nur ein schmaler Saum segmentiert wird. Durch die starke Glättung in den vorherigen Abbildungen und der geringen Breite des Drahts wurden die kräftigen Farben aus dem inneren Bereich abgeschwächt und deshalb auch nicht als Referenzfarben ausgewählt. Diese Farben müssen bei der schwachen Glättung als zusätzliche Referenzfarben definiert werden, wodurch für Zink und Kupfer erheblich mehr Muster nötig sind. Im rechten Bild von Abbildung 11.14 sieht man das Ergebnis einer Klassifikation mit einer speziell angepassten Farbauswahl mit siebzehn Zink-, sieben Kupfer- und achtzehn Messingmustern.

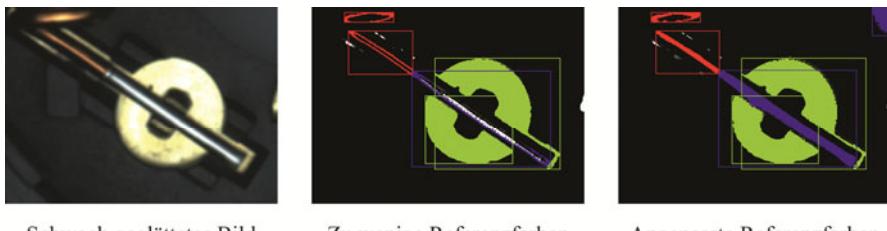


Abbildung 11.14: Segmentierung mit schwacher Glättung

11.2.3 Segmentierung

Bisher haben wir nur jeden einzelnen Pixel in Zink, Kupfer, Messing und Hintergrund klassifiziert, aber noch keine Objekte erzeugt. Der nächste Schritt ist daher die zusammenhängenden Pixel einer Klasse zu Objekten zusammenzufassen. Dies erfolgt genauso wie die in Kapitel 4.3.2 beschriebene Erzeugung von Objektkonturen. Der einzige Unterschied ist, dass in diesem Fall nicht zwischen Übergängen von Weiß nach Schwarz gesucht wird, sondern von der aktuellen Farbklasse zu allen anderen Farben.

Programmhinweis: In NeuroCheck wird die Farbklassifikation der Pixel und die Zusammenfassung zu Objekten im Prüfschritt *Color Matching* durchgeführt. Dieser Prüfschritt arbeitet nicht mit einer Rückweisungsschwelle, sondern mit einer Trennschärfe. Dabei entspricht einer kleinen Rückweisungsschwelle eine große Trennschärfe und umgekehrt. Wenn man eine große Ähnlichkeit zu den Referenzfarben fordert, muss man also eine hohe Trennschärfe einstellen. Insofern verhält sie sich entsprechend der Mindestgüte bei *Template Matching* und *Arbeitsbereiche klassifizieren*.

11.2.4 Anwesenheitsprüfung

Die eigentliche Anwesenheitsprüfung ist jetzt relativ einfach durchzuführen. Nach der Bildaufnahme wird der Suchbereich definiert. Da vorher keine Positionskorrektur stattfinden kann, muss der Arbeitsbereich großzügig definiert werden. Wie beschrieben wird dann das Bild geglättet. Da für die Anwesenheitskontrolle eine grobe Bestimmung der Objektkonturen völlig ausreicht, wird das Bild in diesem Fall relativ stark geglättet. Für die darauffolgende Segmentierung wird man entsprechend mit dem kleineren Referenzfarbensatz auskommen und eine hohe Rückweisungsschwelle verwenden, wie auf der rechten Seite von Abbildung 11.13 dargestellt.

Der letzte Schritt ist, zu prüfen, ob die gefundenen Objekte auch tatsächlich die gesuchten sind. Das verzinkte und das unverzinkte Drahtstück kann man gut an ihrem minimalen und maximalen Feret-Durchmesser erkennen, da es sich um lange, schmale Objekte handelt. Die beiden Kontakthälften kann man gut an ihrer Fläche erkennen. Aufgrund dieser Merkmale lassen sich die irrelevanten Objekte, wie beispielsweise die erste Wicklung der Spule aussortieren. An dieser Stelle muss nur noch geprüft werden, dass ein Kupfer, ein Zink und zwei Messing Objekte gefunden wurden.

Als abschließende Prüfung, ob wirklich die richtigen Objekte segmentiert wurden, kann man die Lage der Objekte zueinander prüfen. Durch die Glättung kann die Segmentierung der beiden Teile nicht bis ganz aneinander reichen, aber der Abstand zwischen ihnen sollte in der Größenordnung der Glättungslänge liegen. Ebenso berühren sich die segmentierten Kontakthälften und der Draht aufgrund der Glättung nicht, aber der Abstand darf auch hier höchstens in der Größenord-

nung der Glättungslänge liegen. Damit ist sichergestellt, dass die richtigen Objekte segmentiert wurden und das Spulendrahtende korrekt auf dem Kontakt liegt. Die gesamte Anwesenheitsprüfung ist in Tabelle 11.4 zusammengefasst.

Beispielbeschreibung 11.4. Die nachfolgende Tabelle fasst das Beispiel aus Abschnitt 11.2 zusammen.

Kategorie	Beschreibung
Verzeichnis	\Beispiele\Farbe
Beleuchtung	Auflichtbeleuchtung mit Weißlicht-LEDs
Prüfschritte	<p>Das Beispiel verwendet folgenden Prüfschrittablauf:</p> <ol style="list-style-type: none"> 1. <i>Bildaufnahme</i> 2. <i>Arbeitsbereiche definieren</i>, setzt einen Arbeitsbereich, in dem der Draht und der Kontakt gesucht wird. 3. <i>Bildbereiche glätten</i>, glättet das Bild, um die Farbnuancen zu reduzieren. 4. <i>Color Matching</i>, Klassifiziert die Pixel aufgrund ihrer Farbe und fasst sie zu Objekten zusammen. 5. <i>Arbeitsbereiche Merkmale erzeugen</i>, ermittelt die Fläche und den minimalen und maximalen Feret-Durchmesser der Objekte. 6. <i>Arbeitsbereiche filtern</i>, sortiert alle Objekte außer dem Kontakt und dem verzinkten und unverzinktem Draht aus. 7. <i>Arbeitsbereiche Anzahl überprüfen</i>, stellt sicher, dass alle vier Objekte gefunden wurden. 8. <i>Maße erzeugen</i>, ermittelt die geometrische Anordnung der Objekte zueinander. 9. <i>Maße Toleranzen überprüfen</i>, prüft ob die gefundenen Objekte richtig zueinander stehen, um Fehlsegmentierungen auszuschließen.

11.3 Zusammenfassung

Farbbildverarbeitung: Es gibt zwei typische Aufgabenstellungen in der Farbbildverarbeitung. Die Farbsegmentierung, bei der Objekte anhand ihrer Farbe gefunden und zusammengefasst werden und die Farbidentifikation, bei der bestehende Objekte in verschiedene Farbklasse eingeteilt werden.

Farbraum: Farträume bieten verschiedene quantitative Beschreibungen von Farbe. Viele Farträume basieren auf drei Kanälen, unterscheiden sich aber in der Bedeutung dieser Kanäle, im zugrundeliegenden Koordinatensystem und im Verwendungszweck.

Farbklassifizierung: Bei der Farbklassifizierung werden Arbeitsbereiche oder auch einzelne Pixel einer Farbklasse zugeordnet. Dies kann entweder über den Vergleich des Farbwertes mit vorgegebenen Grenzen erfolgen oder durch einen Klassifikator. Das erste Verfahren bietet sich an, wenn die Bestimmung der Farbe

sehr stabil ist, beispielsweise durch Mittelung über einen Arbeitsbereich. Das zweite Verfahren kann beispielsweise bei der Klassifikation einzelner Pixel angewendet werden.

RGB-Farbraum: Der Farbraum, in dem digitale Bilder häufig vorliegen und der in Monitoren und Fernsehern verwendet wird. Die einzelnen Komponenten repräsentieren den roten, grünen und blauen Anteil des Lichts.

HSI-Farbraum: An der menschlichen Beschreibung von Farbe orientierter Farbraum. Beschreibt eine Farbe durch ihren Farbton (englisch Hue), ihre Sättigung (englisch Saturation) und ihre Intensität (englisch Intensity) bzw. Helligkeit. Der HSI-Farbraum basiert auf einem Zylinderkoordinatensystem.

Template Matching: Man kann die Korrelation zwischen einem Template und einer Bildposition auch in einem Farbbild berechnen. Dadurch kann man die Objekte einer Farbe und einer bestimmten Form suchen. Gegenüber der Suche im Grauwertbild ist der Rechenaufwand wesentlich höher.

Farbsegmentierung: Die Farbsegmentierung erfolgt in zwei Schritten: Im ersten Schritt werden alle Pixel des Suchbereichs anhand ihrer Farbe klassifiziert. Im zweiten Schritt werden zusammenhängende Pixel gleicher Farbklasse zu Objekten zusammengefasst.

12. Realisierung von Bildverarbeitungsapplikationen

Wir haben in den vorangegangenen Kapiteln eine Vielzahl von Methoden und Algorithmen der Bildverarbeitung und eine Reihe beispielhafter Anwendungen aus der industriellen Praxis beschrieben. Zum Abschluss wollen wir auf einige Aspekte eingehen, die für die Realisierung von Bildverarbeitungsapplikationen von Bedeutung sind.

12.1 Bildverarbeitungsprojekte

Zunächst wollen wir einige Gesichtspunkte ansprechen, auf die man bei der Durchführung einer industriellen Bildverarbeitungsapplikation achten sollte. Manches entspricht allgemeinen Regeln industriellen Projektmanagements, aber es gibt auch einige für die Bildverarbeitung spezifische Aspekte.

Pflichtenheft. Ein Bildverarbeitungssystem in der Fertigungs- oder Qualitätskontrolle ist nichts anderes als eine besondere Maschine. Wie bei jeder Maschine muss auch hier die Konzeption mit der detaillierten Spezifikation der Fähigkeiten dieser Maschine in einem Lastenheft beginnen. Bei der Konstruktion „rein materieller“ Maschinen ist dieser Schritt ganz selbstverständlich. Der hohe Softwareanteil von Bildverarbeitungsanwendungen verführt häufig dazu, die sorgfältige Spezifikation zu vernachlässigen. Bei einem so flexiblen und vielseitigen Gebilde wie Software ist jedoch die Möglichkeit von Missverständnissen in besonderem Maße gegeben. Man kann sehr viele Dinge mit Software verwirklichen, aber man muss sich dann auch frühzeitig darüber klar sein, was man erreichen will.

Die Erstellung dieses Lastenheftes sollte immer vom Auftraggeber vorgenommen werden. Natürlich ist dieser bis zu einem gewissen Grad auf die Mitwirkung des Spezialisten angewiesen, der an manchen Stellen sagen muss, was nicht durchführbar ist, oder umgekehrt, was noch möglich wäre. Es ist aber sehr wichtig, dass sich der Auftraggeber selbst genau darüber im Klaren ist, welche Eigenschaften eines Werkstücks geprüft werden sollen, welche Toleranzen eingehalten werden müssen und in welcher Umgebung die Prüfung stattzufinden hat. Scheinbar triviale Dinge wie der Bewegungsraum für einen Roboterarm, die mögliche Einbauhöhe für Beleuchtung und Kamera und dergleichen können entscheidenden Einfluss auf die endgültige Realisierung haben. Das gleiche gilt für Taktzeiten, die entscheidend für die Wahl des Rechnersystems sind, Kommunikationsmöglichkeiten

ten für die Einbindung in den Fertigungsprozess usw. Solche Fragen müssen darum so frühzeitig wie möglich geklärt werden.

