FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Evaluierung und Visualisierung von Interest-Point-Detektoren

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Medizinische Informatik

eingereicht von

Patrizia Eisele
Matrikelnummer 0226529

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuer: Ao.Univ.Prof. Dr. Horst Eidenberger

Wien, 22.11.2011

(Unterschrift Verfasserin) (Unterschrift Betreuer)

Erklärung zur Verfassung der Arbeit

Patrizia Eisele
Krongasse 4/6, 1050 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasserin)

Abstract

In this work an overview of the most important methods for interest-point-detection is provided. Common concepts are explained in detail. Furthermore, a framework is implemented which integrates common interest-point-detectors and facilitates a visual comparison of the results of the different techniques. A set of images composed of simple structures is allocated for the comparison.

Many applications in the field of computer vision which rely on computer-aided processing of image data, for example tracking, object recognition and 3D-reconstruction depend on the detection of interesting structures in images prior to further processing steps to allow for robust image matching. These structures usually correspond to so-called blobs or corners and are referred to as local features.

The position of local features in images is determined by so-called interest-points or interest-regions. Techniques to identify local features in images are called interest-point-detectors. Methods based on interest-point-detection have proven to be especially well-suited for robust image matching and are therefore most commonly used in current computer vision systems. In contrast to other approaches, methods based on interest-points use local image information for the detection of the aforementioned features. Thus, they produce good results even when objects in images are partially occluded. Furthermore, local features determined by interest-point-detectors are robust to various geometric and photometric image transformations and yield a compact representation of image content. Research on methods for the detection of robust interest-points has been done since the late seventies. Accordingly there is a great number of different interest-point-detectors nowadays. In order to be able to choose the appropriate interest-point-detector for a specific task it is vital to familiarize oneself with the basic concepts and methods of interest-point-detection.

Kurzfassung

In dieser Arbeit wird ein Überblick über die wichtigsten Methoden zur Identifizierung von Interest-Points gegeben. Häufig eingesetzte Konzepte werden im Detail erklärt. Des Weiteren wird ein Framework entwickelt, welches gängige Interest-Point-Detektoren integriert und einen visuellen Vergleich der Ergebnisse dieser Detektoren ermöglicht. Eine dafür geeignete Auswahl an Bildern mit sehr einfachen Strukturen wird für den Vergleich zur Verfügung gestellt.

Für eine Vielzahl von Anwendungen im Bereich der Computer Vision die auf der computergestützten Verarbeitung von Bilddaten basieren, wie beispielsweise Objektverfolgung, Objekterkennung oder 3D-Rekonstruktion, ist es nötig, in einem ersten Schritt interessante Strukturen in Bildern zu identifizieren. Mit Hilfe der interessanten Strukturen soll die Durchführung eines robusten Bildabgleichs ermöglicht werden. Diese interessanten Strukturen werden als lokale Features bezeichnet und entsprechen für gewöhnlich Eckpunkten oder blob-ähnlichen Strukturen in Bildern.

Die Position von lokalen Features in Bildern wird durch sogenannte Interest-Points oder auch Interest-Regions beschrieben. Verfahren zur Identifikation lokaler Features werden als Interest-Point-Detektoren bezeichnet. Methoden, die auf der Verwendung von Interest-Points basieren, haben sich als besonders geeignet zur Durchführung eines robusten Bildabgleichs erwiesen. Aus diesem Grund werden solche Methoden am häufigsten in modernen Computer Vision-Systemen eingesetzt. Verfahren, die auf Interest-Points basieren, verwenden lokale Bildinformation zur Detektion relevanter Features. Im Gegensatz zu vielen anderen Ansätzen sind sie in der Lage, gute Resultate beim Bildabgleich zu erzielen, auch wenn die Objekte in Bildern teilweise verdeckt sind. Zudem sind Features, die auf Interest-Points lokalisiert sind, robust gegenüber unterschiedlichen geometrischen und photometrischen Bildtransformationen und realisieren eine kompakte Repräsentation des Bildinhalts.

Erste Interest-Point-Detektoren wurden bereits Ende der Siebziger entwickelt. Auch heute noch ist die Identifikation interessanter Punkte ein belebtes Forschungsgebiet. Heute existiert eine große Anzahl verschiedener Verfahren zur Detektion von Interest-Points. Um den passenden Detektor für eine bestimmte Aufgabenstellung zu finden, ist es daher notwendig, sich mit den grundlegenden Konzepten und Methoden zur Interest-Point-Detektion zu befassen.

Inhaltsverzeichnis

1 Einführung

In diesem Kapitel wird zunächst die Relevanz von Interest-Point-Detektoren für die computergestützten Verarbeitung von Bildern erläutert (Abschnitt 1.1). Danach werden die in dieser Arbeit verfolgten Ziele definiert und die Vorgehensweise zum Erreichen der Ziele geschildert (Abschnitt 1.2). Es werden einige typische Anwendungsbeispiele von Interest-Point-Detektoren vorgestellt (Abschnitt 1.3). Abschließend wird ein Überblick über Struktur und Inhalt der nachfolgenden Kapitel gegeben.

1.1 Motivation

Die Menge des digitalen Bildmaterials nimmt in vielen Bereichen unseres Lebens stetig zu. Bedingt durch sinkende Preise für Digitalkameras und Speicherplatz, ist es heute beispielsweise auch Privatpersonen möglich, jederzeit Fotos zu machen oder Videos aufzunehmen. Im medizinischen Bereich werden durch verschiedene bildgebende Verfahren wie Computertomographie und Magnetresonanztomographie große Mengen an Bilddaten erzeugt. Auch im Internet finden sich enorme Mengen an Bildmaterial. Aufgrund dieser regelrechten Flut an digitalen Bildern wird es immer wichtiger, computergestützte Systeme für deren Verarbeitung zu schaffen. So will man beispielsweise imstande sein, große Sammlungen digitaler Bilder effizient durchsuchen zu können. Computergestützte Systeme sollen in der Lage sein, automatisiert Bildinhalte zu analysieren und so die automatisierte Verarbeitung der Bilder auf Basis ihrer Inhalte ermöglichen [29].

Für die Realisierung solcher Systeme werden Verfahren benötigt, die einen robusten Bildabgleich (engl.: Image Matching) ermöglichen. Beim Bildabgleich werden Bilder miteinander verglichen, um jeweils korrespondierende Bildbereiche zu identifizieren. Übereinstimmende Bildbereiche geben dann beispielsweise Aufschluss darüber, ob die verglichenen Bilder ähnliche Objekte oder Szenen beinhalten. In einer Vielzahl von Anwendungen, wie beispielsweise im Bereich Bewegungserkennung, Objekt- und Szenenerkennung oder 3D-Rekonstruktion, spielt der Bildabgleich eine zentrale Rolle [39]. Zur Durchführung eines robusten Bildabgleichs haben sich in den letzten Jahren besonders Verfahren als geeignet erwiesen, die auf sogenannten *lokalen Features* (Bildmerkmalen) basieren. Lokale Features sind lokale Strukturen im Bild, die sich von ihrer direkten Umgebung unterscheiden [75], einen hohen Informationsgehalt aufweisen und die aufgrund ihrer lokalen Struktur für die nachfolgenden Bearbeitungsschritte auf irgendeine Art und Weise "interessant" sind [73]. Welche Eigenschaften diese Stellen aufweisen müssen, um als interessant zu gelten, hängt vor allem von der Aufgabe ab, die man gedenkt zu lösen. Die Position lokaler Features im Bild wird durch sogenannte Interest-Points beschrieben und mit Hilfe von Interest-Point-Detektoren ermittelt [71].

Der Einsatz von Interest-Point-Detektoren hat sich aus mehreren Gründen als besonders günstig erwiesen [24]. So sind Verfahren, die auf Interest-Point-Detektoren aufbauen, beispielsweise sehr robust gegenüber verschiedenen Bildtransformationen und können daher beim Bildabgleich selbst dann noch gute Ergebnisse erzielen, wenn Objekte in einem Bild teilweise verdeckt sind. Weiters können sie auch in Anwendungen eingesetzt werden, bei denen a priori keinerlei Wissen über das verarbeitete Bildmaterial verfügbar ist. Mit Hilfe von Interest-Point-Detektoren können schon sehr früh im Verarbeitungsprozess relevante Bildstellen, d.h. Bildbereiche mit hohem Informationsgehalt, identifiziert werden. Dadurch ist eine kompakte Repräsentation des Bildinhalts möglich und nachfolgende Bearbeitungsschritte müssen dann nur mehr auf den ausgewählten Interest-Points ausgeführt werden. Durch den Einsatz von Interest-Points kann der Berechnungsaufwand somit beträchtlich minimiert werden.

