

HANSER



Leseprobe

zu

Angewandte Bildverarbeitung und Bildanalyse

von Joachim Ohser

Print-ISBN: 978-3-446-47910-4
E-Book-ISBN: 978-3-446-48055-1

Weitere Informationen und Bestellungen unter
<https://www.hanser-kundencenter.de/fachbuch/artikel/9783446479104>

sowie im Buchhandel

© Carl Hanser Verlag, München

Vorwort

Die digitale Bildverarbeitung ist seit vielen Jahrzehnten ein sich sehr schnell entwickelndes Gebiet. Die Ursachen liegen in der immer größer werdenden Vielfalt der Bildgebungsverfahren, den inzwischen zahllosen Anwendungsgebieten und verfeinerten technischen Möglichkeiten für die Verarbeitung und Analyse der Bilddaten. Zu Letzteren gehören die Parallelisierung [43], das Processing auf einer Graphikkarte (*graphics processing unit*, GPU) und die Verwendung programmierbarer logischer Schaltungen (vor allem *field programmable gate arrays*, FPGA), siehe z. B. [10], wo die Leistungen dieser Techniken verglichen werden. Eine zunehmende Rolle spielt dabei die Programmierarchitektur CUDA (*compute unified device architecture*), mit deren Hilfe Algorithmen auf der GPU prozessiert werden kann [68, 256]. Die Beschleunigung von Bildverarbeitungsalgorithmen bietet sich wegen ihrer oft inhärenten Parallelität an und eröffnet damit stets neue Anwendungen. Die verschiedenen Anwendungsgebiete beeinflussen ihrerseits die Bildverarbeitung nicht nur dahingehend, dass sie die Entwicklung neuer Methoden anregen; sie tragen darüber hinaus zur Etablierung neuer Teilgebiete bei, die sich durch ihre Ausrichtungen und Fachbegriffe voneinander unterscheiden. Seit den Anfängen der Bildverarbeitung sind das vor allem die Fotogrammetrie sowie die mikroskopische Bildverarbeitung in Medizin [166, 292], Mineralogie und Werkstofftechnik [192]. Heute ist die Palette der Anwendungen praktisch nicht mehr überschaubar, wobei industrielle Qualitätskontrolle [30, 67], die Überwachung und Steuerung von Prozessen, Anlagen und Fahrzeugen bis hin zur Robotik [31] aus wirtschaftlicher Sicht ein besonderes Gewicht erhalten. Hinzu kommt, dass Bildgebung und Bildverarbeitung in vielen Systemen nicht mehr zu trennen sind. In einigen Fällen ist die Bildverarbeitung sogar integraler Bestandteil der Bildgebung. Man denke nur an moderne Fotografie, Computertomographie (CT) [258, 26, 117, 230], konfokale Laserscanningmikroskopie (CLSM) auf der Basis der Lichtblatttechnologie [122, 266, 160] (*light sheet fluorescence microscopy*, LSFM) oder die Ptychographie [155, 55].

Grundsätzlich lassen sich sehr viele Algorithmen der Verarbeitung und Analyse 2-dimensionaler Bilder auf 3-dimensionale (Volumen-)Bilder übertragen [199], die beispielsweise durch Tomographie oder CLSM erhalten werden. Für 3-dimensionale Bilder von Oberflächen, die mit aktiver Triangulation, mono-, bin- oder multiokularer Stereophotogrammetrie, mittels Lichtfeldkameras, durch Photoklinometrie (*shape from shading*, SFS) oder Autofokussensoren, mit Laufzeitmessungen (*time of flight*, TOF), Interferometrie, Shearographie und Holographie erzeugt werden [231], ist das allerdings nur sehr eingeschränkt möglich. Die Verarbeitung und Analyse solcher Bilder scheint derzeit mehr oder weniger eigenständige Wege zu gehen [39].

Mit diesen Entwicklungen einher geht, dass die Bildverarbeitung zu den zentralen Bestandteilen verschiedenster Fachgebiete gehört, die wiederum maßgeblich zu ihrer Entwicklung beigetragen haben. Dazu zählen insbesondere die Elektro- und Kommunikationstechnik [28, 80, 239] und die Informatik (von Computer-Vision [152] bis *geometric deep learning* [173]), aber auch die Mathematik (vor allem die Diskrete oder Digitale Geometrie [154, 153], die Differentialgeometrie [52], das Gebiet der Partiellen Differentialgleichungen [238, 203] und die Numerik [149]). Die Integralgeometrie mit dem exzellenten Buch von Rolf Schneider [243] (in einer

erweiterten Edition [244]) hat sich zu einer wichtigen Grundlage der Bildanalyse entwickelt, und Büchern über Stochastische Geometrie [175, 61, 245, 57, 139] und zufällige Felder [2, 3] können wertvolle Anregungen zur Analyse von Mikrostrukturen entnommen werden.

Für die Bildverarbeitung und Bildanalyse gibt es zahlreiche eigenständige, von der Art der Bildgebung weitgehend unabhängige und daher allgemein anwendbare Softwarepakete. Dazu gehören das System OpenCV (*Open Source Computer Vision Library*) [124], das C++-Toolkit Dlib [151] mit einem Fokus auf maschinelles Lernen, das in Java geschriebene und damit plattformübergreifende System ImageJ [242], das System Halcon [185] der Firma MVTec Software GmbH, die Software der Fa. Stemmer [4], das System ToolIP [88] des Fraunhofer-Instituts für Techno- und Wirtschaftsmathematik, verschiedene Produkte der Firma PixelFerber (Berlin) [211] für die Mikroskopbildverarbeitung, das Modul LabVIEW Vision im System LabVIEW der National Instruments AG [93] und die Programmbibliothek Caffe, die zahlreiche Algorithmen und Deep-Learning-Architekturen für die Klassifikation und Clusteranalyse von Bilddaten enthält [140]. Stärker an die Abbildungstechnik gebunden sind z. B. Softwareprodukte der Firmen Olympus Soft Imaging Solutions GmbH (Münster) und Carl Zeiss MicroImaging GmbH (München). Darüber hinaus sind umfangreiche Pakete für die Bildverarbeitung in die Systeme Python der Python Software Foundation [215], MatLab der Firma MathWorks [177] und IDL (*Interactive Data Language*) [113] integriert. Die Leistungsfähigkeit dieser Systeme, d. h. der Umfang der Algorithmen der Bildverarbeitung und Bildanalyse sowie die Qualität ihrer Implementierung, unterscheidet sich beträchtlich. Dabei scheint es schwerzufallen, ein durchgängiges Konzept hinsichtlich der Ausnutzung der Separabilität, der Berücksichtigung des zugrunde liegenden Gitters, der Wahl der Nachbarschaft der Pixel, der Randbehandlung in Bildern etc. zu wahren [159].

Aus den oben genannten Gründen können im vorliegenden Buch nur wenige Teilgebiete der Bildverarbeitung und Bildanalyse behandelt werden. Das Buch soll vielmehr als ein Lehrbuch verstanden werden, in dem eine Einführung in dieses sehr große Gebiet gegeben wird. Im Vordergrund stehen klassische Methoden der 2-dimensionalen Bildverarbeitung, wobei unter Wahrung der allgemeinen Verständlichkeit des Textes neuere Sichtweisen präsentiert werden. In nur wenigen Abschnitten werden mathematische Kenntnisse vorausgesetzt, die über ein Grundstudium hinausgehen. In didaktischer (und teilweise auch inhaltlicher) Hinsicht soll an den exzellenten Klassiker über morphologische Bildverarbeitung von Jean Serra [250], dem ehemaligen Direktor des Centre de Morphologie Mathématique in Fontainebleau, angeknüpft werden. Dem Lehrbuchcharakter wird durch zahlreiche, zum überwiegenden Teil sehr leicht nachvollziehbare Beispiele Rechnung getragen. Ergänzende Übungsaufgaben und deren im Anhang präsentierte Lösungen sollen es dem Leser erleichtern, sich in das Stoffgebiet einzuarbeiten. Außerdem werden einige Algorithmen in Form von Quellcode präsentiert, um dazu anzuregen, selbst zu programmieren und eigene Methoden zu implementieren. Zum vertiefenden Studium wird auf das bereits in mehrfacher Auflage erschienene Buch von Bernd Jähne [134, 136] (englischsprachige Fassung [135]), das dreibändige Werk ([48, 49]) von Wilhelm Burger und Mark Burge sowie das Handbuch [229] von John Russ und Brent Neal verwiesen. Eine mehr mathematische Behandlung des Themas ist in dem neueren Buch von Kristian Bredies und Dirk Lorenz [44] sowie in der zweibändigen Ausgabe von Jean-Charles Pinoli [209, 210] zu finden, siehe auch [202] zu Level-Set-Methoden und [282] zur Diffusionsfilterung. Die Bearbeitung von Farbbildern kann in [97] vertiefend nachgelesen werden, und auch zur Verarbeitung und Analyse von 3-dimensionalen (Volumen-)Daten gibt es umfangreiche weiterführende Literatur [171, 188, 267, 199]. Schließlich wird noch auf das Buch von Jürgen Beyerer u. a. [35]

hingewiesen, in dem eine ausgezeichnete Übersicht zur 3-dimensionalen Bildgebung profilierter oder gekrümmter Oberflächen enthalten ist.