Musterteile. Wenn die grundsätzlichen Anforderungen an das System spezifiziert sind, kann man daran gehen, ein Lösungskonzept zu erarbeiten. In dieser Phase benötigt man Musterteile in ausreichender Anzahl, die die Produktionsschwankungen widerspiegeln müssen. Das heißt, diese Auswahl sollte eindeutig gute und fehlerhafte Teile ebenso enthalten wie Grenzmuster, also Teile, die gerade noch gut oder gerade schon fehlerhaft sind. Es ist empfehlenswert, Bilder aller verwendeten Musterteile in die Dokumentation des erarbeiteten Lösungskonzeptes aufzunehmen, und zwar auch in digitaler Form, d. h. in einem Format, das auch vom verwendeten Bildverarbeitungssystem verarbeitet werden kann. Dadurch kann man die Lösung auch dann noch demonstrieren, wenn die Musterteile nicht mehr zur Verfügung stehen oder aufgrund von Alterungsprozessen, Korrosion usw. den Spezifikationen der Aufgabe nicht mehr entsprechen. Auch für den Fall nachträglicher Spezifikationsänderungen ist es vorteilhaft, anhand dieser Bilder die Erfüllung des ursprünglichen Pflichtenheftes nachweisen zu können.

In manchen Fällen ist es nicht möglich, eine entsprechende Teilesammlung zu erhalten, z. B. weil die Fertigungsanlage noch gar nicht in Betrieb ist. Dann muss man versuchen, aufgrund verfügbarer Informationen, z.B. Vorserienteilen oder Labormustern, und eigener Erfahrung ein Konzept zu entwickeln, das hinreichend Spielraum für Anpassungen lässt. Das erfordert auf Seiten des eingesetzten Bildverarbeitungssystems die Möglichkeit, vor Ort auf veränderte Bedingungen zu reagieren und entsprechende Umkonfigurationen ohne großen Aufwand vorzunehmen.

Beleuchtung und Bildaufnahme. Der erste Schritt beim Erarbeiten des Lösungskonzeptes ist stets die Entwicklung einer optimalen Beleuchtungskonfiguration. Man muss sich immer vor Augen halten, dass der Mensch Sichtprüfaufgaben unter Einsatz seines gesamten Wissens und eines visuellen Apparates, der sich über Jahrtausende an die Gegebenheiten der realen Welt anpassen konnte, lösen kann. Ein digitales Bildverarbeitungssystem ist dagegen darauf angewiesen, dass die relevante Information in geeigneter Form durch die Beleuchtung herausgearbeitet wird. Information, die an dieser Stelle nicht erzeugt wird, lässt sich ebenso wenig wiedergewinnen, wie Information, die im Verlauf der Bildaufnahme und -Übertragung verloren geht. Aufbau und Qualität von Beleuchtung und bildgebenden Komponenten sind daher von entscheidender Bedeutung für den erfolgreichen Einsatz eines Bildverarbeitungssystems.

Integration in die Fertigung. Die Einbindung der fertigen Bildverarbeitungslösung in die Fertigung erfordert wie bei jeder automatischen Anlage mechanische, elektrische und Informationsschnittstellen. Genau wie die eigentliche Prüfaufgabe müssen auch diese vorher vereinbart werden. Diese Fragestellungen haben auch Rückwirkungen auf den Entwurf der Bildverarbeitungslösung selbst und die Auswahl von Beleuchtung und Bildaufnahmeverfahren. Wie bei jeder Maschine sieht man sich auch hier einem iterativen Optimierungsprozess gegenüber.

Im mechanischen Bereich stellen sich Fragen wie: Ist eine eigene Zuführung oder Vereinzelung notwendig, sollen die Teile von Industrierobotern gehandhabt werden, müssen die Teile während der Bildaufnahme angehalten werden oder ist es erforderlich, die Teile in der Bewegung aufzunehmen ...

Eine sehr wichtige Frage ist auch die des Informationsflusses zwischen Fertigungsprozess und Bildverarbeitungssystem. Aufgrund der großen Bandbreite an Technologien und der Bedeutung für die Bildverarbeitungsapplikation wollen wir darauf in Abschnitt 12.2 genauer eingehen.

Für die Zeit nach der Inbetriebnahme des Bildverarbeitungssystems müssen Vorschriften zur Pflege, Wartung und Ersatzteilhaltung festgelegt werden. Bildverarbeitungssysteme sind automatische Anlagen von hoher Zuverlässigkeit, vorausgesetzt, die Basis ihrer Arbeit bleibt stabil, d. h. die Qualität des aufgenommenen Bildes bewegt sich innerhalb zulässiger Toleranzen. Dazu muss man einwandfreies Funktionieren aller Komponenten sicherstellen. Wenn aufgrund der Fertigungsumgebung mit Verschmutzungen zu rechnen ist, dann müssen Reinigungsintervalle festgelegt werden. In rauer Umgebung sollten alle Komponenten staub- und wasserdicht gekapselt sein, so dass auch bei Reinigungs- und Wartungsarbeiten keine Beschädigung oder Dejustierung von Lampen, Objektiven, Kameras usw. zu befürchten sind.

12.2 Prozessintegration

Für den Einsatz der industriellen Bildverarbeitung ist es von großer Bedeutung mit der Fertigungssteuerung Daten auszutauschen. So muss die Steuerung die Bildverarbeitungsstation als Teil der Fertigungsprozess kontrollieren können und umgekehrt müssen die mittels Bildverarbeitung gewonnen Ergebnisse an die Steuerung gemeldet werden. Für diese Datenübertragung werden eine Reihe verschiedener Technologien verwendet. Die wichtigsten Technologien wollen wir kurz vorstellen:

Digital-I/O: Die Übertragung der Informationen erfolgt über einzelne Signalleitungen, deren Spannungspiegel ein Bit repräsentiert. Der Vorteil dieser Technologie ist die minimale Latenzzeit. Nachteilig ist die begrenzte Bandbreite, so können häufig nur 16 oder 32 Bit übertragen werden. Da die Kabel für jedes Bit eine eigene Ader benötigen sind sie verhältnismäßig teuer und empfindlich. Eine Besonderheit von Digital-I/Os ist die Möglichkeit neben der Informationsübertragung auch einzelne Beleuchtungselemente vom Bildverarbeitungsrechner aus zu schalten. Um von PCs aus auf eine Digital-I/O-Schnittstelle zugreifen zu können, gibt es spezielle Einsteckkarten.

Feldbus: Aus Anwendersicht erscheint ein Feldbus zustandsbasiert wie eine Digital-I/O. Tatsächlich werden die Informationen aber seriell über ein Netzwerk ausgetauscht. Diese Nachrichten werden zyklisch im Takt ausgetauscht, so dass die Informationen im Allgemeinen nur mit Verzögerung aktualisiert werden. Der Vor-

teil von Feldbus zu Digital-I/O ist, dass bei jedem Datenaustausch Pakete von mehreren 100 bis 1000 Bytes problemlos ausgetauscht werden können. Außerdem ist die Verkabelung deutlich günstiger.

Für den Einsatz in der industriellen Fertigung haben sich eine Reihe verschiedener Protokolle etabliert, wie beispielsweise PROFIBUS, CAN Bus, Interbus-S, Foundation Fieldbus oder DeviceNet. Um einen PC in ein Feldbus-Netzwerk zu integrieren, muss in den Rechner eine entsprechende Einsteckkarte eingebaut werden.

Industrial Ethernet: Während die Feldbusse der ersten Generation die Daten über serielle Schnittstellen austauschten, verwenden die Feldbusse der zweiten Generation Ethernet als Übertragungsmedium. Deshalb werden diese Technologien oft auch als Industrial Ethernet bezeichnet, wobei die Verwendung dieses Begriffs nicht immer eindeutig ist.

Gegenüber serieller Datenübertragung bietet Ethernet die Möglichkeit deutlich mehr Daten zu übertragen. Für die Verwendung als Feldbus-Medium hat Ethernet allerdings den Nachteil, dass seine Übertragungszeiten nicht sehr deterministisch sind. Die verschiedenen Industrial-Ethernet-Protokolle wie beispielsweise PROFINET, Ethernet/IP, Ethernet Powerlink oder EtherCAT erweitern deshalb oft den Ethernet-Standard um eine ausreichende Echtzeitfähigkeit zu erreichen. Diese Erweiterungen müssen teilweise in Hardware implementiert werden, was bei PCs trotz integrierter Standard-Ethernet-Schnittstelle den Einbau einer Einsteckkarte erforderlich macht.

Während sich die verschiedenen Feldbus-Protokolle seit Jahren etabliert haben, ist der Markt bei Industrial Ethernet zur Zeit noch sehr stark im Fluss. Man kann allerdings erwarten, dass sich im Laufe der Zeit eine begrenzte Anzahl von Protokollen am Markt durchsetzen wird.

Serielle Schnittstelle: Als serielle Schnittstelle wird im Allgemeinen die Schnittstelle RS-232 bezeichnet, die früher häufig für die Kommunikation zwischen einem PC und seiner Peripherie verwendet wurde. In den letzten Jahren wurde sie vor allem von USB verdrängt. So gibt es heute kaum noch Rechner, die standardmäßig über eine serielle Schnittstelle verfügen.

Die Vorteile der seriellen Schnittstelle waren vor allem die geringen Kosten und die für frühere Verhältnisse große Datenmenge. In der industriellen Kommunikation wird die serielle Kommunikation immer mehr von Ethernet verdrängt, das standardmäßig in fast jedem PC eingebaut ist.

Für die ausgetauschten Daten gibt es kein einheitliches Format, stattdessen wird es für jede Anlage zwischen dem Programmierer der Steuerung und dem Entwickler der Bildverarbeitungsanwendung individuell vereinbart.

Netzwerk: Der de-facto-Standard bei den Netzwerktechnologien ist Ethernet. Da beinahe jeder neue PC mit einer Ethernet-Schnittstelle ausgestattet ist, ist der Datenaustausch mit Ethernet in der Automatisierungstechnik heute sehr verbreitet. Dabei werden im Gegensatz zu Industrial Ethernet oft weitere Protokolle aus dem Bereich der sogenannten Internet-Technologien verwendet wie z. B. die bekannten

IP und TCP. Ethernet zeichnet sich vor allem durch die großen Datenmengen aus die damit übertragen werden können. Allerdings kann es bei Ethernet zu längeren Verzögerungen der Übertragung kommen, besonders wenn mehrere Teilnehmer in einem Netzwerk gleichzeitig senden.