Die Vorgehensweise beim Bildabgleich basierend auf lokalen Features kann grob in folgende Phasen unterteilt werden [71]:

Feature-Detektion: Es werden lokale Features[1] identifiziert. Das sind bestimmte, charakteristische Strukturen im Bild. Die Position des ermittelten lokalen Features wird durch einen Interest-Point festgelegt. Manche Detektoren ermitteln neben der Position des Features auch noch weitere Kenngrößen, wie beispielsweise Informationen über die Ausdehnung des detektierten lokalen Features. In solchen Fällen bezeichnet man das Ergebnis der Feature-Detektion als Interest-Region [73]. Verfahren zum Auffinden lokaler Features werden als Feature-Detektoren oder auch als Interest-Point-Detektoren bezeichnet.

Feature-Deskription: Um die Informationen die ein lokales Feature trägt effizient weiterverarbeiten zu können, wird eine kompakte Beschreibung des Features berechnet. Die Beschreibung wird, wenn ein Interest-Point vorliegt, aus einer bestimmten Umgebung um den Interest-Point herum berechnet [75]. Wenn eine Interest-Region ermittelt wurde, wird die Beschreibung direkt aus dieser Region berechnet. Es wird für jedes identifizierte lokale Feature eine Beschreibung ermittelt. Verfahren zur Berechnung dieser Beschreibung werden als Deskriptoren bezeichnet.

Bildabgleich: Abschließend werden die aus verschiedenen Bildern berechneten Beschreibungen miteinander verglichen, um Übereinstimmungen zu finden. Ähnliche Beschreibungen sollen Hinweis über die vorhandenen Objekte oder Objektkategorien geben.

1.2 Zielsetzung und Vorgehensweise

Im Rahmen der vorliegenden Arbeit soll ein theoretischer Vergleich gängiger Verfahren zur Detektion von Interest-Points durchgeführt und ein praktischer Vergleich ermöglicht

[1] Grundsätzlich unterscheidet man zwischen globalen und lokalen Features. Globale Features dienen dazu, den gesamten Bildinhalt zu beschreiben und nehmen keine Rücksicht auf lokale Gegebenheiten. Sie sind oft schneller und einfacher zu verarbeiten als lokale Features, sind aber nicht sehr robust gegenüber Überdeckungen [71, 75].

werden. Es existiert eine Vielzahl unterschiedlicher Interest-Point-Detektoren die verschiedene Arten von Bildstrukturen identifizieren und einen unterschiedlichen Grad an Robustheit gegenüber Bildtransformationen aufweisen. Zur Realisierung des Vergleichs ist es daher zunächst nötig, sich einen guten Überblick über gebräuchliche Methoden zur Detektion und die prinzipiellen Möglichkeiten zur Berechnung von Interest-Points zu verschaffen.

Dazu wird in einem ersten Schritt eine Zusammenfassung vorhandener Literatur im Fachgebiet erarbeitet. Es wird geklärt, welche Kategorien von Detektoren es grundsätzlich gibt, welche Trends in ihrer Entwicklung sich über die Zeit abzeichnen und welche Methoden als relevant gelten. Darauf aufbauend werden die theoretischen Konzepte ausgewählter, in der Praxis häufig eingesetzter, Interest-Point-Detektoren im Detail analysiert. Die betrachteten Detektoren sind der Harris-, Harris Laplace-, Determinant Of Hessian-, Hesse Laplace-, Laplacian Of Gaussian-, Difference Of Gaussian-, FAST- und MSER-Detektor. Die Grundlagen auf denen die Verfahren aufbauen (wie beispielsweise lokale Operatoren und Skalenräume) werden erläutert. Zudem werden die einzelnen Berechnungsschritte beschrieben, die zur Ergebnisbildung beim jeweiligen Detektor führen.

Anschließend wird eine Applikation in MATLAB implementiert, die den praktischen Vergleich der ausgewählten Interest-Point-Detektoren ermöglicht. Viele Autoren stellen Implementierungen der von ihnen entwickelten Detektoren zur freien Verfügung. Die Herausforderung bei der Entwicklung der Applikation liegt daher vor allem darin, die heterogenen Implementierungen auf geeignete Art und Weise zu integrieren. Beispielsweise sollen alle Detektoren über die gleichen Parameter steuerbar und die berechneten Ergebnispunkte gut visuell vergleichbar sein. Weiters sollen Interest-Points, die von mehreren unterschiedlichen Detektoren identifiziert werden, auf einen Blick sichtbar sein.

Mit Hilfe der im Zuge der Arbeit entwickelten Applikation kann anschließend ein Vergleich der Detektoren hinsichtlich der gefundenen Interest-Points durchgeführt werden. Dazu werden die verschiedenen Detektoren auf eine Auswahl von Bildern mit sehr einfachen Strukturen angewendet und die Ergebnisse nebeneinander dargestellt. Dadurch können visuell Unterschiede und Gemeinsamkeiten der Interest-Point-Detektoren festgestellt werden. So kann beispielsweise herausgefunden werden, welche Bildstrukturen von den jeweiligen Detektoren hauptsächlich identifiziert werden und wie stark die Menge der detektierten Interest-Points variiert. Auf Basis der zuvor erarbeiteten theoretischen Kenntnisse werden Unterschiede und Gemeinsamkeiten in den Ergebnissen der Detektoren erklärbar. Stärken und Schwächen der jeweiligen Detektoren können herausgestellt werden. Die im Rahmen der Arbeit entwickelte Applikation soll zu Demonstrationszwecken in Lehrveranstaltungen eingesetzt werden können.

1.3 Anwendungsgebiete von Interest-Points

Interest-Points haben sich als robuste Ausgangsbasis für vielfältige Anwendungen erwiesen, vor allem im Bereich der Computer Vision [39]. Einige typische Anwendungsgebiete sind das Stitching, die inhaltsbasierte Bildsuche und der Stereo-Bildabgleich. In den nachfolgenden Abschnitten, die sich an [72] orientieren, wird darauf eingegangen, wie diese

Aufgaben mit Hilfe von Interest-Points gelöst werden können.

1.3.1 Stitching

Der Vorgang des Zusammenfügens mehrerer Einzelbilder zu einem Gesamtbild wird als Stitching[2] bezeichnet. Stitching wird beispielsweise zur Fertigung von Satellitenbildern eingesetzt, findet aber auch Einsatz im Bereich der digitalen Fotografie, zum Erstellen von Panoramabildern. Die meisten handelsüblichen Digitalkameras haben bereits eine Stitching-Funktion integriert. Um damit ein Panoramabild anzufertigen, nimmt der Fotograf nacheinander mehrere Einzelbilder derselben Szene auf. Dabei wird der betrachtete Bildausschnitt zwischen jeder Aufnahme leicht verschoben. Die Bildausschnitte werden so gewählt, dass jeweils ein kleiner Bereich der betrachteten Szene in zwei der aufeinander folgenden Einzelbildern enthalten ist. Mit Hilfe des Stitchings werden die Einzelbilder dann so zusammengesetzt, dass es aussieht, als wäre nur ein einziges Bild mit Weitwinkel aufgenommen worden [72].

Heutige Techniken zum Stitching von Bildern basieren zumeist auf der Verwendung von Features [72]. Abbildung 1.1 zeigt die Vorgehensweise beim Erstellen eines Panoramas anhand von zwei Beispielbildern. Die Bilder stammen aus einer Arbeit von Brown und Lowe [8], in der ein automatisiertes Verfahren zur Erzeugung von Panoramas aus einer großen Menge von unsortierten Bildern vorgestellt wird.

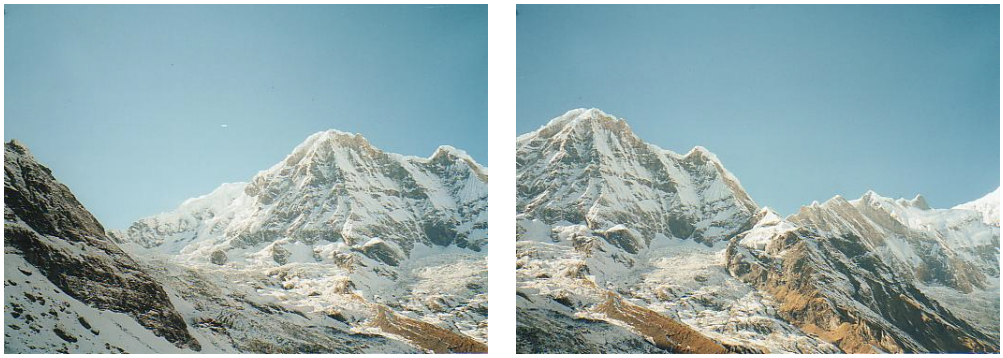
In einem ersten Schritt werden Interest-Points in beiden Einzelbildern detektiert. Durch einen Bildabgleich werden korrespondierende Stellen identifiziert. Die überlappenden Bildbereiche der beiden Bilder werden anhand der Interest-Points zueinander ausgerichtet und übereinandergelegt. Schließlich werden die Farbinformationen aus den überlappenden Bildbereichen so vermischt, dass ein möglichst nahtloser Übergang entsteht. Abschließend werden Deformationen, die durch unterschiedliche Ausrichtung der Kamera beim Erzeugen der Einzelaufnahmen entstanden sind, soweit wie möglich rückgerechnet [72].