Das vorliegende Buch ist wie folgt gegliedert: In Kapitel 1 werden einige Grundlagen der Bildverarbeitung behandelt. Dazu gehören homogene Gitter, auf denen kontinuierliche Bilder gesampelt werden, Pixel und ihre Nachbarschaften sowie der Wechsel des Gitters durch bilineare Interpolation der Pixeldaten. Einige für die Anwendung sehr wichtige Filter wie morphologische Transformationen, lineare und morphologische Filter sowie Rangordnungsfilter sind in Kapitel 2 beschrieben. Bildtransformationen wie das Labeling, die Distanz-, Wasserscheiden-, Radon- und Hough-Transformation werden in Kapitel 3 behandelt. Natürlich zählt auch die Fourier-Transformation zu den Bildtransformationen. Wegen ihrer großen Bedeutung ist ihr und ihren Anwendungen ein eigenständiges Kapitel gewidmet (Kapitel 4), wobei besonderes Augenmerk auf die vielfältigen wechselseitigen Beziehungen zwischen kontinuierlicher und diskreter Fourier-Transformation gelegt wird. Zu den Anwendungen der Fourier-Transformation gehören auch die lineare Filterung und die Korrelationsanalyse via Ortsfrequenzraum, die in Kapitel 5 behandelt werden. Das schließt die bildanalytische Bestimmung der Auto- und Kreuzkorrelationsfunktion von zufälligen Strukturen mit ein. Als weitere Anwendungen der Fourier-Transformation gelten die schnelle Radon-Transformation und ihre Inverse, die tomographische Rekonstruktion (Kapitel 6), wobei Algorithmen wie die gefilterte Rückprojektion zwar ohne Fourier-Transformation auskommen, bei deren Herleitung aber das Projektions-Schnitt-Theorem der Fourier-Transformation verwendet wird. Schließlich wird noch in Kapitel 7 auf einige Aspekte der digitalen Bildanalyse eingegangen, wobei vor allem auf integralgeometrische Ansätze zurückgegriffen wird.

Das Buch richtet sich an Studierende der Elektrotechnik (insbesondere der Automatisierungstechnik und Mechatronik), der Informatik, der Werkstofftechnik sowie der Optotechnik und Bildverarbeitung. Zudem wendet es sich an Entwicklerinnen und Entwickler von Bildverarbeitungssystemen sowie an Ingenieurinnen und Ingenieure, die sich mit dem Einsatz dieser Systeme im industriellen Umfeld befassen. Der Hanser Verlag ermöglicht auf seiner Internetseite plus.hanser-fachbuch.de den Zugriff auf eine kleine C-Bibliothek als ergänzendes Material, das Implementierungen der im Buch beschriebenen Methoden enthält. Außerdem wird dort ein Verzeichnis mit Quellcode zur Verfügung gestellt, der durch Mausklick auf die entsprechende Stelle im eBook geöffnet werden kann. Damit werden diese Methoden noch besser veranschaulicht, und es kann schnell und einfach mit Bilddaten experimentiert werden. Der Zugangscode zu dieser Internetseite ist auf Seite 1 in der Titelei dieses Buches abgedruckt.

Abschließend möchte ich mich bei allen bedanken, die auf die eine oder andere Art zu diesem Buch beigetragen haben, insbesondere bei meinen Kollegen Konrad Sandau und Udo Häberle von der Hochschule Darmstadt, bei dem Schüler Noah Rabe, bei meiner Ehefrau Renate Ohser-Wiedemann und nicht zuletzt bei Franziska Kaufmann, Christina Kubiak, Manuel Lepert und Frank Katzenmayer vom Carl Hanser Verlag für die sorgfältige redaktionelle Bearbeitung des Manuskripts. Mein Dank gilt auch Lars Thieme, Rainer Günzler, Dieter Horn und Uli Sonntag, mit denen ich in den 1980er Jahren die ersten Schritte in der Bildverarbeitung gemacht habe und von denen ich viele Anregungen zu diesem Buch erhalten habe, vor allem zum letzten Kapitel.

Inhalt

| | | |
|----------|--|-----------|
| 1 | Gitter, Bilder und Nachbarschaften | 13 |
| 1.1 | Vorbemerkungen zur Bilddatenstruktur | 13 |
| 1.2 | Euler-Zahl | 14 |
| 1.2.1 | Additive Erweiterung | 14 |
| 1.2.2 | Euler-Poincaré-Formel | 17 |
| 1.2.3 | Netzwerkformel | 18 |
| 1.3 | Homogene Gitter, Sampling und Digitalisierung | 19 |
| 1.4 | Lokale Pixelkonfigurationen | 23 |
| 1.5 | Nachbarschaften von Pixeln und ihre Komplementarität | 26 |
| 1.6 | Digitale Bilder | 29 |
| 1.6.1 | Grautonbilder | 29 |
| 1.6.2 | Interpolation von Pixelwerten | 31 |
| 1.6.2.1 | Bilddrehung | 33 |
| 1.6.2.2 | Verzeichnungskorrektur | 34 |
| 1.6.3 | Lokale Pixeloperationen | 40 |
| 1.6.3.1 | Binarisierung von Grautonbildern | 42 |
| 1.6.3.2 | Manipulation des Grauwertistogramms | 44 |
| 1.6.4 | Elementare Statistik für Pixelwerte | 46 |
| 1.6.5 | Mehrkanalige Bilder | 47 |
| 1.6.5.1 | RGB- und HSV-Farbraum | 48 |
| 1.6.5.2 | Hauptkomponenten in mehrkanaligen Bildern | 51 |
| 1.6.6 | Bildrandfehler und allgemeine Prinzipien ihrer Korrektur | 52 |
| 1.6.7 | Bildrauschen | 54 |
| 2 | Filterung von Bildern | 57 |
| 2.1 | Morphologische Transformationen | 57 |
| 2.1.1 | Minkowski-Addition und Dilatation | 57 |
| 2.1.2 | Minkowski-Subtraktion und Erosion | 60 |
| 2.1.3 | Morphologische Öffnung und Abschließung | 62 |
| 2.1.4 | Top-Hat-Transformationen | 65 |
| 2.1.5 | Algorithmische Implementierung | 66 |
| 2.1.6 | Bildrandfehler morphologischer Transformationen | 68 |

| | | |
|----------|--|------------|
| 2.2 | Lineare Filter | 70 |
| 2.2.1 | Lineare Glättungsfilter | 73 |
| 2.2.1.1 | Mittelwertfilter | 73 |
| 2.2.1.2 | Gauß- und Binomialfilter | 76 |
| 2.2.2 | Ableitungsfilter 1. Ordnung | 80 |
| 2.2.3 | Ableitungsfilter 2. Ordnung | 87 |
| 2.3 | Morphologische Filter | 90 |
| 2.3.1 | Von Transformation zu Filterung | 90 |
| 2.3.2 | Algorithmische Implementierung | 95 |
| 2.4 | Rangordnungsfilter | 95 |
| 2.4.1 | Diskrete Versionen von Rangordnungsfiltern | 97 |
| 2.4.2 | Hinweise zur algorithmischen Implementierung | 98 |
| 3 | Spezielle Bildtransformationen | 99 |
| 3.1 | Labeling von Zusammenhangskomponenten | 99 |
| 3.1.1 | Verbundenheit und Zusammenhangskomponenten | 100 |
| 3.1.2 | Elementarer Labeling-Algorithmus | 102 |
| 3.1.3 | Labeling mit Lauflängenkodierung | 108 |
| 3.2 | Distanztransformation | 112 |
| 3.2.1 | Definition und Bezeichnungen | 112 |
| 3.2.2 | Weitere Distanztransformationen | 114 |
| 3.2.3 | Algorithmische Implementierung | 115 |
| 3.2.3.1 | Der 1-dimensionale Fall | 117 |
| 3.2.3.2 | Der 2-dimensionale Fall | 118 |
| 3.3 | Wasserscheidentransformation | 120 |
| 3.3.1 | Geodätischer Abstand | 122 |
| 3.3.2 | Zerlegung in Einflusszonen | 124 |
| 3.3.3 | Flutungsalgorithmus für die Wasserscheidentransformation | 127 |
| 3.4 | Radon- und Hough-Transformation | 129 |
| 3.4.1 | Radon-Transformation | 130 |
| 3.4.2 | Hough-Transformation | 140 |
| 3.4.3 | Template-Matching | 145 |
| 4 | Fourier-Transformation | 148 |
| 4.1 | Kontinuierliche Fourier-Transformation | 150 |
| 4.2 | Fourier-Bessel-Transformation | 159 |

| | | |
|----------|---|------------|
| 4.3 | Anwendungen | 163 |
| 4.3.1 | Ortssensitive Diffusionsfilter | 163 |
| 4.3.2 | Abtasttheorem und Moiré-Effekt | 168 |
| 4.4 | Diskrete Fourier-Transformation..... | 172 |
| 4.4.1 | Die 1-dimensionale diskrete Fourier-Transformation | 174 |
| 4.4.2 | Schnelle Fourier-Transformation | 180 |
| 4.4.3 | Die 2-dimensionale diskrete Fourier-Transformation | 184 |
| 4.5 | Abel-Transformation | 189 |
| 5 | Faltung und Korrelation im Ortsfrequenzraum | 191 |
| 5.1 | Faltung im Ortsfrequenzraum | 191 |
| 5.2 | Transferfunktionen linearer Filter | 201 |
| 5.2.1 | Transferfunktionen von Binomialfiltern | 203 |
| 5.2.2 | Transferfunktionen von Mittelwertfiltern | 206 |
| 5.2.3 | Transferfunktion von Gauß-Filtern | 207 |
| 5.2.4 | Transferfunktion des Gradientenfilters..... | 210 |
| 5.2.5 | Transferfunktion des Laplace-Filtern | 211 |
| 5.3 | Filterdesign | 212 |
| 5.3.1 | Design von Gradientenfiltern zur Messung von Richtungen | 212 |
| 5.3.2 | Verbesserung der Isotropieeigenschaften von Laplace-Filtern | 214 |
| 5.4 | Tief-, Hoch- und Bandpassfilter | 214 |
| 5.4.1 | Tiefpassfilter | 215 |
| 5.4.2 | Hochpassfilter | 216 |
| 5.4.3 | Bandpassfilter | 217 |
| 5.5 | Inverse Filterung | 218 |
| 5.6 | Auto- und Kreuzkorrelationsfunktionen zufälliger Strukturen | 221 |
| 5.6.1 | Korrelation und Spektraldichte..... | 222 |
| 5.6.2 | Wolkigkeit von Papier..... | 227 |
| 5.6.3 | Kreuzkorrelationsfunktion und ihre Schätzung | 230 |
| 5.6.4 | Über die Ausbreitung des Borkenkäfers | 231 |
| 6 | Radon-Transformation und tomographische Rekonstruktion | 234 |
| 6.1 | Radon-Transformation via Ortsfrequenzraum | 234 |
| 6.2 | Tomographische Rekonstruktion | 237 |
| 6.2.1 | Gefilterte Rückprojektion | 240 |
| 6.2.2 | Algorithmische Implementierung..... | 241 |

| | | |
|----------|---|------------|
| 7 | Grundbegriffe der Bildanalyse | 245 |
| 7.1 | Additive, translationsinvariante, isotrope und stetige Merkmale | 245 |
| 7.1.1 | Messung der Fläche | 248 |
| 7.1.2 | Messung des Umfangs | 250 |
| 7.2 | Momente eines Objekts | 257 |
| 7.3 | Konvexe Hülle und ihre Merkmale | 259 |
| 7.3.1 | Die Breite eines Objekts | 259 |
| 7.3.2 | Algorithmische Bestimmung der konvexen Hülle | 261 |
| 7.3.3 | Die Stützfunktion eines Objekts | 263 |
| 7.3.4 | Implementierung der Messung von Objektmerkmalen | 266 |
| 7.4 | Merkmale zufälliger Strukturen | 266 |
| 7.4.1 | Der Flächenanteil, die spezifische Randlänge und die Euler-Zahl pro Flächeneinheit | 267 |
| 7.4.2 | Verteilungen der Merkmale von Objekten | 272 |
| 7.4.3 | Industrielle Standards für die Bildanalyse | 276 |
| 8 | Lösung der Übungsaufgaben | 284 |
| | Formelzeichen und Abkürzungen | 293 |
| | Literatur | 294 |
| | Index | 309 |