Dateien: Bei Dateien handelt es sich nicht um eine Übertragungstechnologie. Tatsächlich erfolgt der Austausch von Dateien zwischen dem Bildverarbeitungsrechner und der Fertigungssteuerung letztendlich über ein Ethernet-Netzwerk. Mit der Klassifikation als Übertragungstechnologie folgen wir an dieser Stelle (Hornberg 2006). Denn Dateien sind nach wie vor wichtiges Medium, um vor allem größere Daten zusammenzufassen und auszutauschen. Da der Austausch von Dateien zwischen zwei Netzwerk-Teilnehmern über Standardfunktionalität des Betriebssystem erfolgt, ist dieser Weg oft sehr einfach zu realisieren.

Die Datenformate von Dateien sind natürlich so vielfältig wie ihr Inhalt. Während für die Speicherung von Bildern beispielsweise häufig Standardformate wie TIFF, BMP, PNG usw. verwendet werden, kommen bei Dateien mit Messwerten und Leseergebnissen häufig proprietäre Formate zum Einsatz. Bei der Speicherung von alphanumerischen Daten gibt es einen deutlichen Trend zu XML-Formaten, da diese sowohl für Computer als auch für Menschen gut lesbar sind. Zu den Dateien gehören auch Datenbanken, in den Ergebnisse oder prüftypspezifische Parameter gespeichert werden können.

Neben dem Übertragungsmedium kann man den Informationsaustausch auch anhand der Art der Informationen klassifizieren. (Hornberg 2006) unterscheidet nach Umfang und Verwendungszweck der Informationen drei Informationsklassen:

- Steuersignale
- Ergebnisse und Parameter
- Massendaten

Neben dem Inhalt unterscheiden sich die verschiedenen Klassen im Umfang und in den für die Übertragung verwendeten Technologien.

Steuersignale: Das Bildverarbeitungssystem muss mit dem Fertigungsprozess synchronisiert werden, dazu werden sogenannte Steuersignale zwischen der Prozesssteuerung und dem Bildverarbeitungsrechner ausgetauscht. Dabei werden Kommandos von der Steuerung an den Bildverarbeitungsrechner gesendet. Aus Sicht des Bildverarbeitungsrechners handelt es sich dabei um Eingangssignale. Typische Eingangssignale sind:

- Prüfung starten, weist das Bildverarbeitungssystem an die nächste Prüfung auszuführen.
- Prüftyp wechseln, wechselt die Prüfung da z. B. ein Prüfteil eines andern Typs geprüft werden muss.
- Prüfung abbrechen, bricht die Ausführung der aktuellen Prüfung ab.

- Benutzer wechseln, wechselt zwischen Benutzer verschiedener Rechte, damit bestimmte Änderungen an der Konfiguration nur von entsprechend geschultem Personal vorgenommen werden können.

Der Bildverarbeitungsrechner meldet umgekehrt seinen Status und den Status der Prüfung an die Steuerung des Fertigungsprozess. Für den Bildverarbeitungsrechner sind das Ausgangssignale, typische Beispiele dafür sind:

- System bereit, das Bildverarbeitungssystem wartet auf das nächste Kommando
- System läuft, das Bildverarbeitungssystem führt aktuell ein Kommando aus. D. h. im Allgemeinen wird gerade eine Prüfung durchgeführt.
- Prüfung i. O., das letzte geprüfte Teil war in Ordnung (abgekürzt i. O.)
- Prüfung n. i. O., das letzte geprüfte Teil war nicht in Ordnung (abgekürzt n. i. O.)

Die Meldungen Prüfung i. O. und Prüfung n. i. O. werden häufig beide verwendet, um festzustellen, wann das Ergebnis des Bildverarbeitungssystems vorliegt

Da die Steuerungssignale der zeitlichen Synchronisation von Bildverarbeitungssystem und Fertigungsprozess dienen, müssen sie im Allgemeinen mit einer sehr geringen Latenzzeit erfolgen. Aus diesem Grund werden häufig Feldbus, Digital-I/O oder Industrial Ethernet als Übertragungstechnologie verwendet. Bei langsameren Systemen können aber auch Netzwerke oder die serielle Schnittstelle verwendet werden. Die mit Steuersignalen übertragenen Datenmengen sind meistens gering, so lassen sich viele durch einzelne Bits übertragen, was die Verwendung von Digital-I/Os erst ermöglicht.

Ergebnisse und Parameter: In vielen Fällen reicht es nicht aus, dass das Bildverarbeitungssystem nur das Endergebnis der Prüfung an den Steuerungsrechner sendet. Häufig müssen darüber hinaus weitere Ergebnisse wie Messwerte, Leseergebnisse, Objektpositionen usw. an die Steuerung übermittelt werden. Diese Daten können für sehr unterschiedliche Zwecke verwendet werden, typische Beispiele sind die Langzeitarchivierung oder die statistische Auswertung.

Bei vielen Produkten gibt es kleine Änderungen zwischen verschiedenen Chargen, so kann sich beispielsweise die Farbe, eine Typnummer, einzelne Maße oder die Position einzelner Prüfmerkmale auf dem Prüfobjekt ändern. Wenn diese Änderungen klein sind, lohnt es sich nicht für jede Variante eine eigene Prüfung zu entwickeln, sondern es ist günstiger die einzelnen Parameter der Prüfung zu verändern. Dazu müssen die relevanten Parameter von der Steuerung an den Bildverarbeitungsrechner übertragen werden.

Die Datenmengen von Ergebnissen und Parametern beginnen bei einigen Bytes und können speziell bei komplexen Parametrierungen beliebig groß werden. Je nach Umfang der Datenmengen werden verschiedene Übertragungstechnologien verwendet. Bei kleinen Datenmengen ist ein Feldbus oder Industrial Ethernet ausreichend, größere Datenmengen können per Ethernet übertragen werden, während der Austausch sehr großer Datenmengen überwiegend über Datei erfolgt. Die Bandbreite einer Digital-I/O ist im Allgemeinen zu gering für die Übertragung von Ergebnissen oder Parametern.

Massendaten: Als Massendaten sind bei Bildverarbeitungssystemen Bilder gemeint. Bei manchen Bildverarbeitungsstationen müssen sämtliche aufgenommenen Bilder für eine Langzeitarchivierung gespeichert werden. Damit ist es möglich bei Reklamationen zu belegen, dass das Prüfteil die Fertigungsstufe, bei der die Prüfung stattgefunden hat, in einwandfreien Zustand verlassen hat. Bei anderen Anlagen werden beispielsweise die Bilder von Schlechtteilen an einen Nacharbeitsplatz übertragen. Anhand des Bildes kann der Werker erkennen, welches Merkmal nicht in Ordnung war und gegebenenfalls den Fehler beheben. In derartigen Fällen können in die Bilder künstlich weitere Informationen eingeblendet werden.

Da die Datenmengen gerade bei Bildern sehr oft im Bereich von einem oder mehreren Megabytes liegt, kommen als Übertragungsmedien nur Dateien oder Netzwerk in Frage.

12.3 Ausblick

In den letzten Jahren ist ein Trend zu immer anspruchsvollerem Bildverarbeitungsaufgaben zu beobachten. Die Herausforderungen liegen dabei nicht nur in der eigentlichen Bildauswertung. Häufig ist eine umfassende, anschauliche Visualisierung der Prüfergebnisse und Prozessdaten erwünscht. Außerdem ist der Austausch einer Vielzahl von Daten mit der übergeordneten Steuerung erforderlich. Um all diesen Anforderungen gerecht zu werden, benötigen Bildverarbeitungssysteme eine flexible und leistungsfähige Architektur. PC-Systeme mit modernen multithreading-fähigen Betriebssystemen wie Windows® Vista® oder Windows® 7 können aufgrund ihrer Flexibilität und Erweiterbarkeit schnell an vielfältige Anforderungen angepasst werden können. Außerdem stehen für PCs neue Technologien oft sehr frühzeitig zur Verfügung. Von modernen Multi-Core-Prozessoren beispielsweise profitiert gerade die Bildverarbeitung überdurchschnittlich, da sich viele ihrer Algorithmen durch eine geeignete Implementierung auf viele Prozessorkerne verteilen lassen. Der bevorstehende Wechsel zur 64-bit-Technologie wird speziell für speicherintensive Zeilenkamera-Anwendungen einen weiteren Leistungsschub bringen.

Die Verwendung von PC-Hardware und -Betriebssystemen erleichtert auch die Pflege und Wartung durch den Rückgriff auf bewährte Standardkomponenten. Die Standardisierung vereinfacht zudem die Bedienung und Konfiguration der Systeme durch das Bedienungspersonal vor Ort, da auf Kenntnisse aus der täglichen Arbeit mit PCs aufgebaut werden kann. Insbesondere bei häufig wechselnden Anforderungen kann die Fähigkeit des Bedienpersonals, das Bildverarbeitungssystem durch Umkonfiguration anzupassen, entscheidend für die Leistungsfähigkeit sein.

A. Mathematische Ergänzungen

A.1 Backpropagation-Training

In diesem Abschnitt beschreiben wir das in Abschnitt 6.4.2 angesprochene Backpropagation-Trainingsverfahren für neuronale Netze vom Multilayer-Perzeptron-Typ. Zuvor möchten wir noch einige Bemerkungen zur Geschichte dieses Netztyps machen.

A.1.1 Neuronale Netze – Begriff und Geschichte

Neuronales Netz, konnektionistisches System, parallele verteilte Verarbeitung – alle diese Begriffe bezeichnen Systeme, die versuchen, bestimmte Leistungen in der Verarbeitung von Informationen mittels einer Vielzahl von Verarbeitungseinheiten zu erreichen, die für sich genommen sehr einfach sind. Die Leistungsfähigkeit des Systems ergibt sich aus der Komplexität der Verbindungsstruktur zwischen diesen Verarbeitungseinheiten.

Das leistungsfähigste bekannte neuronale Netz ist das menschliche Gehirn. Funktionell – biologisch – betrachtet, ist ein einzelnes Neuron ein relativ einfaches Gebilde, wenn man es z. B. mit einem modernen Computerprozessor vergleicht. Dennoch vollbringt das Gehirn Leistungen, die bislang keinem Computer möglich sind, vor allem im Bereich der Mustererkennung.

Demgegenüber sind bislang technisch realisierte Systeme dieser Art sehr bescheiden, sowohl in ihrem Umfang als auch in ihren Leistungen. Dennoch versprechen sie auf bestimmten Gebieten ein erhebliches Leistungspotential. Zur Unterscheidung von den natürlich entstandenen, biologischen neuronalen Netzen – wie eben dem Gehirn – ist es in der englischen Literatur üblich, sie als *Artificial Neural Nets* (abgekürzt ANN), künstliche neuronale Netze, zu bezeichnen.

Ein erster Meilenstein in der Entwicklung künstlicher neuronaler Netze war die formale Beschreibung eines Neurons durch McCulloch und Pitts in (McCulloch und Pitts 1943). Die Beschreibung eines Neurons als eines einfachen Schwellwertelements war zwar sehr primitiv, enthielt aber schon eine entscheidende Eigenschaft, die Nichtlinearität. Das Potential von Neuronenkomplexen zur Informationsverarbeitung ließ sich damit aufzeigen.

Noch fehlte aber ein sehr wesentliches Element, das natürliche neuronale Netze von künstlichen unterscheidet, die Lernfähigkeit. In (Hebb 1949) wurde erstmals

eine Methode vorgestellt, mit der eine Adoptionsfähigkeit künstlicher neuronaler Netze erreicht werden konnte. Diese Lernregel findet noch heute in verschiedenen Varianten Verwendung.