1.3.2 Inhaltsbasierte Bildsuche

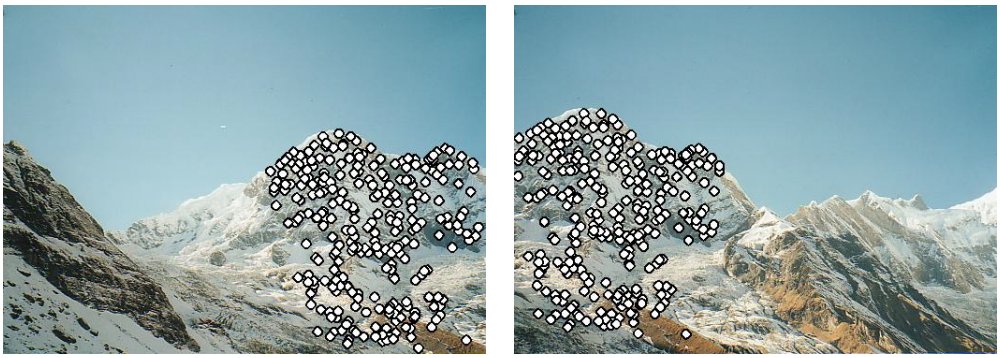
Wie eingangs erwähnt wird es immer wichtiger, Strategien zu entwickeln, die es uns ermöglichen, relevante visuelle Informationen aus großen Ansammlungen digitaler Bilder zu extrahieren. Dies ist keine einfache Aufgabe. Schon die Suche nach einem bestimmten Urlaubsbild in unserer privaten Bildsammlung kann zur Qual werden, wenn eine große Menge von Fotos händisch durchsucht werden muss [73].

Für die meisten gängigen Anwendungen werden Bilddatenbanken auf Basis textueller Information durchsucht. Dabei werden zum Beispiel Bildtitel oder andere geschriebene Informationen in der Umgebung des Bildes durchsucht. In manchen Systemen werden Schlüsselwörter zur Beschreibung des Bildinhalts generiert, die dann durchsucht werden können. Die manuelle Beschreibung jedes einzelnen Bildes mit Schlüsselwörtern ist allerdings sehr zeitintensiv und reicht zumeist nicht aus, den gesamten Bildinhalt zu erfassen [70].

[2] to stitch bedeutet soviel wie nähen, heften



(a)

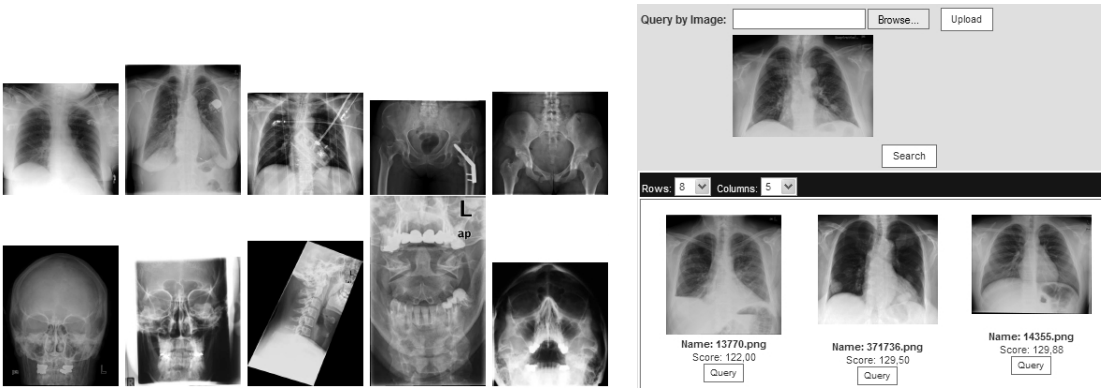


(b)

(c)

(d)

Abbildung 1.1: Zusammensetzung eines Bildpanoramas: (a) Zwei Einzelbilder derselben Szene (b) Mit dem SIFT-Detektor identifizierte Interest-Points (c) Ausrichtung überlappender Bildbereiche anhand der Interest-Points (d) Aus vielen Einzelbildern zusammengesetztes Panorama. Die Übergänge zwischen den Einzelbildern sind kaum mehr sichtbar. Bilder aus [8]



(a) (b)

Abbildung 1.2: System zur inhaltsbasierten Bildsuche für medizinische Bilddatenbanken aus [13]: (a) Beispielbilder aus der verwendeten Bilddatenbank (b) Vom Anwender gewähltes Beispielbild und zurückgelieferte Resultate

Anstelle dieser textbasierten Suche wäre es wünschenswert, Systeme zu entwickeln, die eine Suche nach relevanten Informationen in Bildern auf Basis des Bildinhalts ermöglichen. Während diese Aufgabe für den Menschen ein Leichtes ist, ist sie für den Computer nahezu unlösbar. Der Mensch kann mühelos Objekte in seiner Umgebung und auf Bildern erkennen und entscheiden, welche Inhalte für ihn relevant sind. Wenn wir beispielsweise einmal gelernt haben, was ein Tisch ist, dann sind wir in der Lage, jeden beliebigen Tisch in Bildern zu identifizieren. Es ist dabei egal aus welcher Richtung der Tisch zu sehen ist oder welche Art von Tisch es ist. Für einen Computer ist dies nicht möglich; ihm fehlt das Wissen über die reale Welt und ein Bild stellt für ihn nur eine Anordnung von Farb- oder Helligkeitswerten, angeordnet in einer zweidimensionalen Matrix dar. Der Computer kann den Inhalt des Bildes nicht verstehen [26].

Durch die Extraktion von Features kann eine Repräsentation der charakteristischen Eigenschaften eines Bildes generiert werden, die eine inhaltsbasierte Verarbeitung von Bilddaten durch den Computer ermöglicht. Inhaltsbasierte Bildsuche (engl.: Content-Based Image Retrieval) ist das automatische Holen relevanter Informationen aus Bilddatenbanken basierend auf dem Bildinhalt, der durch Features beschrieben wird [41]. Dazu werden für jedes Bild der Datenbank Features und Feature-Deskriptoren berechnet. Diese werden zu einem sogenannten Feature-Vektor zusammengefasst und abgespeichert. Für die Bildsuche stellt der Anwender ein Beispielbild zur Verfügung, für das dann ebenfalls ein Feature-Vektor erzeugt wird. Abschließend wird der Feature-Vektor des Beispielbildes mit den gespeicherten Vektoren der Bilddatenbank verglichen. Als Suchergebnis werden diejenigen Bilder zurückgeliefert, deren Feature-Vektoren denen des Beispielbildes am ähnlichsten sind [70]. Abbildung 1.2 zeigt ein Beispiel eines Systems zur inhaltsbasierten Bildsuche aus [13], das medizinische Bilddaten verarbeitet.

Die Wahl der Features, die berechnet werden sollen, ist dabei abhängig von der gewünschten Anwendung und von der Art der Bilder in der Bilddatenbank. Features müssen so gewählt werden, dass sie für ähnliche Bilder möglichst ähnliche Werte aufweisen und

(a) (b) (c)

Abbildung 1.3: (a), (b) Stereo-Aufnahmen (c) daraus erzeugtes Disparitätsbild; Bilder aus [72]

für Bilder mit unterschiedlichem Inhalt verschiedene Werte annehmen.