1

Gitter, Bilder und Nachbarschaften

Im folgenden Kapitel sollen einige Grundlagen der Bildverarbeitung und Bildanalyse behandelt werden, die für die Behandlung in den darauf folgenden Kapiteln wichtig sind. Dazu gehört die Einführung digitaler Bilder als ein Sampling von Funktionen auf Gittern. Für viele Algorithmen der Bildverarbeitung und Bildanalyse von Binärbildern werden zudem Nachbarschaften der Pixel vorausgesetzt, die ebenfalls in diesem Kapitel erklärt werden sollen.

■ 1.1 Vorbemerkungen zur Bilddatenstruktur

Um künftig etwas Quellcode zur Notation von Algorithmen in der Programmiersprache C präsentieren zu können, verständigen wir uns darauf, dass ein Bild `img` eine Struktur `IMG` mit der inhaltlichen Beschreibung `*description`, der Pixelzahl `*n`, der Pixelgröße `*a`, dem Datentyp `t` der Pixelwerte und den Pixelwerten `**pix` hat, siehe `Image.h`. Dabei ist `**pix` eine rechteckige Matrix von Pixelwerten mit $n_1 = \text{img.n}[0]$ Zeilen und $n_2 = \text{img.n}[1]$ Spalten, wobei die Allokation des Speicherplatzes in geeigneter Weise vorzunehmen ist (`Malloc()`) mit den Makros `Malloc.h`. Die Pixelgrößen $a_1 = \text{img.a}[0]$ und $a_2 = \text{img.a}[1]$ entsprechen den Gitterabständen rechteckiger Gitter. Ihre Integration in die Bildstruktur ist sinnvoll, da sich die Pixelgröße durch einige Bildverarbeitungsschritte ändert (z. B. bei der Fourier-Transformation) und durch Bildanalyse in der Regel skalierte Kenngrößen (z. B. die Fläche eines Objekts) erhalten werden. Der Daten der Pixelwerte können für $\text{img.t} = 1, 2, \dots$ vom Typ `unsigned char`, `unsigned short`, `unsigned long`, `float`, `double` etc. seien. Die Pixelwerte von Binärbildern ($\text{img.t} = 0$) seien ebenfalls vom Typ `unsigned char`, wobei für Vordergrundpixel lediglich das erste Bit belegt ist. Für Pixel mehrkanaliger Bilder und komplexwertiger Pixel werden später an den entsprechenden Stellen gesonderte Vereinbarungen getroffen werden.

Für eine übersichtlichere Gestaltung von Quelltexten wurde die Struktur `IMG` noch durch die Gesamtzahl `npix` der Pixel des Bildes, der Pixelfläche `apix` und dem Zeiger `str` auf den Datenstream ergänzt; diese Ergänzungen sind aber redundant. Sind also die Pixelwerte `**pix` z. B. vom Typ `unsigned char` und setzen wir einen Datenstream `*str`, der ebenfalls vom Typ `unsigned char` sein muss, mit `str = &((unsigned char **)img->pix)[0][0]` auf die Adresse des ersten Pixels, dann sind die Pixelwerte `pix[i][j]` und `*(str + i * n[0] + j)` für alle Indizes identisch. Formal schreiben wir

$$\text{pix}[i][j] \equiv *(str + i * n[0] + j) \quad (1.1)$$

für $i = 0 \dots n[0]-1$ und $j = 0 \dots n[1]-1$, was bedeutet, dass die Pixelwerte auf dem Datenstream `y`-lokal abgelegt sind. Damit könnte die Invertierung eines 8-Bit-Gräutonbildes beispielsweise mit einer Doppelschleife (`InvertImg0()`) oder kürzer und übersichtlicher mit einer

Einfachschleife (`InvertImg()`) geschrieben werden. Wir werden später von beiden Varianten der Pixeladressierung Gebrauch machen. Zudem wird in diesem Buch noch eine Funktion `NewImg()` zur Erzeugung eines Bildes mit einer vorgegebenen Pixelzahl, einer Pixelgröße und eines Datentyps für die Pixelwerte verwendet.

Natürlich müsste in allen Funktionen noch eine sorgfältige Fehlerbehandlung vorgenommen werden, auf die wir hier aber aus Platz- und Übersichtsgründen verzichten wollen.

■ 1.2 Euler-Zahl

Die Euler-Zahl (Euler-Poincaré-Charakteristik) ist zunächst ein Merkmal (*feature*) eines Binärbildes, das algorithmisch sehr einfach bestimmt werden kann und mit dem sich eine Reihe von Problemen der Bildanalyse lösen lässt. In diesem Abschnitt wird die Euler-Zahl jedoch primär dazu eingeführt, um die Bedeutung von Nachbarschaften der Pixel eines Binärbildes und die Komplementarität von Nachbarschaften zu erklären.

Wir betrachten den kontinuierlichen Fall, in dem der Vordergrund und der Hintergrund eines Binärbildes als Teilmengen des 2-dimensionalen Euklidischen Raumes \mathbb{R}^2 aufgefasst werden können. Außerdem nehmen wir einfachheitshalber an, dass das Bild unendlich ausgedehnt ist. Das erspart uns die Behandlung von Bildrandeffekten, auf die aber später noch einzugehen ist.

Wenn wir mit $X \subseteq \mathbb{R}^2$ den Vordergrund eines kontinuierlichen Binärbildes bezeichnen, dann ist die Komplementärmenge $X^c = \mathbb{R}^2 \setminus X$ der Hintergrund, wobei \setminus die Mengendifferenz ist, die durch $X \setminus Y = \{x \in \mathbb{R}^2 : x \in X, x \notin Y\}$ für $X, Y \subseteq \mathbb{R}^2$ definiert wird. Die Euler-Zahl wird zunächst für kompakte (d. h. beschränkte und abgeschlossene), konvexe Mengen $K \subset \mathbb{R}^2$ erklärt.

Definition 1.1 Sei $K \subset \mathbb{R}^2$ eine kompakte und konvexe Menge. Die Euler-Zahl $\chi(K)$ der Menge K ist definiert durch

$$\chi(K) = \begin{cases} 1, & \text{falls } K \neq \emptyset \\ 0, & \text{sonst} \end{cases} .$$

Das heißt, $\chi(K)$ nimmt für kompakte, konvexe Mengen nur die Werte 0 oder 1 an. Aus der Einfachheit dieser Definition erklärt sich letztendlich auch die Einfachheit der bildanalytischen Bestimmung. Allerdings bedeutet die Einschränkung auf konvexe Mengen eine wesentliche Einschränkung der Anwendung, denn es kann im Allgemeinen nicht vorausgesetzt werden, dass der Vordergrund eines Binärbildes oder auch nur eine Zusammenhangskomponente des Vordergrundes konvex sind. Wir gehen daher zu nichtkonvexen Mengen über.

1.2.1 Additive Erweiterung

- Die Vereinigung $X = K_1 \cup K_2$ zweier kompakter, konvexer Mengen K_1 und K_2 ist im Allgemeinen nicht konvex. Allerdings ist der Durchschnitt $K_1 \cap K_2$ konvex. Daher kann die Euler-Zahl von X durch

$$\chi(X) = \chi(K_1) + \chi(K_2) - \chi(K_1 \cap K_2)$$

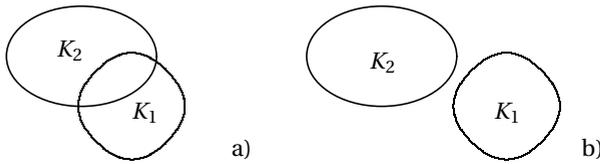


Bild 1.1 Die Vereinigung $X = K_1 \cup K_2$ zweier konvexer Mengen K_1 und K_2 ist im Allgemeinen nicht konvex. a) Der Durchschnitt $K_1 \cap K_2$ ist nicht leer, und folglich ist $\chi(X) = 1$. b) Der Durchschnitt ist leer, und damit ist $\chi(X) = 2$.

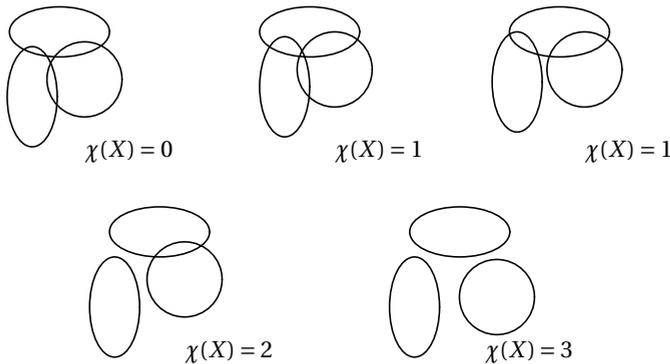


Bild 1.2 Werte für die Euler-Zahl der Vereinigung $X = K_1 \cup K_2 \cup K_3$ dreier konvexer Mengen K_1 , K_2 und K_3

erklärt werden. Sind K_1 und K_2 nicht leer, aber disjunkt, dann ist $\chi(X) = 2$. Ist der Durchschnitt $K_1 \cap K_2$ ebenfalls nicht leer (d. h. bildet $K_1 \cup K_2$ eine Zusammenhangskomponente), dann ist $\chi(X) = 1$.