In (Rosenblatt 1958) wurde dann mit dem Perzeptron ein erstes abgeschlossenes Modell vorgestellt. In der Folgezeit erlahmte das Interesse an neuronalen Netzen, da sich zeigte, dass die ursprünglich gesetzten Ziele nicht so leicht zu erreichen waren. Erst in den 80er Jahren gab es viel beachtete Fortschritte, darunter die Arbeiten von John Hopfield über autoassoziative Netze (Hopfield 1982), Teuvo Kohonen über selbstorganisierende Netzwerke (Kohonen 1982) und v. a. die Entwicklung der Generalisierten Delta-Lernregel durch Rumelhart und Hinton, (Rumelhart und McClelland 1986).

Neben diesen neuen Entdeckungen waren es aber insbesondere die Fortschritte in der Datenverarbeitungstechnologie, die das Interesse an künstlichen neuronalen Netzen wiedererwachen ließen. Erst die gestiegenen Rechenleistungen und Speicherkapazitäten ermöglichten die Untersuchung von Netzen, die reale Anwendungsprobleme bearbeiten konnten.

Als positive Eigenschaften neuronaler Netze für industrielle Anwendungen gelten:

- **Robustheit:** Neuronale Netze reagieren auf gestörte Eingangssignale nicht abrupt mit Fehlern, sondern zeigen ein schrittweises Abnehmen ihrer Leistungen (sogenannte *graceful degradation*). Das macht sie zum Beispiel für Mustererkennungsanwendungen geeignet, in denen mit Störungen der Muster zu rechnen ist. In Hardware realisierte neuronale Netze können auch eine Robustheit gegenüber dem Ausfall einzelner Bauelemente zeigen, da die Information in der Struktur des Netzes verteilt gespeichert ist.
- **Parallelisierung:** Grundsätzlich können alle Verarbeitungseinheiten eines Netzes parallel arbeiten.
- **Adaptivität:** Hiermit wird die Fähigkeit neuronaler Netze bezeichnet, ihren inneren Zustand anhand von Trainingsbeispielen so zu verändern, dass sie bestimmte Aufgaben lösen können, ohne dass ihnen der Lösungsweg explizit vorgeschrieben werden muss. Oft spricht man dabei von „Lernen“, allerdings sind die dafür verwendeten Methoden von biologischem Lernen noch weit entfernt.

A.1.2 Grundlagen

In diesem Abschnitt wiederholen wir in aller Kürze die Grundlagen zur Funktionsweise des Multilayer-Perzeptron-Netzes aus 6.4.2, um die für das Trainingsverfahren benötigten Informationen an einer Stelle zusammenzuführen. Der typische Aufbau eines Multilayer-Perzeptrons ist in Abbildung 6.5 gezeigt. Es besteht typischerweise aus drei Schichten von Verarbeitungseinheiten, einer Eingangs-

schicht, einer sogenannten verborgenen Schicht und einer Ausgangs- oder Ausgabeschicht. Dazwischen befinden sich zwei Lagen von Verbindungen.

Der Aktivierungszustand einer Einheit j ist eine Funktion der gewichteten Summe aller Ausgänge o_k der Einheiten k , die über gewichtete Verbindungen w_{kj} mit Einheit j verbunden sind:

$$a_j = f_a \left(\sum_k w_{kj} o_k - \theta_j \right). \quad (\text{A.1})$$

Als Aktivierungsfunktion wird zumeist die *logistische Funktion* verwendet:

$$f_a(\text{net}_j) = \frac{1}{1 + e^{-(\text{net}_j - \theta)}}. \quad (\text{A.2})$$

Ihre Vorteile liegen einmal in einer Rauschunterdrückung durch das asymptotische Verhalten, zum anderen in ihrer sehr einfachen Ableitung:

$$f(x) = \frac{1}{1 + e^{-x}}, \quad (\text{A.3})$$

$$\frac{df}{dx} = \frac{e^{-x}}{(1 + e^{-x})^2}, \quad (\text{A.4})$$

$$\frac{df}{dx} = f(x)(1 - f(x)). \quad (\text{A.5})$$

In praktischen Implementationen wird wegen des Zeitaufwandes für die Berechnung der Exponentialfunktion die logistische Funktion so gut wie immer tabelliert. Die einfache Form der Ableitung bietet dann den Vorteil, mit einer einzigen Look-Up-Tabelle gleichzeitig sehr schnell die Werte der für das Trainingsverfahren erforderlichen Ableitung berechnen zu können.

A.1.3 Backpropagation

Backpropagation ist eine allgemeine Methode für kontinuierliche Optimierungsaufgaben vom Typ der Gradientenabstiegsverfahren. Es hat enorme Bekanntheit erlangt als Lernverfahren für mehrschichtige Netze vom Perzeptron-Typ, d. h. geschichtete Netze, wie sie in Abbildung 6.7 gezeigt sind. Streng genommen müsste man von *Generalisierter Delta-Regel* sprechen, aber der Begriff Backpropagation ist allgemein verbreitet. Der Backpropagation-Algorithmus soll hier nicht formal hergeleitet, sondern nur seine Grundidee dargestellt werden. Die Herleitung ist in den meisten Büchern über neuronale Netze zu finden, so z. B. in (Zell 1994).

Das Trainieren eines einschichtigen Perzeptron-Netzes erfolgt überwacht. Ein Eingangsmuster, zu dem ein Soll-Output bekannt ist, wird an das Netz angelegt. Der Zustand der Ausgabeschicht, der sich daraufhin einstellt, wird mit dem Soll-Output verglichen. Wenn Abweichungen vorhanden sind, werden die Gewichte des Netzes entsprechend angepasst. Diese Regel ist leider nicht ohne weiteres auf mehrschichtige Netze zu übertragen, denn für die Zwischenschichten ist ja überhaupt kein Sollzustand bekannt. Die Lösung wurde in Form der *Generalisierten Delta-Regel mit Fehlerrückvermittlung* gefunden, die eben allgemein mit dem englischen Namen *Backpropagation* bezeichnet wird, (Rumelhart und McClelland 1986).

Ziel des Algorithmus ist, die Summe der Fehler zu minimieren, die bei der Klassifikation aller Trainingsmuster entstehen. Der Fehler bei der Klassifikation eines Einzelmusters wird definiert als

$$F_m = \frac{1}{2} \sum_j (t_j - o_j)^2 . \quad (\text{A.6})$$

Darin ist m die Nummer des jeweiligen Trainingsmusters; t_j ist der Sollausgangswert für die j -te Ausgabeeinheit, o_j ist der tatsächliche Ausgangswert¹. Gesucht ist das globale Minimum der Gesamtfehlerfunktion

$$F = \sum_m F_m .$$

Zu diesem Zweck werden die Gewichte stets in Richtung abnehmenden Fehlers verändert, d. h. entgegen dem Gradienten des Fehlers:

$$\Delta w_{ij} \propto -\frac{\partial F}{\partial w_{ij}} \quad (\text{A.7})$$

Dies ist die ursprüngliche *Delta-Regel*. Die wesentliche Idee des Backpropagation-Verfahrens besteht nun darin, dass der Ausgangswert der Ausgabeeinheiten – und damit der Fehler F_m – eine Funktion ihrer Eingangswerte ist, diese wiederum eine Funktion der Ausgänge der vorhergehenden – verdeckten – Schicht des Netzes und der Verbindungsgewichte zwischen diesen beiden Schichten. Die Ausgänge der Zwischenschicht aber sind eine Funktion der Gewichte zwischen Eingangs- und Zwischenschicht, und damit ist letztlich der Fehler F_m auch eine Funktion dieser Gewichte. Hiervon ausgehend ist es möglich, durch partielle Differenzieren nach der Kettenregel auch für die Zwischenschicht einen Fehler zu de-

¹ Genau genommen müssten diese Werte ebenfalls noch den Index m für die Nummer des Eingangsmusters tragen. Dieser Index ist im Folgenden stets weggelassen, da alle folgenden Gleichungen nur mit dem Einzelfehler zu tun haben.

finieren, aus dem ein Gradient zur Veränderung der Gewichte bestimmt werden kann.

Die Formeln für die Anpassung des Gewichtes w_{ij} zwischen einer verborgenen Einheit i und einer Ausgabeeinheit j des Netzes lautet mit der Lernrate η der Ableitung der Aktivierungsfunktion f'_a und dem Ausgangswert o_i der verborgenen Einheit:

$$\Delta w_{ij} = \eta \delta_j f'_a(\text{net}_j) o_i . \quad (\text{A.8})$$

Darin ist der Fehlerwert δ_j :

$$\delta_j = -\frac{\partial F}{\partial \text{net}_j} = (t_j - o_j) f'_a(\text{net}_j) . \quad (\text{A.9})$$

Für die verdeckten Einheiten wird ein entsprechender Fehlerwert δ_i aus dem Fehlerwert der Ausgangseinheiten definiert:

$$\delta_i = f'_a(\text{net}_i) \sum_j \delta_j w_{ij} . \quad (\text{A.10})$$

Die Gewichtsadaption erfolgt wieder nach Gleichung (A.8).

Der Fehlerwert wird also zuerst für die Ausgangseinheiten berechnet, dann für die verdeckten Einheiten. Daher kommt auch der Name *Error-Backpropagation*, da sich der Fehler von hinten her durch das Netz fortpflanzt.

Trainingsablauf bei Backpropagation: Die oben vorgestellte Form des Backpropagation-Algorithmus versucht, den Gesamtfehler auf der gesamten Trainingsmenge zu minimieren, d. h. es werden immer zunächst alle Muster einmal klassifiziert, ehe die Fehlerwerte berechnet und die Gewichte geändert werden. Dies bezeichnet man als *Batch-Backpropagation*.

In der Praxis betrachtet man meist nicht die Ableitung des Gesamtfchers, sondern lernt sequentiell, d. h. man nimmt eine Veränderung der Gewichte nach jedem Trainingsmuster vor. Dieses Verfahren wird als *Standard-* oder *Online-Backpropagation* bezeichnet. Das Training läuft dann folgendermaßen ab:

$$\Delta w_{ij} \propto -\frac{\partial F_m}{\partial w_{ij}} . \quad (\text{A.11})$$

Damit wird die partielle Ableitung eines Einzelfehlers als Schätzung des Gradienten benutzt.

Für jedes Eingangsmuster aus dem Trainingsdatensatz wird also folgender Ablauf durchgeführt

1. Vorwärtsdurchlauf (*forward pass*): Ein Eingangsmuster wird an das Netz angelegt; von Schicht zu Schicht werden die Aktivierungen weitergeleitet, bis schließlich an der letzten Schicht das Ausgabemuster als Vektor der Ausgabewerte o_j entsteht.