Lokale Features, deren Positionen durch Interest-Points bestimmt sind, haben sich aufgrund ihrer Eigenschaften (siehe Abschnitt 2.1) als besonders geeignet für den Einsatz bei der inhaltsbasierten Bildsuche erwiesen (vgl. [24]). Zum einen bieten sie die Möglichkeit sich sehr früh auf relevante Informationen im Bild zu konzentrieren und bilden so eine sehr kompakte Repräsentation von Bildern. Dies ist wichtig, weil bei der inhaltsbasierten Bildsuche oft eine riesige Menge von Bildern miteinander verglichen werden muss. Werden hier Interest-Points eingesetzt, so muss pro Bild nur eine kleine Menge von Feature-Deskriptoren miteinander verglichen werden. Zusätzlich ist es bei dieser Art von Anwendung besonders wichtig, dass die extrahierten Features invariant gegenüber Bildtransformationen sind. Nur so kann sichergestellt werden, dass ähnliche Objekte erkannt werden können, auch wenn sie sich in ihrer Ausrichtung oder Größe unterscheiden. Verfahren, die auf Interest-Points basieren, erreichen heute bereits Invarianz gegenüber affinen Transformationen (siehe Abschnitt 2.1.1) [75]. Weil Interest-Point-Detektoren lokale Informationen in Bildern verwenden, funktionieren sie auch gut, wenn Objekte teilweise verdeckt sind. Sie lassen sich für gewöhnlich so steuern, dass eine ausreichende Menge an Interest-Points detektiert wird, um das Wegfallen einiger Punkte auszugleichen. Interest-Points beziehungsweise die daraus generierten Feature-Deskriptoren können auch zu sogenannten Clustern zusammengefasst werden [71]. Ein solches Cluster beschreibt dann die Eigenschaften einer bestimmten Objektkategorie[3].

1.3.3 Bildabgleich von Stereobildern

Wir nehmen unsere Umgebung dreidimensional wahr. Bedingt durch die Anordnung unserer Augen treffen zwei sich nur geringfügig unterscheidende Abbilder der realen Welt auf die Netzhäute des linken und rechten Auges auf. Das Gehirn setzt die beiden Abbilder zu einem Bild zusammen und so entsteht der Tiefeneindruck [72].

Bei Stereo-Vision-Systemen nehmen zwei Kameras gleichzeitig Bilder derselben Szene auf. Die Kameras werden durch einen gewissen Abstand getrennt, um so den Sehprozess

[3]Oft will man kein bestimmtes Objekt erkennen, zum Beispiel den Stephansdom, sondern eine bestimmte Kategorie von Objekten, zum Beispiel eine Kirche.

beim Menschen nachzubilden. Um eine Zusammenführung der beiden aufgenommenen Einzelbilder zu ermöglichen, werden mittels eines Bildabgleichs korrespondierende Positionen in beiden Bildern identifiziert. Wiederum eignen sich Interest-Points sehr gut als Basis für den Bildabgleich. Im einfachsten Fall unterscheiden sich die zwei Aufnahmen nur durch den horizontalen Abstand der Kameras bei der Aufnahme. Es reicht somit aus, ganz wenige korrespondierende Interest-Points zu finden, um auf die geometrischen Beziehungen zwischen den beiden Aufnahmen zu schließen. Wenn die Beziehung zwischen den beiden Bildern ermittelt wurde, kann ein sogenanntes Disparitätsbild berechnet werden. Die Disparität bezeichnet die horizontale Bewegung, die zwischen linker und rechter Aufnahme stattgefunden hat. Anders gesagt ist sie der Unterschied in der Position der Objekte in den beiden Bildern. Abbildung 1.3 zeigt zwei mit leicht unterschiedlichem Betrachtungswinkel aufgenommene Bilder und das daraus generierte Disparitätsbild [72].

Ist die Disparität bekannt, so kann die Tiefe der Objekte (d.h. der Abstand der Objekte in der Szene von der Kamera) im Bild geschätzt werden. Der Bildabgleich von Stereobildern ist daher ein wichtiges Werkzeug beim Erstellen von 3D-Modellen aus 2D-Bildern, wird aber beispielsweise auch bei der Navigation mobiler Roboter eingesetzt.

1.4 Struktur der Arbeit

In den folgenden Kapiteln wird zunächst der Begriff des Interest-Points und die Eigenschaften, die einen solchen ausmachen, genauer untersucht. Lineare Operatoren und Skalenräume werden im Detail beschrieben. Sie bilden die Grundlage zur Ermittlung von lokalen Features (siehe Kapitel 2). In Kapitel 3 wird ein Überblick der Literatur zu verschiedenen Verfahren zur Detektion von Interest-Points gegeben. Es wird aufgezeigt, anhand welcher Merkmale typischerweise zwischen verschiedenen Gruppen von Detektoren unterschieden wird. Weiters werden in diesem Kapitel mögliche Kriterien für die Durchführung einer Evaluierung von Feature-Detektoren sowie Literatur zu bereits durchgeführten Vergleichen vorgestellt. In Kapitel 4 werden die gängigen Methoden zur Interest-Point-Detektion im Detail betrachtet und die Vorgehensweise bei der Identifizierung der lokalen Features wird beschrieben. Es wird auf bestehende Gemeinsamkeiten verschiedener Verfahren hingewiesen und es werden Vor- und Nachteile der Methoden dargelegt. Die Vorgehensweise bei der Umsetzung von VisIP, der im Rahmen dieser Arbeit entwickelten Applikation zum Vergleich gängiger Interest-Point-Detektoren, wird in Kapitel 5 beschrieben. Funktionalität und Bedienung werden erläutert. Im letzten Kapitel folgt eine Zusammenfassung der Inhalte der Arbeit und der gewonnenen Erkenntnisse, sowie ein kurzer Ausblick (Kapitel 6).

2 Grundlagen

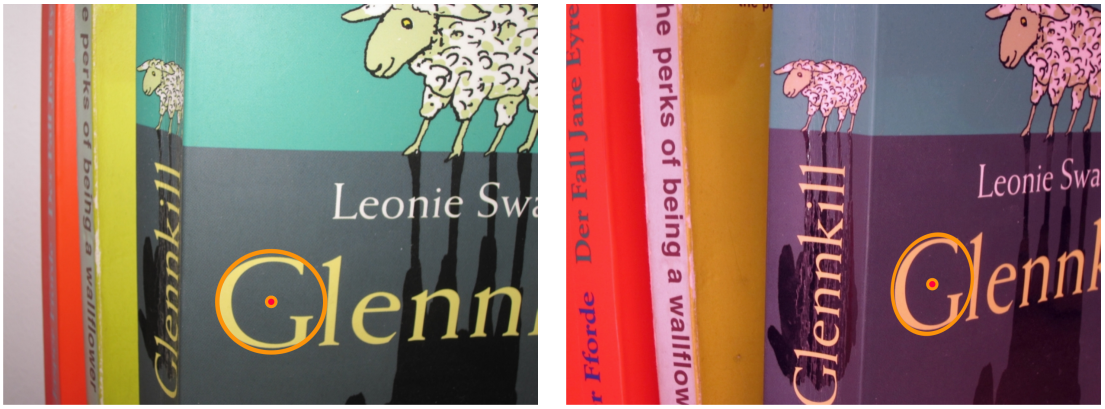
In diesem Kapitel werden die Grundlagen dargelegt, die zum Verständnis der Funktionsweise von Interest-Point-Detektoren notwendig sind. Zunächst wird in Abschnitt 2.1 auf die unterschiedlichen Begriffe eingegangen, die für Interest-Points existieren. Es wird präzisiert, wie der Begriff des Interest-Points in der vorliegenden Arbeit verstanden wird. In Abschnitt 2.2 werden lokale Operatoren und ihre Arbeitsweise erläutert. Lokale Operatoren bilden die Basis für die Detektion von robusten Interest-Points. Eine besonders wichtige Rolle in diesem Zusammenhang spielt die Gauß-Funktion, auf deren Eigenschaften in den nachfolgenden Abschnitten kurz eingegangen wird.

2.1 Interest-Points

Für das Konzept des Interest-Points existieren in der Literatur zahlreiche Bezeichnungen und Definitionen (vgl. [73]). Schmid u.a. [63] beschreiben Interest-Points beispielsweise als Stellen im Bild, an denen sich das Bildsignal in zwei Dimensionen ändert. Andere Autoren verwenden den Ausdruck *saliente Punkte* (engl.: Salient Points) [65, 28]. Der Term Salienz stammt aus den Bereichen Neurobiologie und Psychologie und bezeichnet eine Eigenschaft eines Objekts, durch die es sich von benachbarten Objekten abhebt. Methoden zur Detektion salienter Punkte suchen nach Stellen in Bildern, die auf irgendeine Art visuell hervorstechen [70]. Wir Menschen können unsere Umgebung beispielsweise nicht als Ganzes visuell erfassen, sondern richten unsere Aufmerksamkeit unbewusst sofort auf bestimmte Punkte. Diese Punkte tragen den höchsten Informationsgehalt und sind essentiell, um schnellstmöglich Entscheidungen bezüglich unserer Umgebung zu treffen[1]. Methoden zur Detektion salienter Punkte versuchen solche Punkte in Bildern zu identifizieren [27]. In anderen Arbeiten werden die Begriffe *Schlüsselpunkte* (engl.: Keypoints) [38] oder *Fokus-Punkte* [70] verwendet.