In Bild 1.1 sind die Mengen K_1 und K_2 schematisch dargestellt.

- Für die Vereinigung $X = K_1 \cup K_2 \cup K_3$ dreier Mengen gilt

$$\begin{aligned} \chi(X) &= \chi(K_1) + \chi(K_2) + \chi(K_3) \\ &\quad - \chi(K_1 \cap K_2) - \chi(K_1 \cap K_3) - \chi(K_2 \cap K_3) \\ &\quad + \chi(K_1 \cap K_2 \cap K_3). \end{aligned}$$

Ist X die Vereinigung dreier konvexer Mengen, dann kann die Euler-Zahl von X Werte zwischen 0 und 3 annehmen, siehe Bild 1.2.

- Das lässt sich verallgemeinern. Für eine endliche Vereinigung $X = \bigcup_{i=1}^m K_i$ gilt

$$\chi(X) = \sum_{i=1}^m \chi(K_i) - \sum_{i=1}^{m-1} \sum_{j=i+1}^m \chi(K_i \cap K_j) + \dots - (-1)^m \chi\left(\bigcap_{i=1}^m K_i\right) \quad (1.2)$$

(Inklusions-Exklusions-Prinzip), wobei anzumerken ist, dass sich jede kompakte, nicht-konvexe Menge hinreichend gut durch eine endliche Vereinigung kompakter konvexer Mengen approximieren lässt (wobei wir hier offen lassen, was „approximieren“ bedeutet).

Wegen der Additivität der Euler-Zahl und des sich daraus ergebenden Inklusions-Exklusions-Prinzips lässt sich die Euler-Zahl lokaler Bildinformation bestimmen.

Die Euler-Zahl ist (im 2-dimensionalen Fall) die Anzahl der Zusammenhangskomponenten minus die Anzahl der „Löcher“.

Aufgabe 1.1 Welche Werte kann die Euler-Zahl der Vereinigung von vier konvexen Mengen annehmen? ■

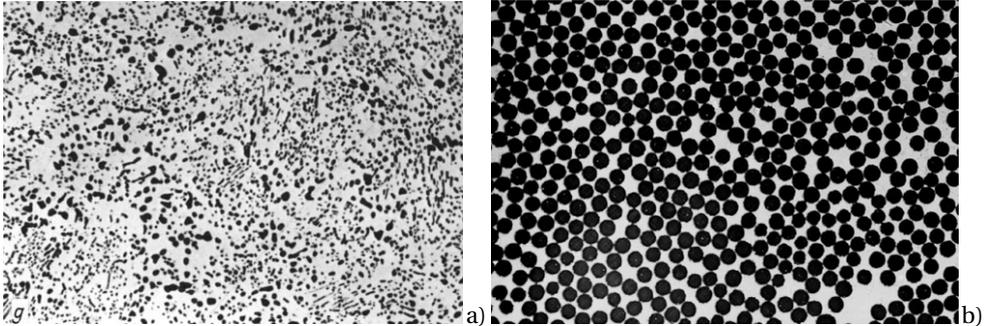


Bild 1.3 a) Zementitausscheidungen (dunkel) in einem eutektoidischen Kohlenstoffstahl in einem ebenen Anschliff, auflichtoptische Aufnahme, 664×948 Pixel, Pixelgröße $a = 0,196 \mu\text{m}$; b) Verbundwerkstoff, Kohlenstofffasern (dunkel) in Epoxidharz im Querschliff, auflichtoptische Aufnahme, 664×948 Pixel der Größe $a = 0,139 \mu\text{m}$

Beispiel 1.1 Einfache Zusammenhangskomponenten (einfach zusammenhängende Objekte, Objekte ohne Loch) haben die Euler-Zahl 1. Folglich kann die Euler-Zahl verwendet werden, um einfache Zusammenhangskomponenten zu zählen. Die Zementitausscheidungen in Bild 1.3a sind einfach zusammenhängend. Also entspricht in diesem Fall die gemessene Euler-Zahl χ der Anzahl N der Ausscheidungen: $\chi = 2068$ für die 4er-Nachbarschaft, $\chi = 2002$ für die 6er-Nachbarschaft und $\chi = 1935$ für die 8er-Nachbarschaft (bei einer Binarisierungsschwelle von 128). Meist wird die Anzahl auf die Größe des Bildausschnittes bezogen, d. h., es wird die Anzahl pro Flächeneinheit angegeben, $N_A = 80\,000 \text{ mm}^{-2}$ für die 8er-Nachbarschaft. ■

Beispiel 1.2 Klar, die Anzahl m der (sich berührenden) Kohlenstofffaserquerschnitte in Bild 1.3b kann bestimmt werden, indem die Objekte in einem binarisierten Bild zunächst durch eine morphologische Transformation getrennt und danach gezählt werden, wobei für die Zählung wie oben die Euler-Zahl verwendet werden kann. ■

Beispiel 1.3 Schwieriger scheint es zu sein, in Bild 1.3b die Anzahl n der Faserkontakte (d. h. der Berührungsstellen der Objekte) zu bestimmen. Wir verwenden die Inklusions-Exklusions-Formel (1.2), wobei zu berücksichtigen ist, dass ausschließlich paarweise Überlappungen auftreten. Die Gl. (1.2) kann also vereinfacht werden,

$$\chi(X) = \underbrace{\sum_{i=1}^m \chi(K_i)}_m - \underbrace{\sum_{i=1}^{m-1} \sum_{j=i+1}^m \chi(K_i \cap K_j)}_n,$$

wobei man sich leicht überlegen kann, dass die Doppelsumme auf der rechten Seite gerade der Anzahl n der Faserkontakte entspricht. Ein Wert für n kann bestimmt werden, wenn neben der Anzahl m der Faserquerschnitte auch die Euler-Zahl der (nicht erodierten) Struktur gemessen und die obige Gleichung nach n umgestellt wird,

$$n = m - \chi(X).$$

Die mittlere Anzahl κ der Faserkontakte pro Faser (d. h. die Koordinationszahl) erhält man aus

$$\kappa = 2 \left(1 - \frac{\chi(X)}{m} \right).$$

Der Faktor 2 ist dadurch motiviert, dass sich jeweils zwei einander berührende Objekte eine Berührungsstelle „teilen“. Für Bild 1.3b erhält man $\chi = -227$ (bezüglich der 6.1er-Nachbarschaft), $m = 471$ und damit $\kappa = 1,036$. ■

Bemerkung 1.1 Ist X eine endliche Vereinigung konvexer Mengen, dann gilt das nicht für ihre Komplementärmenge X^c . Folglich lässt sich die Euler-Zahl von X^c nicht mithilfe des Inklusions-Exklusions-Prinzips erklären. In diesem Fall kann $\chi(X^c)$ durch Hadwigers rekursive Formel definiert werden [108], [193], und es gilt

$$\chi(X) = -\chi(X^c),$$

d. h., durch die Invertierung eines Binärbildes ändert sich das Vorzeichen der Euler-Zahl. ■

1.2.2 Euler-Poincaré-Formel

Für (nichtkonvexe) polygonale Mengen, d. h. für endliche Vereinigungen konvexer Polygone, kann die Euler-Zahl mit der Euler-Poincaré-Formel berechnet werden.

Satz 1.1 Sei X ein (nicht notwendig konvexes oder zusammenhängendes) Polygon mit v Ecken (*vertices*), e Kanten (*edges*) und f konvexen Flächen (*faces*). Für die Euler-Zahl $\chi(X)$ des Polygons gilt

$$\chi(X) = v - e + f \tag{1.3}$$

(Euler-Poincaré-Formel). ■

Beispiel 1.4 Für ein konvexes n -Eck X ist $v = e = n$ und $f = 1$. Somit ist $\chi(X) = n - n + 1 = 1$. ■

Beispiel 1.5 Für die polygonale Menge X in Bild 1.4b ist $v = 15$, $e = 20$ und $f = 5$. Damit erhält man $\chi(X) = 15 - 20 + 5 = 0$. Das Ergebnis ist plausibel, denn in Bild 1.4a ist eine Zusammenhangskomponente mit einem „Loch“ abgebildet. ■

Die Euler-Poincaré-Formel wird in diesem Kapitel vor allem für die bildanalytische Bestimmung der Euler-Zahl gebraucht.

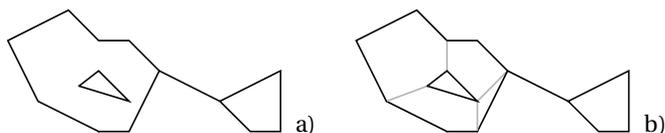


Bild 1.4 a) Eine zusammenhängende polygonale Menge X mit einem „Loch“: Die Euler-Zahl von X ist erwartungsgemäß gleich 0. b) Durch die eingefügten Kanten erhält man eine Unterteilung in konvexe Polygone, deren Vereinigung X ist.

1.2.3 Netzwerkformel

Für viele Anwendungen ist die sogenannte Netzwerkformel hilfreich, die etwas über den Zusammenhang zwischen der Euler-Zahl pro Flächeneinheit χ_A und der Anzahl der Knoten pro Flächeneinheit N_A in einem Netzwerk (d. h. in einem Graphen) aussagt. Wir bezeichnen mit $\bar{\mu}$ die mittlere Anzahl der Kanten des Netzwerks, die von einem Knoten ausgehen. Mit $\bar{\mu}$ wird die mittlere Ordnung der Knoten bezeichnet, also die mittlere Anzahl der Kanten, die von einem Knoten ausgehen. Dann gilt

$$\chi_A = N_A \left(1 - \frac{\bar{\mu}}{2} \right) \quad (1.4)$$

(Netzwerkformel) [178].