2. Fehlerrückvermittlung (*error backpropagation*):

- Das Ausgabemuster wird mit dem Sollausgabemuster verglichen und daraus der Fehlerterm nach (A.9) berechnet. Daraus ergeben sich für die Eingangsgewichte der Ausgabeeinheiten die Gewichtsänderungen nach (A.8).
- Für die vorletzte Schicht, d. h. die letzte verdeckte Schicht vor der Ausgeschicht werden die Fehlerterme nach (A.10) aus denen der Ausgeschicht berechnet; daraus ergeben sich dann ebenso die Gewichtsänderungen für deren Eingangsgewichte.
- Für eventuelle weitere versteckte Schichten werden die Fehlerterme ebenfalls nach (A.10) und die Gewichtsänderungen ebenfalls nach (A.8) berechnet, nur werden statt der Fehlerterme der Ausgabeeinheiten in (A.10) nun die Fehlerterme der nachfolgenden versteckten Schicht benutzt.

Dieser Ablauf wird wiederholt, bis das Fehlerminimum gefunden ist. In der Praxis wird man, schon aus Rechenzeitgründen, einen gewissen Gesamtfehler als untere Schranke ansetzen müssen oder die Anzahl der Durchläufe beschränken.

Probleme und Erweiterungen von Backpropagation: Backpropagation hat sich als sehr erfolgreiches Trainingsverfahren erwiesen, das in vielen Anwendungen eingesetzt wird. Es ist jedoch nicht frei von Problemen, für die verschiedene Abhilfen vorgeschlagen worden sind. Ein wichtiges Problem ist die Abhängigkeit von der Lernrate. Wird sie zu groß gewählt, kann das Netz mit einem Optimierungsschritt das Fehlerminimum überspringen und dann anfangen zu oszillieren; wählt man sie hingegen ausreichend klein, so ergeben sich außerordentlich lange Trainingszeiten. Abhilfe kann dagegen die in (Rumelhart und McClelland 1986) vorgeschlagene Einführung eines Momentum-Terms in die Formel für das Gewichtsupdate bringen. Sie lautet dann:

$$\Delta w_{ij}(t+1) = \eta o_i \delta_j + \alpha \Delta w_{ij}(t). \quad (\text{A.12})$$

Der Gewichtsänderung auf der Basis des aktuellen Fehlerwertes wird also ein gewisser Bruchteil der vorigen Gewichtsänderung hinzugefügt. Dies hat den Effekt, das Training auf großen Fehlerplateaus zu beschleunigen und in zerklüfteten Bereichen zu bremsen, um dort Oszillationen zu dämpfen. Der Momentum-Term α wird normalerweise auf einen Wert von 0,2 und 0,99 eingestellt.

Ein anderes Problem entsteht, wenn Neuronen im Sättigungsbereich der sigmoiden Aktivierungsfunktion arbeiten, da deren Ableitung dort nahezu verschwindet und somit ein gradientenbasiertes Verfahren außerordentlich langsam wird. In (Fahlman 1988) wird daher vorgeschlagen, konstant 0,1 zur Ableitung der

Aktivierungsfunktion zu addieren. Diese Änderung bezeichnet man als *Flat Spot Elimination*, da sie den flachen Bereich der Aktivierungsfunktion beseitigt.

A.2 Herleitung der Schärfentiefe

In diesem Abschnitt werden die in 8.7.3 angeführten Formeln für die Schärfentiefe hergeleitet.

A.2.1 Grenzabstände

Abbildung A.1 zeigt noch einmal die Entstehung eines Zerstreuungskreises in der Abbildungsebene. Wir nehmen an, dass das Objektiv auf eine Gegenstandsweite g_0 eingestellt ist, bei der das Bild in einem Abstand b_0 von der Linse entsteht. Nun werde der Gegenstand näher an das Objektiv herangebracht, auf eine Gegenstandsweite $g_1 < g_0$. Gemäß der Abbildungsgleichung (8.9) muss bei sinkender Gegenstandsweite die Bildweite wachsen. Ein scharfes Bild des Objekts entsteht demnach nun in einem Abstand $b_1 > b_0$ vom Objektiv. Wenn aber die Einstellung des Objektivs nicht geändert wird, befindet sich die Bildebene weiterhin im Abstand b_0 vom Objektiv, so dass dort keine punktförmige Abbildung entsteht, sondern ein Zerstreuungskreis.

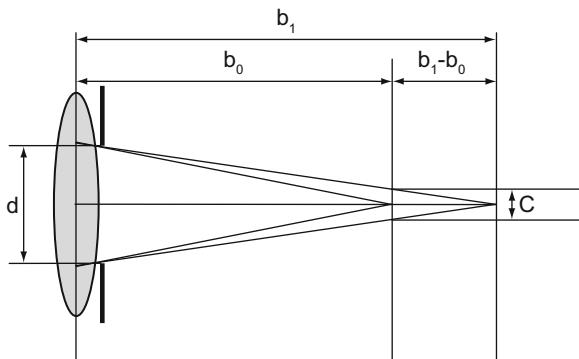


Abbildung A.1: Entstehung eines Zerstreuungskreises

Für die Größe des Zerstreuungskreises muss aufgrund des Strahlensatzes gelten:

$$\frac{C}{d} = \frac{b_1 - b_0}{b_1}, \quad (\text{A.13})$$

$$C = \frac{b_1 - b_0}{b_1} d. \quad (\text{A.14})$$

Wenn man darin die Blendenöffnung durch die Blendenzahl k (siehe Abschnitt 8.7.1) und die Brennweite f ausdrückt:

$$d = \frac{f}{k}. \quad (\text{A.15})$$

erhält man:

$$C = \frac{b_1 - b_0}{b_1} \frac{f}{k}. \quad (\text{A.16})$$

Nach der Abbildungsgleichung (8.9) lässt sich die Bildweite b_i eines Objekts durch Brennweite f und Gegenstandsweite g_i wie folgt ausdrücken:

$$\frac{1}{b_i} = \frac{1}{f} - \frac{1}{g_i} = \frac{g_i - f}{fg_i}, \quad (\text{A.17})$$

$$b_i = \frac{fg_i}{g_i - f}, \quad (\text{A.18})$$

Dies für b_0 und b_1 in Gleichung (A.16) eingesetzt, ergibt:

$$C = \frac{\frac{fg_1}{g_1 - f} - \frac{fg_0}{g_0 - f}}{\frac{fg_1}{g_1 - f}} \frac{f}{k}, \quad (\text{A.19})$$

$$C = \frac{fg_1(g_0 - f) - fg_0(g_1 - f)}{fg_1(g_0 - f)} \frac{f}{k}, \quad (\text{A.20})$$

$$C = \frac{fg_0 - fg_1}{g_1(g_0 - f)} \frac{f}{k}, \quad (\text{A.21})$$

$$C = \frac{f^2 g_0 - f^2 g_1}{kg_1(g_0 - f)}. \quad (\text{A.22})$$

Damit haben wir den Durchmesser des Zerstreuungskreises in Abhängigkeit von der eingestellten Gegenstandsweite g_0 und der tatsächlichen Gegenstandsweite $g_1 < g_0$. Es ist offensichtlich, dass dieser Durchmesser wächst, je kleiner g_1 wird. Dies äußert sich anschaulich so, dass das Bild immer unschärfer wird, je näher wir den Gegenstand an das Objektiv heranbringen. Es erhebt sich die Frage, wie nahe der Gegenstand an die Linse herangebracht werden kann, ohne dass eine *wahrnehmbare* Unschärfe entsteht. Das Kriterium hierfür ist der Durchmesser des Zerstreuungskreises. Wenn dieser hinreichend klein ist, wird die Unschärfe als nicht

wahrnehmbar betrachtet. Zu jedem Durchmesser C des Zerstreuungskreises lässt sich die Gegenstandsweite g_1 aus Gleichung (A.22) berechnen, und damit auch die minimal zulässige Gegenstandsweite g_{min} für einen maximal zulässigen Durchmesser C^* :

$$g_1 = \frac{f^2 g_0 - f^2 g_1}{Ck(g_0 - f)}, \quad (\text{A.23})$$

$$g_1 = \frac{f^2 g_0}{f^2 + Ck(g_0 - f)}, \quad (\text{A.24})$$

$$g_{min} = \frac{f^2 g_0}{f^2 + C^* k(g_0 - f)}. \quad (\text{A.25})$$

Damit ist jetzt der geringste Abstand zur Linse bekannt, in dem ein Objekt noch scharf abgebildet wird. Für den umgekehrten Fall, dass der Gegenstand vom Objektiv entfernt wird, kann man ganz entsprechend vorgehen, nur muss man berücksichtigen, dass der neue Abstand g_1 größer ist als die eingestellte Nennentfernung. Damit befindet sich jetzt das Abbild des Objekts dichter an der Linse, es gilt also $b_1 < b_0$. Da wir uns hier nur für betragsmäßige Größen interessieren, kehren wir die Differenz in Gleichung (A.14) um. Trigonometrisch ändert sich aufgrund der Dreiecksähnlichkeiten nichts, so dass jetzt für den Streukreisdurchmesser gilt:

$$C = \frac{b_0 - b_1}{b_1} \frac{f}{k}, \quad (\text{A.26})$$

und nach einer entsprechenden Umformung entsteht für den hinteren Grenzabstand:

$$g_{max} = \frac{f^2 g_0}{f^2 - C^* k(g_0 - f)}. \quad (\text{A.27})$$

Aus den Gleichungen (A.25) und (A.27) für den vorderen und hinteren Grenzabstand ergibt sich die Größe des Schärfentiefbereichs zu:

$$\Delta g = g_{max} - g_{min} = \frac{f^2 g_0}{f^2 - C^* k(g_0 - f)} - \frac{f^2 g_0}{f^2 + C^* k(g_0 - f)} \quad (\text{A.28})$$

$$= \frac{2f^2 g_0 C^* k(g_0 - f)}{f^4 - C^{*2} k^2 (g_0 - f)^2}. \quad (\text{A.29})$$

Die zulässige Unschärfe C^* ist eine ziemlich kleine Größe. Wenn man die halbe Ausdehnung eines Sensorelementes benutzt, so liegt C^* im Bereich von fünf Mik-

rometer, C^{*2} also bei etwa $2,5 \cdot 10^{-5}$ mm. Solange g_0 nicht erheblich größer ist als die Brennweite f , fällt der zweite Term im Nenner gegenüber f^4 nicht ins Gewicht, so dass man Gleichung (A.29) entsprechend (Ernst 1991) für solche Nahaufnahmen folgendermaßen vereinfachen kann:

$$\Delta g = \frac{2C^* k g_0 (g_0 - f)}{f^2}. \quad (\text{A.30})$$

A.2.2 Schärfentiefe im Unendlichen

Die Näherung in Gleichung (A.30) erzeugt bei großen Gegenstandsweiten einen durchaus erheblichen Fehler. Andererseits beobachtet man in Gleichung (A.29), dass für große Werte von g_0 der Nenner negativ wird, man also einen Bereich negativer Größe erhält. Entsprechend wird in Gleichung (A.27) für den Grenzfall unendlichen Nennabstandes $g_0 \rightarrow \infty$ der hintere Grenzabstand negativ. Das hängt damit zusammen, dass die Gleichungen oben für den maximal zulässigen Durchmesser des Zerstreuungskreises abgeleitet wurden. Im Grenzfall unendlichen Abstandes fallen jedoch alle Lichtstrahlen von dem Objekt parallel ein, so dass sie immer in der Brennebene exakt fokussiert sind (das ist natürlich auch genau der Grund, warum ab einer gewissen Entfernung jedes Objekt scharf abgebildet wird). Damit stellt sich hier die Frage, ab welchem Abstand g der Zerstreuungskreis eines Objekts in der Brennebene der Linse kleiner ist als der höchstzulässige Durchmesser.