In der vorliegenden Arbeit werden *Interest-Points* als Positionen verstanden, an denen sich interessante lokale Features befinden. Ein *lokales Feature* ist dabei nach Tuytelaars [75] definiert als Bildstruktur die sich stark von ihrer direkten Nachbarschaft unterscheidet und mit einer Änderung einer bestimmten Bildeigenschaft einhergeht. Bei den in dieser Arbeit untersuchten Interest-Point-Detektoren handelt es sich dabei um Änderungen der Bildintensität. Die gesuchten Bildstrukturen können dabei Eckpunkten oder blob-ähnlichen Strukturen entsprechen. In Kapitel 3 wird genauer auf Eckpunkte und Blobs eingegangen.

[1] Beispielsweise können so schnell lauernde Gefahren erkannt werden.



(a) (b)

Abbildung 2.1: Auswirkung der Änderung des Betrachtungswinkels beziehungsweise der Beleuchtung auf abgebildete Objekte (siehe [49])

Zudem zeichnet sich ein geeignetes lokales Feature nach [75] durch folgende Eigenschaften aus:

- Reproduzierbarkeit oder Wiederholbarkeit (engl.: Repeatability): Werden zwei Bilder desselben Objekts oder derselben Szene betrachtet, so soll ein möglichst hoher Anteil der detektierten Features in beiden Bildern detektiert werden, wenn sie auf Bildteilen liegen, die in beiden Bildern vorkommen. Das bedeutet, der Feature-Detektor soll möglichst robust gegenüber Bildtransformationen sein (siehe Abschnitt 2.1.1) [63].

- Informationsgehalt: Die lokale Struktur des Features soll genügend Informationen beinhalten, um einen automatisierten Bildabgleich zu ermöglichen. Die identifizierten Features müssen dazu möglichst einzigartig sein. Für Objekte desselben Typs sollte das Feature sehr ähnliche, für andere Objekttypen stark unterschiedliche Werte aufweisen.

- Effizienz: Die Detektion der Features soll möglichst effizient sein, sodass der Einsatz für zeitkritische Anwendungen möglich ist.

- Genauigkeit: Der Interest-Point soll eine wohldefinierte Position im Bildraum für das lokale Feature beschreiben. Zusätzlich möchte man häufig Informationen über die Größe des Features haben. Diese steht in engem Zusammenhang mit der betrachteten Skalierung (siehe Abschnitt 2.3).

- Quantität: Es sollen ausreichend viele Features in Bildern detektiert werden, sodass auch kleinere Objekte abgedeckt werden. Welche Anzahl von Features als ausreichend angesehen werden kann, hängt dabei hauptsächlich von der Anwendung ab.

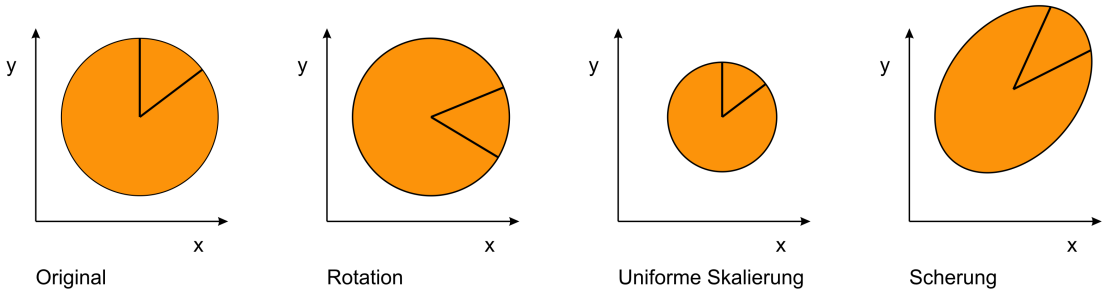


Abbildung 2.2: Darstellung der Auswirkung von verschiedenen affinen Transformationen auf ein durch ein Kreissegment repräsentiertes lokales Feature [30]

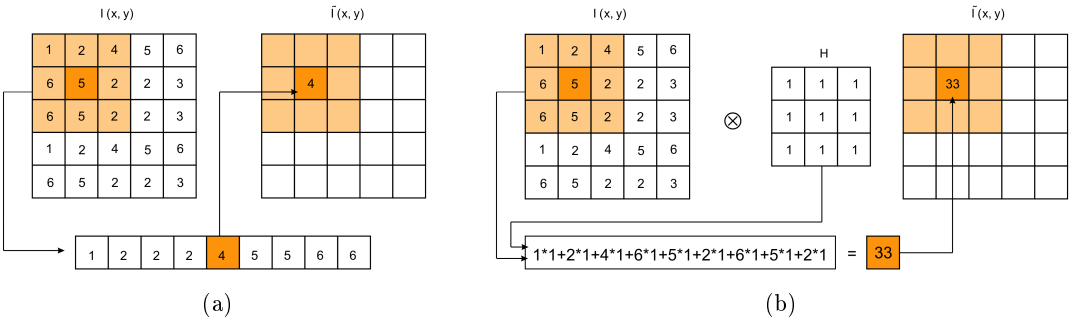
2.1.1 Bildtransformationen

Abbildung 2.1 zeigt zwei Bilder derselben Szene, die unter verschiedenen Bedingungen aufgenommen wurden [46]. Die Aufnahmen unterscheiden sich sowohl durch den Betrachtungswinkel als auch durch die Beleuchtungsbedingungen während der Aufnahme. Daraus resultiert auch ein Unterschied der in den Bildern gezeigten Objekte: die orange dargestellte Region, die den Buchstaben G umschließt verändert beispielsweise ihre Form und Ausdehnung. Selbst die Farbe des Buchstabens sieht in den beiden Bildern unterschiedlich aus. Möchte man nun mit Hilfe eines Interest-Point-Detektors dasselbe Objekt in beiden Bildern identifizieren, so ist das nur möglich, wenn der angewendete Detektor möglichst invariant ist, sowohl gegenüber geometrischen als auch photometrischen Transformationen [43].

Geometrische Transformationen sind Veränderungen des Bildes, die die Position beziehungsweise Anordnung der Bildelemente beeinflussen. *Photometrische Transformationen* sind Veränderungen der Intensitätswerte der Bildelemente. Zu den geometrischen Transformationen zählen beispielsweise Translation (Verschiebung), Rotation (Drehung) und Skalierung (uniforme Größenänderungen). Eine wichtige Gruppe der geometrischen Transformationen sind die affinen Transformationen. Affine Transformationen umfassen alle zuvor genannten Transformationen und die so genannte Scherung (nicht-uniforme Skalierung) [72, 21]. Abbildung 2.2 skizziert diese geometrischen Transformationen. Weiterführende Informationen zu Bildtransformationen können in [72] gefunden werden.

2.2 Lokale Operatoren

Die nachfolgenden Abschnitte bis inklusive Abschnitt 2.2.2.2 orientieren sich an [10]. *Lokale Operatoren* sind Operationen auf Bildern, die eine kleine Nachbarschaft $H(i, j)$ um ein Bildelement $I(x, y)$ im Eingangsbild mit einbeziehen, um einen neuen Intensitätswert $\tilde{I}(x, y)$ im Ergebnisbild zu bestimmen [10]. Sie werden auch als Nachbarschaftsoperatio-



(a) (b)

Abbildung 2.3: Anwendung von nichtlinearen und linearen Filtern [9] (a) Nichtlinearer Medianfilter; Werte innerhalb der Filterregion werden aufsteigend sortiert, der mittlere Wert wird ausgewählt (b) Linearer Mittelwertfilter

nen oder Filter bezeichnet[2]. Lokale Operatoren bilden die Grundlage zur Berechnung von Interest-Points, da sie beispielsweise eingesetzt werden können, um Intensitätsänderungen zu identifizieren.