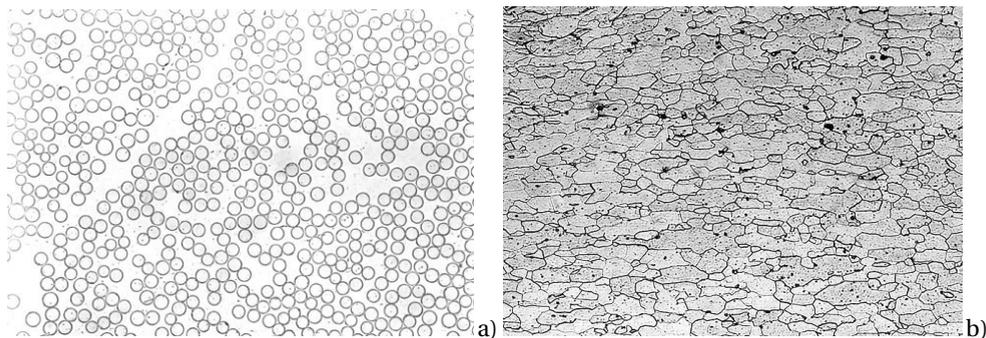


Bild 1.5 a) Glasfasern in Epoxidharz, Querschliff, auflichtmikroskopische Aufnahme, Dunkelfeld, 664×948 Pixel der Größe $a = 0,139 \mu\text{m}$ und b) ferritischer Stahl, Längsschliff eines Drahts, auflichtmikroskopische Aufnahme, Hellfeld, 768×1024 Pixel der Größe $a = 0,451 \mu\text{m}$. Die sich berührenden Kreislinien der Ränder der Glasfasern bzw. das Kantensystem der Ferritkörner bilden Netzwerke.

Beispiel 1.6 Wir fassen die Berührungstellen der Glasfasern in Bild 1.5a als Knoten eines Netzwerkes auf. Von jedem dieser Knoten gehen vier Kanten aus, und folglich ist $\bar{\mu} = 4$. Vorausgesetzt, das Netzwerk lässt sich segmentieren, dann wird die Euler-Zahl pro Flächeneinheit χ_A bestimmt und die Netzwerkformel (1.4) nach der Anzahl der Berührungstellen pro Flächeneinheit N_A umgestellt. Man erhält unmittelbar

$$N_A = -\chi_A.$$



Beispiel 1.7 Für das Kantensystem in Bild 1.5b ist $\bar{\mu} = 3$. Die Anzahl der Knoten (Kornzwickel) je Flächeneinheit N_V lässt sich folglich mithilfe der Netzwerkformel

$$N_A = -2\chi_A \quad (1.5)$$

berechnen, sofern sich das Kantensystem vernünftig segmentieren lässt und χ_A mit der erforderlichen Genauigkeit bestimmt werden kann. ■

■ 1.3 Homogene Gitter, Sampling und Digitalisierung

Digitale Bilder sind Daten auf Gitterpunkten (*points*, pt), wobei in der Sprache der Bildverarbeitung die Daten den Pixelwerten und die Gitterpunkte den Pixelpositionen entsprechen. Die Menge der Gitterpositionen eines Bildes bildet ein 2-dimensionales Gitter \mathbb{L} . Eine besondere Rolle spielen in der Bildverarbeitung homogene Gitter.

Definition 1.2 Ein Gitter \mathbb{L} heißt homogen, wenn es invariant bezüglich Gitterverschiebungen ist, d. h. $\mathbb{L} + x = \mathbb{L}$ für alle $x \in \mathbb{L}$. ■

Seien $u_1, u_2 \in \mathbb{R}^2$ zwei linear unabhängige Vektoren und \mathbb{Z} die Menge der ganzen Zahlen. Dann bildet die Menge

$$\mathbb{L} = \{x \in \mathbb{R}^2 : x = iu_1 + ju_2, i, j \in \mathbb{Z}\}$$

der ganzzahligen Linearkombinationen der Basisvektoren u_1 und u_2 ein homogenes Gitter mit der Einheitszelle $C = \{x \in \mathbb{R}^2 : x = pu_1 + qu_2, 0 \leq p, q \leq 1\}$, d. h., C hat die Eckpunkte $0, u_1, u_2$ und $u_1 + u_2$. Die Kantenlängen $a_1 = \|u_1\|$ und $a_2 = \|u_2\|$ sind die Gitterabstände, die den Pixelgrößen entsprechen. Fassen wir die beiden Basisvektoren u_1 und u_2 zu einer Matrix $U = (u_1, u_2)$ zusammen, dann ist $\text{Fläche}(C) = |\det U|$ die Fläche der Einheitszelle C , die häufig mit der Pixelgröße eines digitalen Bildes assoziiert wird. Homogene Gitter sind unbeschränkt; in Bild 1.6a wird lediglich ein Ausschnitt gezeigt.

Das Gitter \mathbb{L} ist durch seine Basisvektoren charakterisiert, jedoch sind für ein gegebenes homogenes Gitter die Basisvektoren nicht eindeutig bestimmt. Für ein inhomogenes Gitter, siehe Bild 1.6b, gibt es keine Basis. Ein inhomogenes Gitter ist durch eine Basis nicht charakterisierbar, weshalb solche Gitter für die Bildverarbeitung ungeeignet sind.

Beispiel 1.8 Quadratische Gitter $\mathbb{L} = a\mathbb{Z}^2$ mit einem Gitterabstand $a > 0$ werden z. B. in CCD- und CMOS-Kameras zugrunde gelegt, Bild 1.7a. Die gebräuchliche Gitterbasis ist

$$u_1 = \begin{pmatrix} a \\ 0 \end{pmatrix}, \quad u_2 = \begin{pmatrix} 0 \\ a \end{pmatrix}.$$

Alternativ könnte aber auch die Basis

$$u_1 = \begin{pmatrix} a \\ 0 \end{pmatrix}, \quad u_2 = \begin{pmatrix} a \\ a \end{pmatrix}$$

gewählt werden. ■

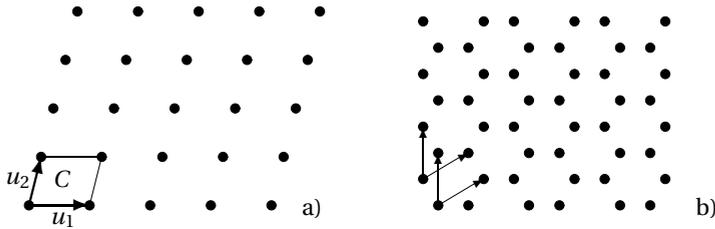


Bild 1.6 a) Ausschnitt aus einem homogenen 2-dimensionalen Gitter \mathbb{L} mit 25 Punkten (25 pt), den Basisvektoren u_1 und u_2 und der Einheitszelle C ; b) Beispiel für ein inhomogenes Gitter als Superposition zweier homogener Gitter

Aufgabe 1.2 Häufig wird die laterale Auflösung in der Maßeinheit Pixel pro inch (*dots per inch*, dpi) angegeben. Welchen Wert hat die Pixelgröße a eines quadratischen Gitters bei einer lateralen Auflösung von 600 dpi? ■

Beispiel 1.9 Zeilenkameras liefern meist Bilder auf rechteckigen Gittern, die z. B. durch die Basisvektoren

$$u_1 = \begin{pmatrix} a_1 \\ 0 \end{pmatrix}, \quad u_2 = \begin{pmatrix} 0 \\ a_2 \end{pmatrix}$$

charakterisiert sind. In Bild 1.7b ist das Verhältnis der Seitenlängen $a_1/a_2 = 1,5$. Die Pixelgröße a_1 entspricht der lateralen Auflösung in Zeilenrichtung, und a_2 ist von der Relativbewegung abzubildener Objekte orthogonal zur Zeilenrichtung abhängig (also z. B. von der Geschwindigkeit eines Förderbandes, über dem die Zeilenkamera angebracht ist). ■

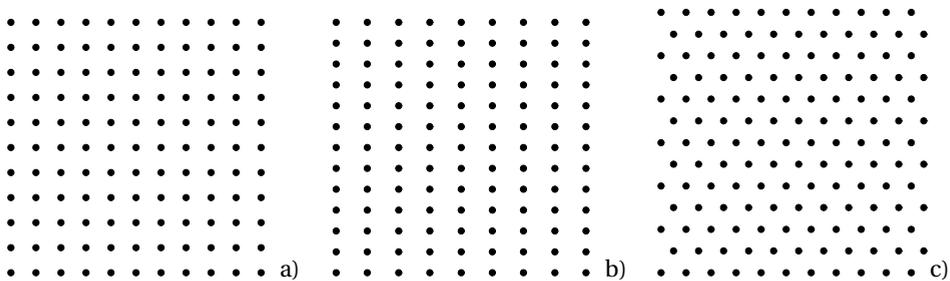


Bild 1.7 Ausschnitte homogener Gitter: a) quadratisch, b) rechteckig, c) hexagonal

Beispiel 1.10 Ein Gitter mit der Basis

$$u_1 = \begin{pmatrix} a \\ 0 \end{pmatrix}, \quad u_2 = \frac{1}{2} \begin{pmatrix} a \\ \sqrt{3}a \end{pmatrix}$$

und dem einheitlichen Gitterabstand $a > 0$ ist hexagonal, siehe Bilder 1.7c und 1.8a. In den 1970er Jahren wurden von der Firma Leitz Kameras mit hexagonalen Pixelrastern eingeführt, was für die Bildverarbeitung und Bildanalyse Vorteile im Vergleich zu

quadratischen Rastern hat. Die Grundlagen dafür gehen auf Entwicklungen des Centre de Morphologie Mathématique in Fontainebleau zurück, siehe [250] und darin zitierte Literatur. ■

Bei der Fourier-Transformation von Bildern spielt das inverse Gitter eine Rolle, das im Zusammenhang mit der Beugung von Röntgenstrahlen an Kristallgittern auch als reziprokes Gitter bezeichnet wird. Die Matrix \hat{U} der Gitterbasis des zu \mathbb{L} inversen Gitters $\hat{\mathbb{L}}$ ist die Transponierte der Inversen von U ,

$$\hat{U} = (U^{-1})',$$

wobei die Spaltenvektoren \hat{u}_1 und \hat{u}_2 von \hat{U} die Basis des inversen Gitters bilden, siehe auch Abschnitt 4.4.3. Offensichtlich haben die Gitterabstände $\|\hat{u}_1\|$ und $\|\hat{u}_2\|$ von $\hat{\mathbb{L}}$ die Maßeinheit m^{-1} , wenn die Maßeinheit der Gitterabstände $\|u_1\|$ und $\|u_2\|$ in m gegeben ist. Für die Pixelgröße des inversen Gitters mit der Einheitszelle \hat{C} gilt $F(\hat{C}) = 1/F(C)$.