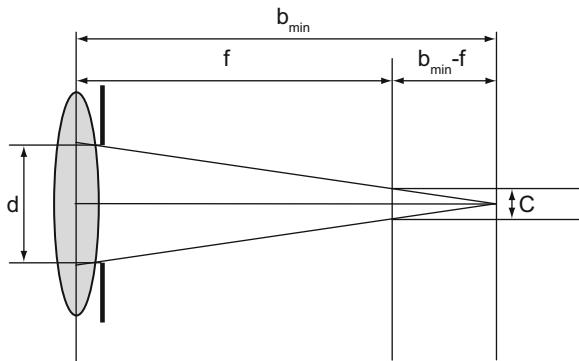


Abbildung A.2: Zerstreuungskreis bei Fokussierung auf die Brennebene

Abbildung A.2 zeigt noch einmal das Entstehen des Zerstreuungskreises, wobei hier angenommen wird, dass das Objekt so eingestellt ist, dass es gerade in seiner eigenen Brennebene fokussiert. Die Nennbildweite b_0 ist also jetzt gleich der Brennweite f . Ein Objekt, das zu nahe an der Linse ist, wird hinter der Brennebene

fokussiert, erzeugt also einen Zerstreuungskreis vom Durchmesser C in der Brennebene. Der Durchmesser dieses Zerstreuungskreises ergibt sich aus Gleichung (A.16), indem dort für die Nennbildweite b_0 die Brennweite eingesetzt wird:

$$C = \frac{b_{min} - f}{b_{min}} \cdot \frac{f}{k}. \quad (\text{A.31})$$

Indem man nach dem Abbildungsgesetz die Bildweite durch die Gegenstandsweite wie folgt ausdrückt:

$$b_{min} = \frac{fg_{min}}{g_{min} - f}, \quad (\text{A.32})$$

ergibt sich für den Durchmesser des Zerstreuungskreises in der Brennebene abhängig vom Objektabstand g_{min} :

$$C = \frac{\frac{fg_{min}}{g_{min} - f} - f}{\frac{fg_{min}}{g_{min} - f}} \cdot \frac{f}{k}, \quad (\text{A.33})$$

$$C = \frac{f^2}{g_{min}k}. \quad (\text{A.34})$$

Wenn wir jetzt fordern, dass dieser Zerstreuungskreisdurchmesser kleiner sein muss als C^* , dann folgt für den erforderlichen Abstand:

$$C^* > \frac{f^2}{g_{min}k}, \quad (\text{A.35})$$

$$g_{min} > \frac{f^2}{C^*k}. \quad (\text{A.36})$$

Das bedeutet, jedes Objekt, das sich in mindestens $f^2/(C^*k)$ Abstand vom Objektiv befindet, wird mit einer Unschärfe von weniger als C^* in der Brennebene abgebildet. Wenn man als zulässige Unschärfe die halbe Sensorelementgröße verwendet,

also etwa fünf Mikrometer, ergibt sich bei einem 12 mm Objektiv und einer Blendenzahl von 11 ein Wert von 2618 mm für diesen Mindestabstand².

A.2.3 Abhängigkeit der Schärfentiefe von der Brennweite

Man kann leicht zeigen, dass bei einer Vergrößerung der Brennweite der vordere Grenzabstand zunimmt. Wenn die Brennweite um einen Betrag Δ vergrößert wird, dann ergibt sich aus

$$g_{\min}(f + \Delta) > g_{\min}(f), \quad (\text{A.37})$$

$$\frac{(f + \Delta)^2 g_0}{(f + \Delta)^2 + C^* k(g_0 - (f + \Delta))} > \frac{f^2 g_0}{f^2 + C^* k(g_0 - f)}, \quad (\text{A.38})$$

nach einigen Umformungen

$$2f g_0 - 2f^2 + \Delta g_0 > 0. \quad (\text{A.39})$$

Wir hatten vorausgesetzt, dass die Brennweite vergrößert wird, daher ist $\Delta > 0$. Da wir mit betragsmäßigen Größen rechnen, ist auch $g_0 > 0$. Der letzte Summand ist also in jedem Fall positiv. Da innerhalb der Brennweite keine scharfe Abbildung möglich ist, ist die eingestellte Gegenstandsweite g_0 größer als die Brennweite f , womit die Differenz $fg_0 - f^2$ ebenfalls positiv sein muss. Ungleichung (A.39) ist daher immer wahr, so dass auch die ursprüngliche Ungleichung (A.38) wahr sein muss. Damit ist gezeigt, dass bei einer Vergrößerung der Brennweite die minimale Gegenstandsweite für eine scharfe Abbildung ebenfalls größer wird. Entsprechend kann man zeigen, dass der hintere Grenzabstand bei wachsender Brennweite sinkt. Das erklärt den geringen Schärfentiefenbereich von Teleobjektiven mit ihrer großen Brennweite.

² Der Wert für den hinteren Grenzabstand nach Formel (A.27) wird erst bei einem größeren Abstand negativ, was damit zusammenhängt, dass C in dieser Formel unveränderlich ist, tatsächlich aber mit zunehmender Entfernung schrumpft. Für alle praktischen Zwecke ist sie aber genau genug.

B. Software-Download

Alle Beispiele dieses Buches und die Bildverarbeitungssoftware **NeuroCheck**, mit der sowohl die Beispiele als auch die entsprechenden industriellen Anwendungen realisiert wurden, stehen als Download auf der Seite <http://extras.springer.com/> des Springer-Verlags zur Verfügung. Die Software läuft auf den Betriebssystemen Windows® XP, Windows® Vista® und Windows® 7 der Firma Microsoft. Für die Verwendung der Software ist das .NET-Framework mindestens in der Version 3.5 erforderlich. Dieses ist nicht im Download enthalten, kann aber kostenlos von der Website von Microsoft (www.microsoft.com) bezogen werden. Wegen der dynamischen Entwicklung im Software-Bereich planen wir die Software und die weiteren im Download enthaltenen Daten regelmäßig zu aktualisieren. Von daher können sich die oben genannten Anforderungen verändern. Wir möchten Sie bitten, weitere Informationen über den Inhalt und Aufbau des Downloads der Datei LiesMich.txt zu entnehmen.

In der zum Download bereitgestellten Demo-Version von NeuroCheck ist die gesamte Bildverarbeitungsfunktionalität uneingeschränkt verfügbar. Dadurch können alle im Buch beschriebenen Beispiele vollständig nachvollzogen werden und weitere Experimente mit den mitgelieferten Bildern durchgeführt werden. Die funktionalen Einschränkungen der Demo-Version betreffen vor allem die Benutzung der Software für den Aufbau und Betrieb automatischer Sichtprüf'anlagen.

Sämtliche im Buch beschriebenen Beispiele sind im Verzeichnis „Beispiele“ des Downloads enthalten. In den Beschreibungstabellen der umfangreichen Anwendungsbeispiele sind die entsprechenden Unterverzeichnisse angegeben.

Weitere Informationen zur Software NeuroCheck und ihren Anwendungen finden sie auf der Website <http://www.neurocheck.com> der NeuroCheck GmbH.

C. Weblinks zur industriellen Bildverarbeitung

Die hier zusammengestellten Weblinks bilden eine kleine Auswahl an Websites rund um das Thema industrielle Bildverarbeitung. Unser Ziel war es dabei, die Vielfalt an Firmen und Organisationen zu präsentieren, die sich in irgendeiner Form mit industrieller Bildverarbeitung beschäftigen, und nicht, eine repräsentative oder auch nur in Teilen vollständige Liste vorzulegen.

- www.vdma.org/vision/
Fachabteilung Industrielle Bildverarbeitung, Verband Deutscher Maschinen- und Anlagenbau e.V. (abgekürzt VDMA, Frankfurt am Main)
- www.emva.org
Europäischer Fachverband European Machine Vision Association (abgekürzt EMVA, Frankfurt am Main)
- www.neurocheck.de
Firma NeuroCheck GmbH (Remseck, Stuttgart)
- www.baumer.com/vision/
Firma Baumer Optronic (Radeberg)
- www.baslerweb.com
Firma Basler AG (Ahrensburg)
- www.messe-stuttgart.de/vision/
Fachmesse VISION, Landesmesse Stuttgart GmbH
- www.inspect-online.com
Fachzeitschrift Inspect, GIT Verlag GmbH & Co. KG (Darmstadt)
- www.fbmh.h-da.de
Studiengang Optotechnik und Bildverarbeitung, Hochschule Darmstadt

Literaturverzeichnis

1. Ballard DH (1981) Generalizing the Hough transform to detect arbitrary shapes. Pattern recognition 13(2): 111-122
2. Bishop CM (1995) Neural Networks and Pattern Recognition. Oxford University Press, Oxford
3. Brause R (1991) Neuronale Netze. Teubner, Stuttgart
4. Bronstein IN, Semendjajew KA, Musiol G, Mühlig H (2005) Taschenbuch der Mathematik. 6. Aufl. Harri Deutsch, Frankfurt a. M.
5. Davis LS (1981) Image texture analysis techniques – a survey. In Simon JC, Haralick RM (Hrsg) Digital Image Processing, Reidel, Boston
6. Ernst H (1991) Einführung in die digitale Bildverarbeitung. Franzis, München
7. Fahlman SE (1988) An empirical study of learning speed in back-propagation networks. In Touretzky D, Hinton G, Sejnowski T (Hrsg) Proceedings of the 1988 Connectionist Models Summer School, Morgan Kaufmann, San Francisco
8. Freeman H (1961) On the encoding of arbitrary geometric configurations. IEEE Transactions EC 10(2):260–268
9. Gonzalez RC, Woods RE (2008) Digital Image Processing. 3. Aufl. Pearson Education, Upper Saddle River
10. van Gool L, Dewaele P, Oosterlinck A (1983) Texture analysis anno 1983. Computer Vision, Graphics and Image Processing 29(3):336–357
12. Hebb DO (1949) The Organization of Behavior. John Wiley & Sons, New York
13. Hopfield JJ (1982) Neural networks and physical Systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences 79(8):2554–2558
14. Hornberg A (2006) Handbook of Machine Vision. Wiley-VCH-Verlag, Weinheim
15. Hu MK (1962) Visual Pattern Recognition by Moment Invariants. IRE Transactions on Information Theory 8(2):179–187
16. Illuminating Solutions (1996) Dolan Jenner Industries, Lawrence, USA
17. Jähne B (2005) Digitale Bildverarbeitung. 6. Aufl. Springer, Berlin Heidelberg New York
18. Jähne B, Massen R, Nickolay B, Scharfenberg H. (1995) Technische Bildverarbeitung – Maschinelles Sehen. Springer, Berlin, Heidelberg, New York
19. Julesz B (1975) Experiments in the visual perception of texture. Scientific American 232(4):34–43
20. Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biological Cybernetics 43:59–69
21. Kohonen T (1989) Self-Organization and Associative Memory. Springer, Berlin, New York, Heidelberg
22. Lenk B (2005) Optische Identifikation. Monika Lenk Fachbuchverlag, Kirchheim unter Teck
23. Mandelbrot B (1987) Die fraktale Geometrie der Natur. Birkhäuser, Basel
24. McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5:113–133
25. Nischwitz A, Fischer M, Haberäcker P (2007) Computergrafik und Bildverarbeitung. 2. Aufl. Vieweg, Wiesbaden
26. Pao YH (1989) Adaptive Pattern Recognition and Neural Networks. Addison-Wesley, Reading
27. Parker JR (1994) Practical Computer Vision Using C.Wiley, New York
28. Pentax (2008) Pentax CCTV Catalogue. Pentax Imaging System Division, Tokyo
29. Pschyrembel W (2007) Pschyrembel Klinisches Wörterbuch. 261. Aufl. de Gruyter, Berlin.