Die Vorgehensweise bei der Anwendung lokaler Operatoren ist dabei folgende [9]: Über dem aktuell betrachteten Bildpunkt $I(x, y)$ wird ein so genannter Filterkern[3] $H(i, j)$ gelegt. Diesen Filterkern kann man sich als Matrix vorstellen, durch deren Größe und Form die Beschaffenheit der betrachteten Nachbarschaft um das aktuelle Bildelement $I(x, y)$ festgelegt wird. Die ausgewählte Nachbarschaft wird nach [10] auch als Filterregion bezeichnet. Der Filterkern $H(i, j)$ wird gewöhnlich so gewählt, dass sein Zentrum über dem aktuellen Bildelement platziert werden kann. Dazu eignen sich beispielsweise quadratische Filterkerne mit ungerader Seitenlänge in unterschiedlichen Größen (beispielsweise $3x3$ oder $5x5$). Im Folgenden wird das Zentrum eines Filterkerns durch den Punkt $\cdot$ gekennzeichnet. Der Ergebniswert $\tilde{I}(x, y)$ für die betrachtete Bildposition (x, y) wird durch eine Kombination der Intensitätswerte innerhalb der Filterregion berechnet. Der Vorgang wird für jedes Bildelement im Eingangsbild wiederholt indem der Filterkern einmal über jedes Bildelement gelegt wird.

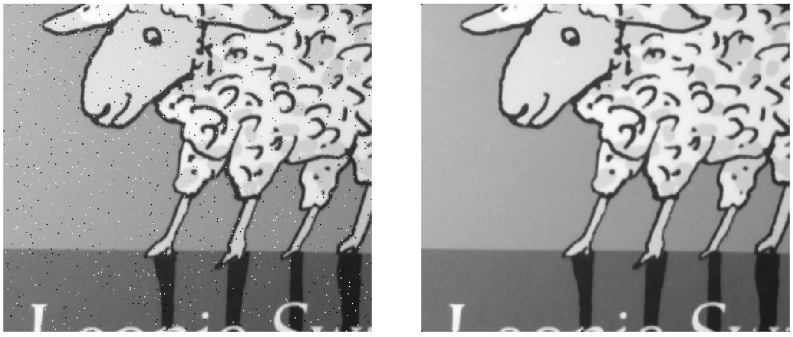
Lokale Operatoren werden in *nichtlineare Filter* (Abschnitt 2.2.1) und *lineare Filter* (Abschnitt 2.2.2) unterteilt, abhängig davon, ob die Intensitätswerte innerhalb der betrachteten Filterregion in linearer oder nichtlinearer Form miteinander kombiniert werden [10].

2.2.1 Nichtlineare Filter

Zu den nichtlinearen Filtern zählen beispielsweise der Maximum-, Minimum- und Medianfilter [9]. Der Maximum-Filter extrahiert den maximalen Intensitätswert aus der betrachteten Filterregion um den aktuellen Bildpunkt, der Minimum-Filter dementspre-

[2]Eine andere große Gruppe von Bildoperatoren sind die *Punktoperatoren*. Diese berechnen einen neuen Wert für ein Bildelement allein auf Basis des ursprünglichen Werts desselben Bildelements, d.h. $\tilde{I}(x, y) = f(I(x, y))$ [69].

[3]Häufig werden dafür auch die Ausdrücke Filtermaske, Faltungskern oder Operatorfenster verwendet.



(a) (b)

Abbildung 2.4: Anwendungsbeispiel eines $3x3$ Medianfilters: (a) Bild gestört durch Salz- und Pfeffer-Rauschen (kleine schwarze und weiße Punkte) (b) Ergebnisbild nach Anwendung des Medianfilters

chend den minimalen. Der Medianfilter sortiert alle Intensitätswerte in der Filterregion aufsteigend nach ihrem Intensitätswert und liefert den in der Mitte gelegenen Wert als Ergebnis zurück. In Abbildung 2.3a ist die Berechnung des Resultats eines $3x3$-Medianfilters für einen Bildpunkt dargestellt. Alle drei genannten Filter eignen sich sehr gut, um Störungen im Bild zu minimieren, ohne dabei vorhandene Strukturen verwaschen erscheinen zu lassen, wie das bei linearen Glättungsfiltern der Fall ist (siehe Abschnitt 2.2.2.1). Abbildung 2.4 zeigt, wie sich das Filtern mit einem Medianfilter auf ein mit Salz- und Pfeffer-Rauschen versetztes Bild auswirkt. In diesem Fall konnten die Störungen durch den Medianfilter fast vollständig entfernt werden.

Eine besondere Form nichtlinearer Filter stellen die *morphologischen Filter* dar. Diese basieren auf der Verwendung sogenannter Strukturelemente zur Extraktion relevanter Strukturen in Bildern. Beispiele für morphologische Operatoren sind die Dilatation, die Strukturen wachsen lässt und die entgegengesetzte Operation, genannt Erosion. Ausführliche Erklärungen verschiedener morphologischer Operatoren können in [69, 10] gefunden werden.

2.2.2 Lineare Filter

Bei *linearen Filtern* werden die Intensitätswerte innerhalb der Filterregion in linearer Form miteinander kombiniert, indem jedes Bildelement der Filterregion mit dem darüber liegenden Wert des Filterkerns multipliziert wird und die so erhaltenen Ergebnisse aufsummiert werden [10]. Die Filterkoeffizienten des eingesetzten Filterkerns legen bei linearen Filtern die Gewichtung der einzelnen Intensitätswerte fest. Durch unterschiedliche Gewichtungen können verschiedene Arten von linearen Filtern realisiert werden. In Abbildung 2.3b ist der Vorgang bei der linearen Filterung dargestellt [9]. Die mathematische Basis linearer Filter bildet die Faltung (engl.: convolution).

Diese ist im diskreten Fall definiert als [10]:

$$\tilde{I}(x,y) = \sum_i \sum_j I(x-i, y-j) \cdot H(i,j) \tag{2.1}$$

Durch Verwendung des Faltungsoperators $\otimes$ kann die Faltung der Funktionen $I(x,y)$ und $H(i,j)$ auch angeschrieben werden als [10]:

$$\tilde{I}(x,y) = I(x,y) \otimes H(i,j) \tag{2.2}$$

Dabei entspricht $I(x,y) = I$ dem Eingangsbild und $H(i,j) = H$ dem eingesetzten Filterkern. Die lineare Faltung weist einige sehr günstige Eigenschaften auf, wie beispielsweise Kommutativität und Assoziativität (siehe [10]).

Aus diesen Eigenschaften resultiert auch die sogenannte Separierbarkeit linearer Filter, die formal beschrieben werden kann als [10]:

$$I \otimes H = I \otimes (H_1 \otimes H_2 \otimes ... \otimes H_n) = (...((I \otimes H_1) \otimes H_2) \otimes ... \otimes H_n) \tag{2.3}$$

Durch die Separierbarkeit wird eine Schachtelung von Filterkernen ermöglicht, das heißt ein Filterkern H kann durch Faltung von mehreren Filterkernen $H_1, H_2, ..., H_n$ erzeugt werden. Die Separierbarkeit erweist sich bei der Bearbeitung von Bildern als besonders nützlich, weil dadurch ein zweidimensionaler Filterkern $H_{x,y}$, der in x- und y-Richtung operiert, durch zwei eindimensionale Filterkerne H_x und H_y ersetzt werden kann. Die Faltung mit den zwei kleineren eindimensionalen Filterkernen H_x, H_y bedeutet erheblich geringere Berechnungskosten[4] als die Faltung mit dem Filterkern $H_{x,y}$ [10].

Bei linearen Filtern unterscheidet man, je nach Anwendungsgebiet, zwischen Differenzfiltern und Glättungsfiltern. *Glättungsfilter* (Abschnitt 2.2.2.1) sind Filter, die das Bild unschärfer machen, indem sie eine Mittelung der Intensitätswerte innerhalb der betrachteten Filterregion durchführen. Sie unterdrücken Rauschen und lassen Strukturen in Bildern verwaschen erscheinen. *Differenzfilter* (Abschnitt 2.2.2.2) approximieren Ableitungen diskreter Bilder und heben lokale Intensitätsänderungen hervor, also Stellen, an denen sich die Intensität zwischen benachbarten Bildelementen sehr stark ändert [69, 10].

2.2.2.1 Lineare Glättungsfilter

Lineare Filter mit rein positiven Filterkoeffizienten berechnen eine Mittelung der Intensitätswerte in der betrachteten Filterregion [10]. Dadurch wird eine Glättung der in den Bildern vorhandenen Strukturen erwirkt. Die Gewichtung der einzelnen Intensitätswerte und somit der Grad der Glättung wird durch die Koeffizienten des Filterkerns bestimmt. Diese Filterkoeffizienten werden für gewöhnlich so gewählt, dass ihre Summe 1 ergibt. Dadurch ist gewährleistet, dass keine Intensitätswert außerhalb des ursprünglichen Wertebereichs erzeugt werden können.