Beispiel 1.11 Das zum hexagonalen Gitter \mathbb{L} inverse Gitter $\hat{\mathbb{L}}$ ist ebenfalls hexagonal. Aus der in Beispiel 1.10 gegebenen Basis folgt

$$U = \frac{a}{2} \begin{pmatrix} 2 & 1 \\ 0 & \sqrt{3} \end{pmatrix}, \quad U^{-1} = \frac{1}{\sqrt{3}a} \begin{pmatrix} \sqrt{3} & -1 \\ 0 & 2 \end{pmatrix}, \quad \hat{U} = \frac{1}{\sqrt{3}a} \begin{pmatrix} \sqrt{3} & 0 \\ -1 & 2 \end{pmatrix}.$$

Die Basisvektoren \hat{u}_1 und \hat{u}_2 des inversen Gitters sind also

$$\hat{u}_1 = \frac{1}{\sqrt{3}a} \begin{pmatrix} \sqrt{3} \\ -1 \end{pmatrix}, \quad \hat{u}_2 = \frac{1}{\sqrt{3}a} \begin{pmatrix} 0 \\ 2 \end{pmatrix}.$$

In Bild 1.8 sind die beiden zueinander inversen Gitter \mathbb{L} und $\hat{\mathbb{L}}$ dargestellt. Die Pixelgröße von \mathbb{L} ist $a_1 = a_2 = a$, die des inversen Gitters ist $a_1 = a_2 = \frac{2}{\sqrt{2}a}$. ■

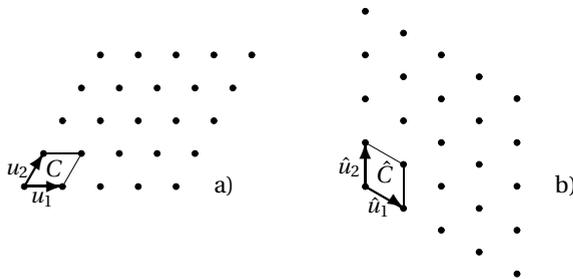


Bild 1.8 Ausschnitte a) aus dem hexagonalen Gitter \mathbb{L} und b) dem zugehörigen inversen Gitter $\hat{\mathbb{L}}$

Aufgabe 1.3 Gegeben sei ein Gitter \mathbb{L} mit den Basisvektoren

$$u_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad u_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

Bestimmen Sie eine Basis des zu \mathbb{L} inversen Gitters $\hat{\mathbb{L}}$. ■

Aufgabe 1.4 Unter welcher Voraussetzung ist $\mathbb{L} = \hat{\mathbb{L}}$?


Als einfachstes Modell der Abtastung wird das Sampling $X_{\square} = X \cap \mathbb{L}$ einer Menge $X \subset \mathbb{R}^2$ auf einem homogenen Gitter \mathbb{L} eingeführt. Es besteht aus allen Gitterpunkten, die in X liegen, Bild 1.9b. Unter $X \cap \mathbb{L}$ kann die Menge der Vordergrundpixel eines digitalen Binärbildes verstanden werden, und das Sampling $X^c \cap \mathbb{L}$ der Komplementärmenge X^c ist die Menge der Hintergrundpixel, Bild 1.9c. Sampling ist in der Regel mit einem Informationsverlust verbunden. Es ist also im Allgemeinen nicht möglich, X aus dem Sampling $X \cap \mathbb{L}$ zu rekonstruieren, siehe Abschnitt 4.3.2.

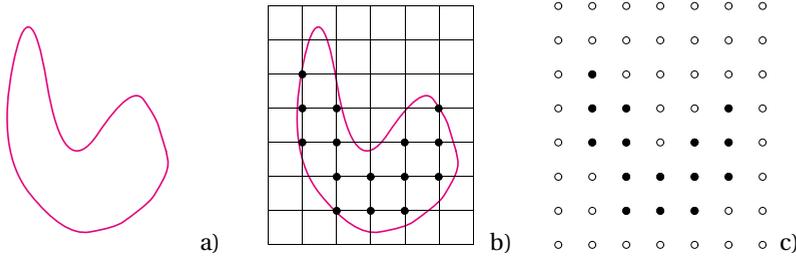


Bild 1.9 a) Eine Menge X , b) ihr Sampling $X \cap \mathbb{L}$ auf einem quadratischen Gitter \mathbb{L} und c) das zugehörige Binärbild, wobei \bullet die Vorder- und \circ die Hintergrundpixel kennzeichnen

Digitalisierungen sind Approximationen von X , die aus einer Abtastung $X \cap \mathbb{L}$ von X auf dem Gitter \mathbb{L} erzeugt werden.

- Die Gauß-Digitalisierung ist die Vereinigung aller Gitterzellen, deren untere linke Ecke in X liegt,

$$\bigcup_{x \in X \cap \mathbb{L}} (C + x), \quad (1.6)$$

Bild 1.10a.

- Die innere Jordan-Digitalisierung ist die Vereinigung aller Gitterzellen, die vollständig in X liegen,

$$\bigcup_{x \in \mathbb{L}} \{C + x : X \subseteq C + x\},$$

Bild 1.10b.

- Die äußere Jordan-Digitalisierung ist die Vereinigung aller Gitterzellen, die von X geschnitten werden,

$$\bigcup_{x \in \mathbb{L}} \{C + x : X \cap (C + x) \neq \emptyset\},$$

Bild 1.10c.

- Betrachtet wird ein Rendering von X , das als Polygonzug die Mittelpunkte aller Kanten der Gitterzellen von \mathbb{L} verbindet, für die jeweils ein Ende in X und das andere außerhalb von X liegt. Das zugehörige Polygon kann als eine Digitalisierung von X aufgefasst werden, Bild 1.10d.

Index

A

Abelsche Integralgleichung 129
Abel-Transformation 129, 189, 275
Ableitungsfilter
– 1. Ordnung 80, 210
– 2. Ordnung 87, 89, 211
Abschließung, morphologische 62, 93
Absorption 134
Absorptionskoeffizient 132
Abstand
– Euklidischer 112, 122
– geodätischer 26, 122, 124
Abtasttheorem 168
Aliasing 171
alternierender Filter 62, 93
Amplitudenspektrum 150
Anordnung 281
Äquivalenz-
– Klasse 102, 103, 106, 107
– Relation 101
arg-Funktion 80, 85, 258
ART 243
aspect ratio 260
assoziativ 58, 60, 72, 91
Autokorrelation 148, 149, 221–223, 226, 267

B

Bandpassfilter 217
Bartlett-Spektrum 151, 224
Beregnungsalgorithmus 121
Bessel-
– Funktion 1. Art 157, 159, 216, 288
– modifizierte 162
– Funktion 2. Art 228
– Transformation 160, 162, 188, 189, 224
bilateraler Filter 55, 97
Bild
– Binär- 42

– Farb- 47
– Grauton- 29
– Hyperspektral- 47
– mehrkanalig 47
– Spektral- 47
– Ultraspektral- 47
Bild-
– Analyse 245
– Drehung 33, 136
– Größe 30
– Randfehler 52, 68, 225, 231, 272
– morphologischer Transformationen 68
– Registrierung 147
– Schärfe 88, 89, 201
– global 148, 151
– lokal 71, 89
– Verzeichnung 33
Bild-Rauschen 54
Binarisierung 42, 52, 92, 95
– des Distanzbildes 112, 113
– Otsu- 43, 269
Binomialfilter 76, 85, 171, 172, 201, 203, 209, 212, 214–217
Binärbild 14, 29, 42
bit reversal permutation 182
Blascke-Formel 264
Bourgersches Gesetz 132
Boxfilter 75
Breite 224, 259, 264
– maximale 260, 262, 281
– minimale 260, 262
– mittlere 94, 259, 260, 265, 272
Breitenfunktion 264, 266
butterfly algorithm 182

C

Canny-Kantendetektion 86
chain code 99

Chamfer-
 – Distanztransformation 115
 – Metrik 115
 Chebychev-Norm 114
 CIELab-Farbraum 50
 City-Block-Norm 114
cloudiness index, CLI 230
 CLT 113
 Colorierung, Pseudo- 47
 Computertomographie 96, 129, 133, 135,
 148, 234, 237, 244
 – diskrete 243
conditional median filter 96
crack code 99
 Crofton-Formel 250–252, 254, 256, 260
 CUDA 5, 80, 128, 175

D

Dawson-Funktion 162
 Detektion
 – Ecken- 71, 87
 – Kanten- 71, 81, 87–90, 94, 95, 216
 dFT 174
 Differenz, Mengen- 14, 52, 53, 60, 65, 112,
 226, 231, 233
 Diffraktion 224
 Diffusionsfilter 97, 163
 – anisotrop 167, 207
 – isotrop 165
 Digitalisierung 22
 – Gauß- 22, 23, 170
 – Jordan- 23
 – innere 22, 23
 – äußere 22, 23
 Dilatation 57, 60, 112–114, 246
 Dirac-Funktion 133, 135, 156, 216
 Dispersion 282
 Distanz
 – Chamfer- 115
 – Euklidische 112
 – geodätische 26, 122
 Distanz-
 – Bild 90
 – Transformation 90, 112, 120, 270, 271
 – Chamfer- 115
 – diskrete 115

– Euklidische 112, 280
 – geodätische 124
 – signierte 113
 distributiv 58, 68, 70, 72, 92
 DoB-Filter 217
 DoG-Filter 217
dots per inch (dpi) 20
 DT 243
 Durchmesser 225
 – des flächengleichen Kreises 248, 281
 – Feretscher 259, 281
 – Kaliber- 259