30. Reilly DL, Cooper LN, Elbaum C (1982) A neural model for category learning. *Biological Cybernetics*, 45:35–41
31. Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and Organization in the brain. *Psychological Review* 65(6):386–408
32. Rosenfeld A, Kak AC (1982) *Digital Picture Processing*. 2. Aufl. Academic Press, New York.
33. Rumelhart DE, McClelland JL (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, Cambridge
34. Russ JC (2007) *The Image Processing Handbook*. 5. Aufl. CRC Press, Boca Raton
35. Sahoo PK, Soltani S, Wong AKC, Chen, Y.C. (1988): Survey of thresholding techniques. *Computer Vision, Graphics and Image Processing*, 41(2):233–260
36. Schröder G, Treiber H (2002) *Technische Optik*. 9. Aufl. Vogel, Würzburg
37. Schürmann J (1977) *Polynomklassifikatoren für die Zeichenerkennung: Ansatz, Adaption, Anwendungen*. R. Oldenbourg, München
38. Schwenk M (1991): *Mustererkennung mit Neuronalen Netzen*. Diplomarbeit, Institut für Technische Optik, Universität Stuttgart
39. Smith M (1993) *Neural networks for Statistical modeling*. Van Nostrand Reinhold, New York
40. Sonka M, Hlavac V, Roger B (2008) *Image Processing, Analysis, and Machine Vision*. 3. Aufl. Cengage Learning, Stamford
41. Steger C, Ulrich M, Wiedemann C (2008) *Machine vision algorithms and applications*. Wiley-VCH-Verlag Weinheim
42. Theuwissen AJP (1996) *Solid-State Imaging with Charge-Coupled Devices*. Kluwer Academic Publishers, Dordrecht, Boston, London
43. Tönnies KD (2005) *Grundlagen der Bildverarbeitung*. Pearson Studium, München, Boston.
44. Waszkewitz P (1999) *Detektierung von Beschriftungen auf metallischen Oberflächen mit Hilfe von Texturmethoden und Neuronalen Netzen*. Dissertation, Universität Stuttgart
45. Zell A (1994) *Simulation Neuronaler Netze*. Addison-Wesley, Bonn, Paris, Reading

Index

A

A/D-Wandler *Siehe* Analog-Digital-Wandler
Abbildungsfehler 271, 278
Abbildungsgleichung 268, 269, 272, 294
Abbildungsmaßstab 198, 223, 267, 268, 269, 294
Aberration, chromatische 278
Abstandsmessung 84, 204
Abstandsschwankungen 220
Abtasttheorem 11
Abwickeln 133, 318
Achromate 278
Achsenberechnung 80, 81
Achsensymmetrie 82
Active Pixel Sensoren 233
Adaptionsregel 188
AEC 244
AGC 244
Aktivierungsfunktion 185–91
Aktivierungszustand 187, 188
Analog-Digital-Wandler 231, 234
Antastmessung 205
Anwesenheitskontrolle 73, 196, 297–338
Apochromate 278
Artefakt 102, 315
Auflagemäß 279, 294
Auflichtbeleuchtung 18, 155, 213, 286,
 295
 diffus 286, 295
 gerichtet 215, 286, 295, 362
Auflösung 239
Ausgabeeinheit 190
Ausgabeschicht 186, 188, 190
Ausgangsverstärker 231
Ausgangswert 175, 187
Ausgleichsgeometrie 211, 215, 218, 225, 326
Ausgleichsgerade 208–10, 214, 217, 326
Ausgleichskreis 217, 219, 324

B

Backpropagation 188, 194
Barcode-Identifikation 127–33, 167
Bayer-Pattern 237, 245
Bayes, Satz von 177
Bayes-Klassifikator 178, 193
Beleuchtung
 Auflicht *Siehe* Auflichtbeleuchtung
 Blitz- 160, 163, 295
 Dunkelfeld- 155–57, 159, 287, 295
 Durchlicht *Siehe*
 Durchlichtbeleuchtung
 Hellfeld- 286
 im Strahlengang 289, 296
 polarisiert 138, 288, 292, 296
 ringförmig 137, 278, 284, 288, 295,
 314
 strukturiert 289, 296
 telezentrisch 291, 296
Beleuchtungsgradient 36, 42, 70, 121
Belichtungszeit 161, 230, 233, 234, 243,
 265, 266
Bereichsbinarisierung 102, 125
Beschriftung 117, 136, 138, 145, 149–51,
 152, 154, 168
Bestkreise 218
Bildaddition *Siehe* Bildarithmetik,
 Addition
Bildarithmetik 29, 38, 42, 69, 70
 Addition 39
 Division 42
 Maximum 42
 Minimum 42
 Mittelung 39
 Shading-Korrektur 42–44
 Subtraktion 39–41
Bilddatenübertragung 250–59
Bilddivision *Siehe* Bildarithmetik,
 Division
Bildkontrast 33
Bildmittelung *Siehe* Bildarithmetik,
 Mittelung

Bildrate 232, 244
 Bildsensor *Siehe* Halbleitersensor
 Bildsubtraktion *Siehe* Bildarithmetik,
 Subtraktion
 Bildtransformation 69
 Bildvergleich 6, 163
 Bildvorverarbeitung 16, 20, 29–71, 98,
 103, 104, 129, 135, 152, 156,
 168, 314
 Bildweite 266–72, 273, 276, 277, 279,
 280, 294
 Binning 245
 Blende 223, 265–66, 273, 274, 276, 278,
 294
 Blendenzahl 265–66, 274, 294
 Blitzbeleuchtung *Siehe* Beleuchtung,
 Blitz-
 Blooming 242
 Brennebene 268, 274, 282
 Brennpunkt 266, 282
 Brennweite 265–72, 274, 275, 276, 277,
 278, 281, 294

C

CameraLink 250, 251–53, 259, 293
 CCD-Sensor 228, 230–32, 242, 292
 Chromatische Aberration *Siehe*
 Aberration, chromatische
 Closing 64, 71
 CMOS-Sensor 229, 233–35, 242, 293
 C-Mount *Siehe* Auflagemaß
 Color Filter Array *Siehe* Farbfiltermaske
 CS-Mount *Siehe* Auflagemaß

D

Dark signal *Siehe* Rauschen, thermisch
 Datei 385
 DCAM 254
 Demosaicing 245, 293
 Differenzbild 165, 166, 169
 Digital-I/O 383
 Digitalisierung 11
 Dilatation 63, 71, 158, 314, 319, 321, 335
 Diskretisierung 105, 133
 DMA 251
 Dunkelfeldbeleuchtung *Siehe*
 Beleuchtung, Dunkelfeld-
 Dunkelstrom *Siehe* Rauschen, thermisch
 Durchlichtbeleuchtung 18, 77, 197, 216,
 220, 224, 226, 286, 289, 296

diffus 201, 211, 218, 290, 296
 gerichtet 290, 296
 polarisiert 304
 telezentrisch 220, 222, 291, 296
 Dynamikumfang 230, 248

E

Eingangsschicht 186
 Eingangsvektor 177, 179, 184, 185
 Endoskop 283, 295
 Entscheidungsfunktion 177, 184, 191, 194
 Ergebnis 386
 Ergebnisvektor 175, 178, 193–94
 Erosion 62, 71
 Ethernet 384
 Exposure *Siehe* Belichtungszeit

F

Farbbeschreibung 362
 Farbbildverarbeitung 357–79
 Farbfiltermaske 237, 245
 Farbidentifikation 358–72
 Farbkanal 236, 245
 Farbsegmentierung 372–78, 379
 Fehlerrate 145, 180, 193
 Feldbus 383
 Fernaufnahme 276
 Fertigungsintegration 382
 Filter 79, 126, 168, 169
 angepasst 52–53
 Differenz- 65, 157, 158
 Gauß- 53–54
 Glättungs- 48–54, 71
 Kanten- 54–60, 71
 Kontrast- 65, 71, 335, 367
 Laplace- 58
 linear 44–60
 Median- 60–61, 71
 Mittelwert- 46, 48–52, 60, 61, 71, 335
 morphologisch 62–64, 71
 Randeffekt 47–48
 Rangordnungs- 60
 Rechteck- 49–52, 60, 71
 reduziert 51–52
 Sobel- 58–59
 Sobel-Betrags- 59–60
 Filterkern *Siehe* Filteroperator
 Filtermaske *Siehe* Filteroperator
 Filteroperator 45–59, 60, 62, 65, 70
 FireWire 250, 253–55, 293

Fixed Pattern Noise 233, 248
 Fläche, lichtempfindlich 229, 234, 238,
 240, 245
 Format, optisch 239
 Fouriertransformation 69
 Frame Grabber 250, 251
 Full Well Capacity 229
 Füllfaktor 240
 Funktionalklassifikator 184–85, 194

G

Gain *Siehe* Signalverstärkung
 Gamut 238
 Gegenstandsweite 266–72, 276, 277, 280,
 294
 Gewichtsfaktor 186, 188, 194
 Gigabit-Ethernet 250, 256–59, 293
 GigE Vision 256
 Glasfaserlichtleiter 285, 289, 295
 Global Shutter 234
 Graustufenquantisierung 12
 Grauwertgradient 102, 214
 Grauwertstatistik 104
 Grauwerttransformation 29, 30–38, 69
 Grenzmuster 143, 189

H

Halbbild 232
 Halbleitersensor 227–42, 292
 Farb- 236–38, 293
 Frame-Transfer- 231
 Full-Frame- 231
 Interline-Transfer- 230, 273
 Multi-Tap- 232, 235
 Halogenlampe 285, 295
 Hauptachsenberechnung 80–83, 88, 93
 Hellfeldbeleuchtung *Siehe* Beleuchtung,
 Hellfeld-
 Helligkeitsschwankung 220, 284
 Hintergrundbeleuchtung *Siehe*
 Durchlichtbeleuchtung
 Histogrammanalyse 102, 103, 124
 Histogrammausgleich 102
 Histogramm-Ebnung 34–36, 70
 Homogenität 248
 Hough-Transformation, generalisierte
 119, 126
 HSI-Farbraum 362, 374, 379
 Hue 362

I

IEEE1394 *Siehe* FireWire
 IIDC *Siehe* DCAM
 Industrial-Ethernet 384
 Informationsreduktion 14
 Informationsverlust 11, 12
 Infraroteinstrahlung 278
 Innenbohrungen 289
 Integraltransformationen 71
 Intensity 363
 Interlaced Scanning 232
 Interpolation
 bilinear 67
 Pixelwiederholung 67
 Interpolationsfunktion 122, 214