Typische Glättungsfilter sind der Mittelwertfilter und der Gauß-Filter. Bei der Glättung mit dem Mittelwertfilter werden alle Werte innerhalb einer quadratischen Filterregion gleichmäßig gewichtet, das heißt jeder Filterkoeffizient hat denselben Wert [66]. Ein

[4] Die Rechenzeit bei lokalen Operatoren wächst quadratisch mit der Größe des Filterkerns [9].

Beispiel für einen $3x3$ Mittelwertfilter in zwei Dimensionen ist in Gleichung 2.4 zu sehen. Die Multiplikation mit dem Faktor $1/9$ bewirkt eine Skalierung der Ergebniswerte auf den gültigen Wertebereich.

$$H^{MW} = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1. & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{2.4}$$

Ein großer Nachteil des Mittelwertfilters besteht darin, dass er nicht richtungsunabhängig ist, was aber gerade für den Einsatz bei der Interest-Point-Detektion ein entscheidendes Kriterium darstellt um Invarianz gegenüber geometrischen Transformationen zu erreichen (Abschnitt 2.1.1) [10].

Wesentlich besser zur Durchführung einer Glättung geeignet ist daher der Gauß-Filter, dessen Filterkoeffizienten auf Basis der *Gauß-Funktion* bestimmt werden [66]. Die zweidimensionale Gauß-Funktion ist definiert als [69]:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.5}$$

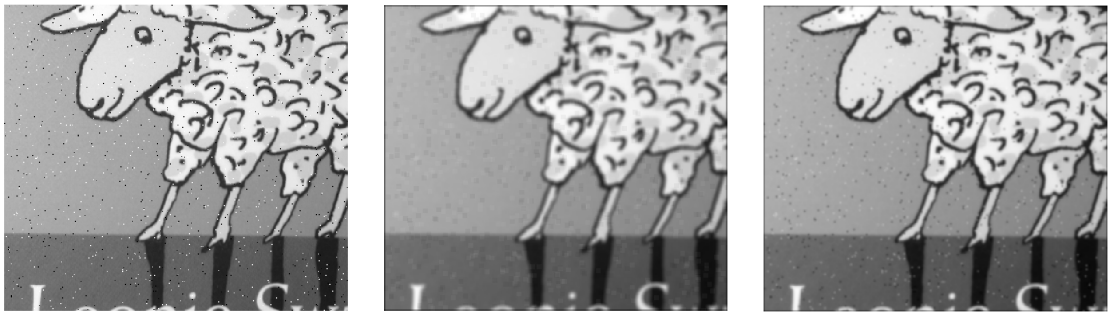
Die Form der Gauß-Funktion ist in Abbildung 2.8a dargestellt. Die Gauß-Funktion nimmt ein Maximum in der Mitte an und fällt dann auf allen Seiten gleichmäßig ab. Die Funktion ist kreisförmig symmetrisch und somit richtungsunabhängig. Der Parameter σ wird als Standardabweichung bezeichnet. Sie beeinflusst die Ausdehnung der Funktion [10]. Ist σ klein, so ist die Kurve sehr schmal und fällt schnell ab; ist σ hingegen groß, so ist die Kurve flacher. Die zweidimensionale Gauß-Funktion $G(x, y)$ kann als Produkt zweier einzelner Gauß-Funktionen berechnet werden und ist somit separierbar, das heißt es gilt [70, 10]:

$$G(x, y) = G(x)G(y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2}{2\sigma^2}} \frac{1}{2\pi\sigma^2} e^{\frac{y^2}{2\sigma^2}} \tag{2.6}$$

Bei der Konstruktion eines Gauß-Filters steht σ in engem Zusammenhang mit der Größe des Filterkerns[5] und somit auch mit dem Anteil der Bildelemente in der Umgebung des betrachteten Bildelements, der für die Glättung herangezogen wird [69]. Durch die Koeffizienten des Gauß-Filters werden die Bildelemente innerhalb der Filterregion unterschiedlich gewichtet. Das aktuelle Bildelement im Zentrum erhält das maximale Gewicht, die Gewichte für die anderen Bildelemente nehmen mit der Entfernung von der Mitte ab. Durch die unterschiedliche Gewichtung der Bildelemente können bessere Ergebnisse bei der Glättung erzielt werden, da lokale Strukturen nicht so stark durch weit weg liegende Intensitätswerte verfälscht werden [10].

Der Gauß-Filter ist, wie auch schon die Gauß-Funktion separierbar. Das bedeutet wiederum, dass der zweidimensionale Filterkern $H_{x,y}^{G(\sigma)}$ auf Basis der Gauß-Funktion als Faltung von zwei eindimensionalen Filtern $H_x^{G(\sigma)}$ und $H_y^{G(\sigma)}$ in x- und y- Richtung berechnet werden kann als [10]:

[5] Als geeignet haben sich Filterkerne mit einer Seitenlänge von -2.5σ bis $+2.5\sigma$ erwiesen [10]; Gauß-Filter können also in unterschiedlichen Größen erzeugt werden.



(a) (b) (c)

Abbildung 2.5: Anwendungsbeispiel linearer Glättungsfilter (a) Ein mit Salz-und Pfeffer-Rauschen gestörtes Bild (b) Ergebnis der Anwendung eines $5x5$ Mittelwertfilters (c) Ergebnis der Anwendung eines $5x5$ Gauß-Filters; Das Ergebnis der Anwendung des Mittelwertfilters ist deutlich verschwommener

$$H_{x,y}^{G(\sigma)} = \begin{bmatrix} 1 & 2 & 1 \\ 2 & \mathbf{4.} & 2 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ \mathbf{2.} \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & \mathbf{2.} & 1 \end{bmatrix} = H_x^{G(\sigma)} \otimes H_y^{G(\sigma)} \tag{2.7}$$

Der Gauß-Filter ist zudem isotrop, das heißt richtungsunabhängig. Gauß-basierte Filter bilden die Grundlage für die Bildung von isotropen Differenzfiltern (Abschnitt 2.2.3) und die Konstruktion von Skalenräumen (Abschnitt 2.3) und spielen daher eine wichtige Rolle bei der Interest-Point-Detektion. Abbildung 2.5 zeigt zum Vergleich das Ergebnis der Anwendung eines Mittelwertfilters und eines Gauß-Filters auf ein mit Salz- und Pfeffer-Rauschen gestörtes Bild.

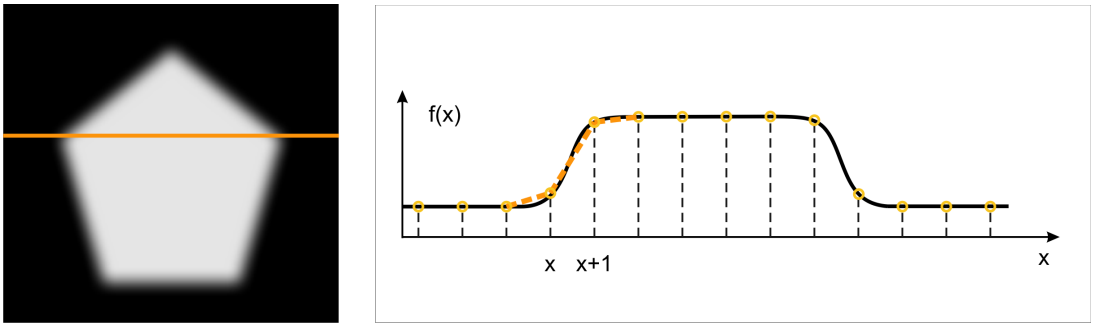
2.2.2.2 Lineare Differenzfilter

Für die Detektion von Interest-Points werden häufig Stellen in Bildern identifiziert, an denen starke *lokale Intensitätsänderungen* auftreten. Die Faltung mit einem linearen Differenzfilter hebt solche lokalen Intensitätsänderungen hervor.

Abbildung 2.6a zeigt ein helles Polygon auf dunklem Hintergrund (siehe [10]). Starke lokale Intensitätsänderungen treten hier an den Grenzen des Polygons auf, wenn sich die Intensitätswerte plötzlich von hell auf dunkel oder von dunkel auf hell ändern. Das Intensitätsprofil[6] der orange gekennzeichneten Bildzeile könnte wie die Funktion $f(x)$ in Abbildung 2.6b oben aussehen [10, 66].