E

Eckendetektion 71, 87
 – Kanade-Tomasi- 87
 EDT 112–114, 280
 – diskrete 115
 Einflusszone 122, 124
 Einheitszelle, eines Gitters 19, 30
 elliptisches Integral
 – vollständig 265
 Energie 47
 – Dichtespektrum 151
 Entropie 46, 47
 Erosion 61, 68, 112, 113, 246
 Erwartungswert 222, 230
 Euklidisch
 – Abstand 112
 – Distanz 112
 – Distanztransformation 112, 280
 – Norm 112
 Euler-
 – Formel 153
 – Zahl 14, 15, 17, 18, 23, 24, 26–28, 245, 248,
 255, 260, 267
 – pro Flächeneinheit 267
 Euler-Poincaré-
 – Charakteristik 14
 – Formel 17, 27
 Exponentialfunktion 160, 188
 Extinktion
 – integrierte 248
 – mittlere 248, 279
 – Variation 248, 279

F

- Faltung 72, 145, 164, 200
 - diskrete 23, 72, 191
 - kontinuierliche 71
 - von Funktionen 152
- Faltungstheorem der Fourier-Transformation 152, 196, 197, 199, 206, 240
- Farbraum
 - CIELab- 50
 - HSI- 50
 - HSL- 50
 - HSV- 47, 48, 83
 - Lab- 50
 - LCH- 50
 - RGB- 48
 - XYZ- 50
- Farbwinkel 48
- Feature 107, 245
- Feldkamp-Algorithmus 244
- Feretscher Durchmesser 259, 281
- FFT 180
- FIB 147
- Filter
 - alternierender 62, 93
 - Bandpass- 217
 - bilateral 55, 97
 - Binomial- 76, 85, 171, 172, 201, 203, 209, 212, 214–217
 - Box- 75
 - Design 212
 - Diffusions- 97, 163, 165, 207
 - DoB- 217
 - DoG- 217
 - Gabor- 217
 - Gauß- 76, 87, 90, 207, 208, 215
 - Glättungs- 73, 90, 215
 - Gradienten- 80, 210, 212
 - Hochpass- 216
 - Kantendetektions- 81
 - Kasten- 75
 - Laplace- 87, 89, 211, 214, 216, 217, 220
 - linear 23, 71, 148
 - LoG- 88, 217
 - LSE- 72, 200, 201
 - Maximum- 90, 96
 - Median- 96
 - bedingt 96
 - Minimum- 90, 96
 - Mittelwert- 73, 74, 78, 86, 145, 206, 215
 - morphologisch 90, 92, 93, 95
 - nichtlinear 92, 95
 - optimal 148
 - Paganin- 148, 220
 - Prewitt- 86
 - Rangordnungs- 95, 96
 - Roberts- 85
 - Scharr- 212
 - Sobel- 85, 212
 - Tiefpass- 171, 215
 - Wiener- 148, 220
 - Young-van Vliet- 76
- Filter-
 - Kern 72
 - Maske 23, 72, 73, 191, 193, 200
- filtered back-projection* 241
- Filterung 23, 57, 200
 - diskrete 191
 - inverse 80, 148, 201, 218, 226
- floor-Funktion 31, 33, 41, 44, 45, 49, 277, 290
- Flutungsalgorithmus 121, 127, 128
- Fläche 146, 245, 248, 255, 267
 - der konvexen Hülle 259
- Flächenanteil 96, 267, 268, 269
- Formation von Papier 227
- Formfaktor 247, 260, 261, 265, 281
- Fourier-
 - Bessel-Transformation 159, 160, 162
 - Co-Transformation 150
 - Interpolation 32, 174
 - Koeffizient 265
 - Kosinustransformation 160
 - Optik 149
 - Reihe 173, 265
 - Sinustransformation 160
 - Transformation 21, 148, 160, 189, 191
 - diskrete 148, 172, 174
 - inverse 150, 197, 219
 - kontinuierliche 150, 194
 - n -dimensionale 150
 - schnelle 47, 180, 234
- Fourier-Transformation 13
- Freemans Kettencode 99, 250, 255

- Frequenz 150
 - Funktion
 - arg- **80**, 85, 258
 - Bessel- 157, 159, 228, 288
 - Breiten- 264
 - Dawson- 162
 - Dirac- 133, 135, 156, 216
 - Exponential- 160, 188
 - floor- 31, 33, 41, 44, 45, 49, 277, 290
 - Gamma- 161, 228, 278
 - Gauß- **76**, 86, 131, 132, 145, 154–159, 164, 205, 216
 - Indikator- 29, 42, 52, 57, 74, 90, 135, 138, 145, 156, 159, 161, 170, 215, 226, 272
 - sinc- 153, 156, 157
 - Struve- 162
 - Stütz- **263**, 264, 266
 - zufällige 221
- G**
- Gabor-Filter 217
 - Gammafunktion 161, 228, 278
 - Gauß-
 - Digitalisierung 22, 23, 170, 250
 - Filter 87, 90, 145, 208, 215
 - anisotrop 207, 208
 - isotrop **76**, 207
 - Funktion **76**, 86, 131, 132, 145, 154–159, 164, 205, 208, 209, 216
 - Prozess 227
 - Verteilung 54
 - Gaußscher Prozess 227
 - Rauschen 73
 - Fehlerintegral 162
 - Rauschen 54, 166, 220, 229
 - gefilterte Rückprojektion 241
 - geodätisch
 - Abstand 115, **122**, 124
 - Distanztransformation 115, **124**
 - Geradengleichung 130
 - Gesamtenergie 151, 171
 - Gitter
 - Abstand 19, 21
 - Basis 19
 - Einheitszelle 19
 - hexagonal 20, 21, 27, 187
 - homogen **19**, 30
 - inverses 21, 188
 - quadratisch 20
 - rechteckig 20
 - Gleichdick 260
 - Gleichmäßigkeit 47
 - Glättungsfilter 215
 - eckenerhaltend 97
 - kantenerhaltend 96, 97, 165
 - linear 73, 90
 - lokal adaptiv 163
 - morphologisch 90
 - ortssensitiv 163
 - GPU 76
 - Gradient 86
 - morphologisch 90, 93, 94
 - Norm 80, 85, 89, 93, 165
 - Richtung 80, 81, 86
 - Gradientenfilter **80**, 210, 212
 - Graham-Scan-Algorithmus 261
 - Granulometrie, sphärische 54, 64, 65, 113, 270, **271**
 - Grautonbild **29**
 - digital 30
 - kontinuierlich 29
 - Grauwert
 - Histogramm 42, **44**, 46, 96
 - maximal 46
 - minimal 46
 - mittlerer 43, 46, 248
 - Standardabweichung 43, 46
 - Transformation 44
 - Übertragungsmatrix 47
 - Variation 248
 - Grenzfläche, spezifische 269
 - GRF 227
- H**
- Hadwiger, Satz von 247
 - Hadwigers rekursive Formel 17
 - Hankel-Transformation **160**, 189
 - Haralick-Koeffizienten 47
 - Hauptachsentransformation **51**, 258
 - Hesse-Matrix **89**
 - HFA-Kreis 189
 - Hilbert-Transformation 133

Histogrammausgleich 44–46, 52
 Hochpassfilter 216
 Homogenität 47
 Hough-Transformation 129, 140
 HSI-Farbraum 50
 HSL-Farbraum 50
 HSV-Farbraum 47, 48, 52, 83
 Hülle, konvexe 259, 261
 Hyperspektralbild 47

I

Idempotenz morphologischer
 Transformationen 62
 Impulsantwort 201
 Indikatorfunktion 29, 42, 52, 57, 74, 90, 95,
 135, 138, 145, 156, 159, 161, 170, 215, 226,
 272
 Inklusions-Exklusions-Prinzip 15, 246
 inneres Volumen 247
 Interferenz 149
 Interpolation
 – bilinear 31, 32, 137
 – linear 32, 243
 – mit bikubischen Splines 32
 – Whittaker-Shannon- 168
 inverse
 – Filterung 80, 148
 – Fourier-Transformation 150, 174
 – Radon-Transformation 129, 131
 – Skalierung der Fourier-Transformation 151
 Ionendünnung 147
 Isotropie
 – eines Filters 72
 – eines Merkmals 245, 246

J

Jarvis-March-Algorithmus 261
 Johnsson-Nyquist-Rauschen 54
 Jordan-Digitalisierung 23
 – innere 22, 23
 – äußere 22, 23

K

Kaliberdurchmesser 259
 Kanade-Tomasi-Eckendetektion 87
 Kanten-
 – Detektion 71, 81, 87–90, 94, 95, 216
 – Canny- 86
 – Erhaltung 96, 165
 – Schärfe 88, 89
 Karhunen-Loeve-Transformation 51
 Kastenfilter 75
 Kennzahl 107
 kommutativ 58, 72
 Komplementarität 26
 Komplementärmenge 14
 Kongruenzklassen von Pixelkonfigurationen
 24, 254, 256
 Kontaktverteilung, sphärische 54, 62, 64, 270
 Kontrast 47
 – Michelson- 46
 – Otsu- 43
 – RMS- 46
 konvexe Hülle 259, 261
 Konvexität 259, 262
 Koordinationszahl 17
 Korngröße 277
 Korrelation 193, 194, 222, 228
 – diskrete 23, 73
 – von Funktionen 152
 Korrelationskoeffizient 146
 Kosinustransformation 160
 – diskrete 176
 – schnelle 183
 Kovarianz 222
 – Mengen- 163
 Kovarianzmatrix 156
 Kreisfrequenz 150
 Kreuzkorrelation 144, 145, 148, 221, 222, 230,
 267
 Kreuzleistungsspektrum 231
 Krümmung, Integral der mittleren 269
 Kurvenlänge 122, 140

L

Label 99
 – Propagation 108

- Labeling 26, 48, 51, **99**, 100, 102, 107, 245, 257
 - Algorithmus 102, 106
 - implizit 107
 - mit Lauflängenkodierung 108
- Lab-Farbraum 50
- Lambert-Beersches Gesetz 132
- Länge
 - der Projektion 251
 - spezifische 269
- Laplace-
 - Filter **87**, 89, 211, 214, 216, 217, 220
 - Operator **71**, 87, 151, 164, 211
 - Pyramide 217
- Lauflängenkodierung 108
- LCH-Farbraum 50
- Leistungsspektrum 224
- linearer Filter **71**, 148
- Linienintegral 140
- Logarithmierung von Pixelwerten 40
- LoG-Filter **88**, 217
- Lookup-Tabelle 104, 112
- LSE-Filter 72, 200, 201