K

Kalibrierfaktor 198
 Kalibriermodus 223–24
 Kalibriernormal 223, 226
 Kaltlichtquelle 285, 289, 295
 Kamera
 Ausgabeformat 247
 digital 222, 226
 Digital 293
 Digital- 242–50
 Drei-Chip-Farb- 237
 Dreilagenfarb- 238
 Ein-Chip-Farb- 237, 245
 intelligente 9
 Schwenk-Neigekopf-Zoom- 298
 Zeilen- 259–64, 294
 Kamera, Farb- 245
 Kantenantastung 121, 122, 126
 Kantenbild 42, 121, 126
 Kantendetektion 126, 149, 213, 214–15,
 225
 Kantenmodell 121, 122, 124, 126, 225
 Kantenrichtung 57–59, 121
 Kantensteilheit 57
 Kennzeichnungidentifikation 5, 162
 Kettencode 108, 140, 144
 Klassengrenze 179–84, 193
 Klassenindex 175, 176
 Klassenraum 172, 173, 176–79, 192
 Klassifikation 171–94
 Klassifikationsanwendung 139, 141–47,
 153, 168, 171, 192, 357
 Klassifikationsfunktion 174, 179, 192, 194

- Klassifikationsverfahren 189, 190, 193, 194
 Kollimatorlinse 291
 Kontrastfilter *Siehe* Filter, Kontrast-Kontrastnormierung 33–34, 39, 70, 145, 146, 168
 Lokale 36–37
 Kontrastumfang *Siehe* Dynamikumfang
 Kontrastverbesserung 34, 35
 Konturabstände 201–5, 225
 Konturdarstellung 108, 125
 Konturverfolgung 75, 104, 107, 125, 140, 145, 155, 169, 321
 Konvexes Objekt 202
 Koordinatentransformation 66, 71
 Korrelationskoeffizient 109, 110, 112, 115–20, 125, 194, 319, 338
 Korrelationspunkte *Siehe* Template Matching, Korrelationspunkte
 Kostenfunktion 177, 178
 Kreisberechnung 216
- L**
- Ladungsverstärker 231, 233, 241
 Lageerkennung 5, 73, 96
 Laplacefilter *Siehe* Filter, Laplace-Laserlinie 285, 325–26, 338
 Lastenheft 381
 Lernverfahren 188, 193
 Leuchtdiode 284, 288, 289, 290, 295
 Leuchtfelder 284
 Leuchtplatte 197, 207, 211, 285, 290, 292
 Leuchtstoffröhre 285, 289, 295, 314
 Lichtempfindlichkeit 159, 161, 163
 Lichtstärke 266, 294
 Linearität 241, 248
 Linienprojektor 287, 289
 Linsenabbildungen 266–79
- M**
- Makroaufnahme 277
 Massendaten 387
 Matching
 formbasiertes 118, 126
 geometrisches 120, 126
 Template *Siehe* Template Matching
 Matrixsensor 228, 239
 Maximumbild 59
 Maximumoperator 42, 59
 Medianfilter *Siehe* Filter, Median-Merkmalsraum 145, 172, 178, 180, 183, 190–91, 192, 193
 Merkmalskalierung 189–90, 194
 Merkmalsvektor 174, 177, 178, 182, 183, 192–93
 Midspan Devices 259
 Mikroaufnahme 277
 Mikrolinse 240
 Minimumoperator 42
 Mittelwertfilter *Siehe* Filter, Mittelwert-Multi-Core-Prozessoren 387
 Multireferenzenklassifikator 179–84, 190, 191, 193
 Musterteil 382
- N**
- Nachbarschaftsdefinition 105
 Nächster-Nachbar-Klassifikator 179–81, 193, 374
 Nahaufnahme 203, 280
 Nebenachse 82
 Netzwerk 384
 NeuroCheck 27, 401
 Neuronale Netze 185–92, 194
- O**
- Objekterkennung 5
 Objektform 77
 Objektgruppe 89, 93, 217
 Objektiv 227–83
 1", 2/3", 1/2", 1/3" 281
 C-Mount 279
 CS-Mount 279
 Fischaugen 282
 Größe 281, 295
 Superweitwinkel 282
 Tele- 268, 275
 telezentrisch 220, 222, 226, 282, 291, 295
 Objektkante 76–80, 93, 121, 126, 147
 Objektkontur 93, 96, 106, 107
 Objektmerkmale
 Fläche 315
 Kompaktheit 315
 Umfang 315
 Objektschwerpunkt 75, 82, 92, 93
 Objektsegmentierung 22
 Objektsuche 161
 Opening 63–64, 71
 Ortsdiskretisierung 80

Overlapped-Modus 231

P

Parallelisierung 8, 45, 162
 Parameter 386
 Partial Scan 245
 Perzeptron 171, 185–92, 194
 Pflichtenheft 381
 Photonenrauschen 241
 Pixel Pitch *Siehe* Pixelgröße
 Pixelgröße 238
 Pixelwiederholung *Siehe* Interpolation,
 Pixelwiederholung
 Plug-and-Play 253
 Polarabstandsprojektion 83–85, 88, 93
 Polarisationsfilter 138, 285, 288, 292, 296
 Polynomklassifikator 179, 184–85, 190,
 194
 Positionierung 73–93
 Drehzentrum 90
 Gesamtobjekt 74–76
 Kante 76–80
 Objektgruppe 89–90
 Orientierung Hauptachse 80–83
 Orientierung Kontur 83–85
 Pseudoverschiebung 92
 Referenz 76, 86, 149, 169, 217, 315,
 316
 Referenzobjekt 73–76, 82, 83, 85, 91,
 93, 165, 310, 315, 316, 333
 Referenzposition 74, 76, 89, 92, 93
 Rotation 85, 88, 91–93, 133
 Rotation Einzelobjekt 80–85
 Verschiebung 86, 90, 91, 92, 93
 Verschiebung Einzelobjekt 73–80
 Positionskorrektur 76, 80, 93, 133, 147–
 51, 163, 214, 216, 221, 300,
 309, 314
 Positionsreferenz 300
 Positionsschwankungen 73, 219
 Power Injectoren 259
 Primärvalenz 236, 238
 Progressive Scan 232
 Prototypmuster 179, 182, 183
 Pseudokanten 47
 Pseudoverschiebung 79, 92, 93
 Punktnachbarschaft 105
 Punktoperation 34, 37, 38
 Punktoperationen 29, 69
 Pyramiden suche *Siehe* Template
 Matching, Pyramiden suche

Q

Quadratische Pixel 198, 223, 226
 Quanteneffizienz 239, 240, 293

R

Radius 216, 218, 322
 Randstrahlen 270, 272, 282
 Rangordnungsfilter *Siehe* Filter,
 Rangordnungs-
 Rasterbild 140, 141, 159, 161, 163, 183,
 189, 192, 319
 Rauschen 241
 Auslese- 248
 thermisch 248
 Verstärker- 248
 RCE-Netz 181–82, 193
 Rechteckfilter *Siehe* Filter, Rechteck-
 Referenzbild 30, 40, 41, 43, 150, 164, 169
 Referenzobjekt *Siehe* Positionierung,
 Referenzobjekt
 Referenzteil 223, 224, 226, 338
 Reflexion 65, 137, 138, 140, 155, 287,
 288, 307
 Region of Interest 16
 Regressionsgerade 208, 210, 214, 225
 Relativ Response 241
 RGB-Farbraum 362, 374, 379
 Ringbeleuchtung *Siehe* Beleuchtung,
 ringförmig
 Ringlicht *Siehe* Beleuchtung, ringförmig
 Roboterpositionierung 85–92, 92
 Rolling Shutter 234
 Rundheit 218

S

Saturation 363
 Schärfentiefe 272–76, 294
 Schieberegister *Siehe* Transportregister
 Schrifterkennung 16, 17, 18–21, 136–59,
 172, 173, 192
 Schrittweite *Siehe* Template Matching,
 Schrittweite
 Schutzgrad 249
 Schwerpunktsabstand 202
 Schwerpunktsberechnung 76, 78, 200, 226
 Schwerpunktskoordinaten 224

- Segmentierung 14, 16, 18, 23, 76, 95–125, 149, 150, 151, 155–59, 308, 318, 321, 323, 325, 333, 338
 binär 96–104
 Farbe 357
 Kanten- 121–24
 Kontur- 104–8
 Template-Matching- 108–20
 Seitenlicht 220, 290, 305
 Sensorelement 272, 273, 294
 Serielle Schnittstelle 384
 Shading-Korrektur *Siehe* Bildarithmetik, Shading-Korrektur
 Shutter 161
 Signal-Rausch-Verhältnis 248
 Signalverstärkung 243
 Skalierungsmethode 101
 Smear 242
 Sobel-Betragsfilter *Siehe* Filter, Sobel-Betrags-
 Sobelfilter *Siehe* Filter, Sobel-Speicherprogrammierbare Steuerung 8
 Spektrale Empfindlichkeit 240, 293
 Spezialoptiken 283
 Spezifikation 4, 149, 153, 164
 Spotvorsatz 285
 Standardisierung 8
 Steuersignal 385
 Strukturbreite 113, 158, 159
 Subpixelberechnung *Siehe* Subpixelinterpolation
 Subpixelgenauigkeit 124, 226
 Subpixelinterpolation 124, 220, 221–22, 225, 226
- T**
- Telezentrisches Objektiv *Siehe* Objektiv, telezentrisch
 Template Matching 108–20, 125, 149, 157–59, 169, 183, 194, 300, 316, 318, 323, 338, 370
 Ähnlichkeitsmaß 109
 Korrelationspunkte 115
 Optimierung 113, 125
 Prinzip 109
 Pyramidensuche 113–14, 125, 149
 Schrittweite 113–14, 118, 125, 149, 159, 319
 Unterabtastung 114–15
- Textur 69
 Time-of-Flight-Sensor 235
 TOF-Sensor *Siehe* Time-of-Flight-Sensor
 Training 143, 168, 188, 189, 191
 Trainingsdaten 143, 146, 152, 154, 168, 175, 184, 188, 190, 193, 194
 Translation *Siehe* Positionierung, Verschiebung
 Transportregister 230
 Trennebene 190
 Trigger 160, 163, 215, 244
- U**
- USB 250, 255–56
- V**
- Vektorquantisierung 182, 193
 Verarbeitungseinheit 185, 186, 188
 Verdrehung 79, 83, 112, 117, 125, 147, 150
 Vermessung 11, 85, 121, 195–226, 282, 287, 289, 291, 296, 323, 338
 Vorverarbeitung *Siehe* Bildvorverarbeitung
- W**
- Weißabgleich 247
 Wellenlänge 240
 White Balance *Siehe* Weißabgleich
 Windows 2, 387
 Winkelinkrement 93
 Winkelmessung 205–6, 208, 212, 225
- X**
- Xenon-Lampen 285
- Z**
- Zeichenerkennung 140
 Zeilenkamera *Siehe* Kamera, Zeilen-Zeilsensor 228, 239
 Zerstreuungskreis 272–76
 Zwischenring 279, 280, 295