M

- maximale Breite 262
- Maximumfilter 90, **96**
- Medianfilter **96**
 - bedingt 96
- Mengendifferenz 14, 52, 53, 60, 65, 112, 226, 231, 233
- Merkmal eines Objekts 53, 245, 272
- Michelson-Kontrast 46
- minimale Breite 262
- Minimumfilter 90, **96**
- Minkowski
 - Addition **57**, 66, 90
 - diskrete 66
 - Subtraktion **60**, 90
- Mittelwertfilter **73**, 74, 78, 86, 145, 206, 215
- mittlere Breite 95, **259**, 260, 265, 272
- Moiré-Effekt **168**, 171, 172, 216
- morphologisch
 - Filter **90**, 92, 93, 95
 - Gradient 90, 93, 94

N

- Nabla-Operator 80, 90, 93, 151, 210
- Nachbarschaft 14, 17, 24, **26**, 27, 28, 102, 103, 106, 121, 255, 261
 - Graph 26
 - komplementär 14, 28, 29
- Netzwerkformel **18**, 277
- Nodularität 281
- Norm
 - Chebychev- 114
 - City-Block- 114
 - des Gradienten 80, 83, 85, 89, 165
 - Erhaltung 72, 150
 - Euklidisch 112
 - Maximum 114
 - verallgemeinert 114
- Nyquist-Frequenz, kritische 169

O

- Oberfläche, spezifische 269
- Objekt 53, 99, 100, 245
- Objekt-
 - Größe 248
 - Merkmal 53, 99, 245
 - Richtung 257
 - Trennung 100, 112
- Öffnung, morphologische **62**, 93
- Öffnungs-Transformation 113, 272
- Offset eines Bildes 30
- OFR 152, 191
- optimale Filter 148
- OR 152, 191
- Ortsfrequenzraum 148, 152, 191
- Ortsraum 152, 191
- Otsu-
 - Binarisierung **43**, 269
 - Kontrast 43

P

- Paganin-Filter 148, 220
- Parallelisierbarkeit
 - linearer Filter 72
 - morphologische Transformation 59, 72
 - morphologischer Filter 92
- Parseval-Gleichung 150
- Pfad 100, 122
 - diskreter 102
 - kürzester 122
 - Länge 123
 - Verbundenheit 100–102
- Phasen-
 - Gang 151
 - Rekonstruktion 148
 - Spektrum 151
- Pixel 30
- Pixel-
 - Größe 19, 30
 - Konfiguration 23, 24, 248, 254–256
 - Matrix 30
- Plancherel-Gleichung 150, 175
- Poisson-
 - Verteilung 55
 - Voronoi-Mosaik 278
- Poissonsches Rauschen 55, 73
- Polarkoordinaten 33, 39
- pores per inch*, PPI 225
- Porosität 280
- Powerspektrum 54, 151, 223, 224, 267
- Prewitt-Filter 86
- Projektions-Schnitt-Theorem 235
- Pseudocolorierung 47, 104, 107, 255
- PSF 201
- Ptychographie 148
- Punktantwort 201

Q

- Quantil 96
- Quermaß 267
 - -dichte 267
- Quickhull-Algorithmus 261

R

- Radon-Transformation 33, 129, 130, 131, 189, 234, 235, 251
 - inverse 33, 129, 131, 133, 234, 240, 241
 - schnelle 148
- Randfehler 137, 139
- Randlänge 250
- Rangordnungsfilter 95, 96
- Rasterelektronenmikroskopie 147
- Rauschen 54, 70, 221
 - 1/f- 54
 - Gaußsches 54, 73, 166, 220, 229
 - Johnson-Nyquist- 54
 - Poissonsches 55, 73
 - rosa 54
 - Salz- und Pfeffer- 55, 90, 96, 97
 - Schrot- 55
 - thermisch 54
 - weißes 54
- Rauschleistungsdichte 54
- Referenzbild 70, 78, 90
- Regionenwachstum 125
- Registrierung 33, 147, 148
- Rekonstruktion
 - algebraisch 243
 - Phasen- 148
- REM 147
- Rendering 22, 23, 170, 250, 251, 254, 255
 - Oberflächen- 23
 - Volumen- 148
- Reuleaux-Dreieck 260
- RGB-Farbraum 47, 48, 52
- Richtung 71, 80
 - des Gradienten 81, 86, 212
 - eines Objekts 257
- Richtungs-
 - Ableitung 80, 210
 - Verteilung 113, 139
- RMS-Kontrast 46, 51
- Roberts-Filter 85
- Röntgenschwächung 134
- Rotationsmatrix 33, 137
- Rückprojektion 238, 239
 - gefilterte 133, 148, 240, 241, 242
- Rundheit 247, 260, 281
 - der konvexen Hülle 259

S

Salz- und Pfefferrauschen 55, 90, 96, 97
 Sampling 22, 23, 30, 168, 172, 250
 Schar-Filter 212
 Schrotrauschen 55
 Schärfe, Bild- 88, 89, 148, 151
 Schwerpunkt eines Objekts 257
 Schwächungskontrast 134
 SDM 113
 SDT 113
 Segment 120
 Segmentierung 52, 61, 90, 100, 113, 129, 149, 245
 – Über- 90, 128
 Sehnenlängentransformation 113
 Separabilität 89
 – der Distanztransformation 118
 – der Filtermaske 72, 75, 200, 206
 – der Fourier-Transformation 152, 155, 184, 199
 – des strukturierenden Elements 60
 – in Koordinatenrichtung 75, 206
 – morphologischer Transformationen 58, 59, 67
 – von Binomialfiltern 77
 – von Gauß-Filtern 76
 Serienschmitttechnik 147
 Shading 41, 78
 – additiv 41, 216
 – Korrektur 41, 70, 90, 94, 186, 216
 – multiplikativ 41
 Shells Algorithmus 98, 261
shot noise 55
 Signal-Rausch-Verhältnis 55
 sinc-Funktion 153, 156, 157
 Sinogramm 132–138, 237–239, 244
 Sinusoid 101
 Sinustransformation 160, 162
 – diskrete 176
 – schnelle 183
 Skelettierung 26
 Sobel-Filter 85, 212
 Sortierung von Pixeln
 – distributiv 98, 127
 – Shells Algorithmus 98, 128, 261

Spektral-
 – Bild 47
 – Dichte 53, 54, 148, 149, 151, 218, 222, 223–226, 230
 Spektrum
 – Amplituden- 150
 – Bartlett- 151, 224
 – Energiedichte- 151
 – Kreuzleistungs- 231
 – Leistungs- 224
 – Phasen- 151
 – Power- 54, 151, 223, 224, 267
 spezifische Randlänge 267
spike noise 55
 Steiner, Satz von 246, 247, 262
 Streckungsgrad 260, 279
 Struktur, zufällige 148, 149, 222, 266, 272
 strukturierendes Element 53, 57, 60, 65, 90, 94, 95, 112, 115
 Strukturmodellindex 269
 Strukturtensor 86, 90
 Struve-Funktion 162
 Stützfunktion 263, 264, 266
 Subpixelgenauigkeit 257
support function 263, 266
 Symmetrie eines Filters 72

T

Template-Matching 129, 145, 146–148
 Tiefe von Defokus 89
 Tiefpassfilter 171, 215
 Tomographie 237
 Top-Hat-Transformation 65, 93, 94
 Transferfunktion
 – der Richtungsableitung 210
 – des Gradientenfilters 210
 – des Laplace-Filters 211
 – linearer Filter 201, 202
 – von Binomialfiltern 203, 204
 – von Gauß-Filtern 168, 207
 – von Mittelwertfiltern 206
 Fourier-Bessel 159
 – Abel- 129, 189, 275
 – Bessel- 188, 189, 224
 – Distanz- 90, 112, 120, 270, 271
 – Chamfer- 115

- diskrete 115
- geodätische 115, 124
- Fourier- 13, 21, 148, 150, 189, 191
 - diskrete 172, 174
 - inverse 150, 174, 197, 219
 - kontinuierliche 150
 - schnelle 47, 180, 234
- Hankel- 160, 189
- Hauptachsen- 51, 258
- Hilbert- 133
- Hough- 129, 140
- Karhunen-Loeve- 51
- morphologische 53, 57, 66, 90, 92, 93
- Öffnungs- 113, 272
- Radon- 33, 129, 130, 131, 189, 234, 235, 251
 - inverse 33, 129, 131, 133, 234, 240, 241
 - schnelle 148
- Sehnenlängen- 113
- Top-Hat- 65, 93, 94
- Wasserscheiden- 26, 48, 51, 90, 100, 112, 120, 121
- Zylinderhut- 65, 93, 94
- translations-äquivariant 71
- trigonometrische Rekurrenz 176, 188

U

- Übersegmentierung 90, 128
- Ultraspektralbild 47
- Umfang 245, 250, 251, 253–256, 265, 267
 - der konvexen Hülle 259, 260

V

- Varianz 222, 230
- Verbundenheit 100
 - Pfad- 100, 101

- Verzeichnung 33
 - kissenförmig 39
 - tonnenförmig 39
- Verzeichnungskorrektur 34
- Volumen-
 - anteil 268, 277
 - rendering 148
- Volumen, inneres 247
- Voronoi-Zerlegung 127, 278

W

- Wasserscheidentransformation 26, 48, 51, 90, 100, 112, 120
 - Inter-Pixel- 121
 - topologisch 121
- Whittaker-Shannon-Interpolation 168
- Wicksellsches Korpuskelproblem 129
- Wiener-
 - Filter 148, 220
 - Khintchine-Theorem 226
- Wolkigkeit von Papier 227
- WST 120

X

- XYZ-Farbraum 50

Y

- Young-van Vliet-Filter 76

Z

- Zeiligkeit 278
- Zusammenhangskomponente 14, 17, 99, 100, 107, 245
- zusammenhängend, topologisch 100
- Zylinderhuttransformation 65, 93, 94