Wir wissen, dass *Änderungen einer kontinuierlichen Funktionen* durch die Bildung von Ableitungen ermittelt werden können. So entsprechen Stellen, an denen starke lokale Intensitätsänderung auftreten, in der Ableitung erster Ordnung $f'(x)$ der Funktion $f(x)$

[6] Ein Intensitätsprofil ist ein eindimensionaler Schnitt durch ein Bild, wobei die Intensitätswerte entlang dieses Schnittes eine Funktion der Position sind [10].



(a) (b)

Abbildung 2.6: (a) Grauwertbild mit Schnitt entlang horizontaler Richtung (b) Intensitätsprofil $f(x)$ des Schnitts; Die Ableitung erster Ordnung in einem Punkt x kann im diskreten Fall approximiert werden, indem eine Gerade (orange gestrichelte Linie) durch zwei benachbarte Abtastwerte (gelbe Kreise) gelegt und die Steigung dieser Geraden geschätzt wird [10, 66]

beispielsweise lokalen Extrema und in der Ableitung zweiter Ordnung $f''(x)$ Nulldurchgängen [69].

Die *Ableitung erster Ordnung* $f'(x)$ einer Funktion beschreibt wie sehr sich der Funktionswert $f(x)$ ändert, wenn x um einen verschwindend kleinen Schritt $\triangle x$ verändert wird. Sie ist für kontinuierliche Funktionen in einer Variablen definiert als [40]:

$$f'(x) = \frac{df(x)}{dx} = \lim_{\triangle x \to 0} \frac{f(x + \triangle x) - f(x)}{\triangle x} \tag{2.8}$$

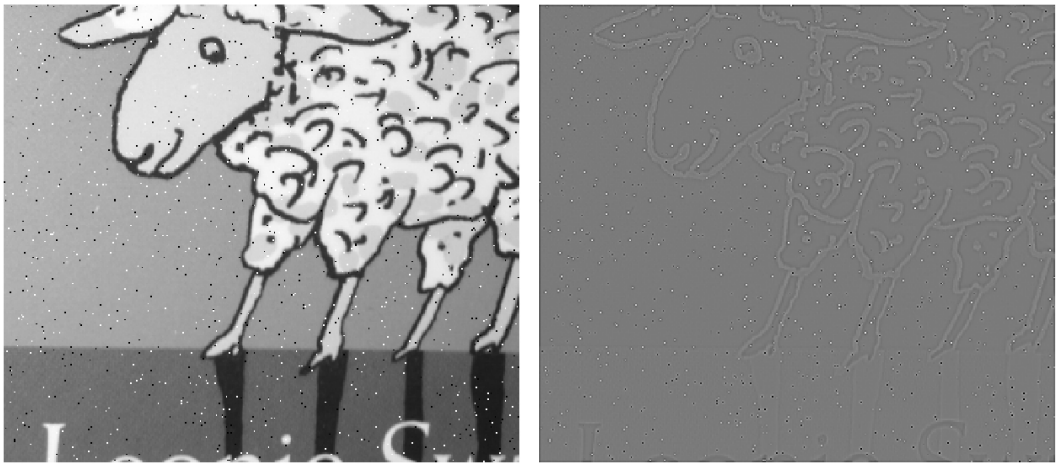
Da digitale Bilder diskret sind ist diese Ableitung nicht definiert und muss somit approximiert werden. Die Ableitung erster Ordnung in einem Punkt x kann geschätzt werden, indem eine Gerade durch zwei benachbarte Abtastwerte der Funktion gelegt wird und die Steigung dieser Geraden berechnet wird (siehe Abbildung 2.6b). Damit ergibt sich eine Schätzung für die *erste Ableitung einer diskreten Funktion* zu [66]:

$$\frac{df(x)}{dx} \approx \frac{f(x + \triangle x) - f(x)}{\triangle x} = \frac{f(x+1) - f(x)}{1} \tag{2.9}$$

Dabei steht $\triangle x$ für den Abstand zwischen den beiden Abtastwerten. Wenn man $\triangle x = 1$ wählt, so erhält man eine Schätzung der Ableitung erster Ordnung für eine diskrete Funktion durch Berechnung der Differenz benachbarter Abtastwerte in horizontaler Richtung[7]. Diese Differenzbildung kann durch Faltung des Eingangsbilds I mit einem horizontalen Filterkern der Form $\begin{bmatrix} -1 & \mathbf{1.} \end{bmatrix}$ realisiert werden [66]:

$$\frac{dI}{dx} \approx I \otimes \begin{bmatrix} -1 & \mathbf{1.} \end{bmatrix} \tag{2.10}$$

[7] In einem Bild entspricht das der Berechnung der Differenz der Intensitätswerte zweier nebeneinander liegender Bildelemente in einer Zeile.



(a) (b)

Figure 2.7: Anwendungsbeispiel eines linearen Differenzfilters: (a) Ein mit Salz-und Pfeffer-Rauschen gestörtes Bild (b) Ergebnis der Anwendung eines $3x3$ Laplace-Filters; die Störungen im Ausgangsbild werden durch Bildung der Ableitung verstärkt

Die Bildung der *Ableitung zweiter Ordnung* entspricht einer zweimaligen Ableitung erster Ordnung. Die Approximation der Ableitung zweiter Ordnung für eine diskrete Funktion kann daher beispielsweise wie zuvor bestimmt werden als [22]:

$$\begin{aligned} \frac{d^2 f(x)}{dx^2} &\approx \frac{d}{dx} f(x+1) - \frac{d}{dx} f(x) \\ &\approx (f(x+1) - f(x)) - (f(x) - f(x-1)) \\ &= f(x+1) - 2f(x) + f(x-1) \end{aligned} \tag{2.11}$$

Das entspricht einer Faltung mit dem Filterkern der Form $[1 - 2\,.\,1]$ [66]:

$$\frac{d^2 f(x)}{dx^2} \approx I \otimes [1 - 2\,.\,1] \tag{2.12}$$

Alle *Differenzfilter* bilden Schätzungen der ersten oder zweiten Ableitung eines Bildes [15]. Die Koeffizienten eines Differenzfilters nehmen, im Gegensatz zu den Koeffizienten der Glättungsfilter, auch negative Werte an. Sie werden meistens so gewählt, dass ihre Summe Null ergibt. Dadurch ist gewährleistet, dass die Filterantwort in Regionen homogener Intensität nicht groß wird und somit keine Intensitätsänderung detektiert wird [66]. Es existieren verschiedenste Filterkerne, die eine Ableitung diskreter Bilder durch Bildung von Differenzen simulieren. Gradientenfilter sind Filter, die die erste Ableitung eines zweidimensionalen Bildes approximieren, wohingegen Laplace-Filter die Ableitung zweiter Ordnung schätzen [15]. Auf diese beiden Gruppen von Differenzfiltern wird nun näher eingegangen.

Gradientenfilter Wir möchten dazu in der Lage sein, Intensitätsänderungen auf isotrope Weise[8] zu erkennen [25]. Der Gradient einer Funktion ist ein Vektor, der alle möglichen partiellen Ableitungen der Funktion beinhaltet. Für zweidimensionale Bilder wird der Vektor $\nabla I(x,y)$ als Gradient des Bildes I an der Stelle (x,y) bezeichnet und ist definiert als [10]:

$$\nabla I(x,y) = \left[\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right] \tag{2.13}$$

Der Gradient zeigt in Richtung der stärksten Änderung der Funktion $I(x,y)$. Er eignet sich daher gut zur Identifizierung von lokalen Intensitätsänderungen in Bildern. Der Betrag und der Winkel des Gradienten liefern Informationen zur Stärke und Richtung der detektierten Intensitätsänderung (siehe [69]).

Um den Bildgradienten zu bestimmen, müssen wir die partiellen Ableitungen erster Ordnung in x- und y-Richtung approximieren. Dies kann im einfachsten Fall mittels Differenzbildung zwischen benachbarten Bildelementen in horizontaler und vertikaler Richtung realisiert werden [66]:

$$\frac{\partial I(x,y)}{\partial x} \approx I \otimes H^{\nabla_x} = I \otimes \begin{bmatrix} -1 & 1_{\bullet} \end{bmatrix} \tag{2.14}$$

$$\frac{\partial I(x,y)}{\partial y} \approx I \otimes H^{\nabla_y} = I \otimes \begin{bmatrix} -1 \\ 1_{\bullet} \end{bmatrix} \tag{2.15}$$

Ein etwas weiter fortgeschrittener Filter zur Approximation des Gradienten ist der sogenannte Sobel-Operator. Dieser setzt sich aus zwei Filterkernen H^{S_x} und H^{S_y} zusammen, die jeweils die Ableitungen in horizontale und vertikale Richtung schätzen.

$$H^{S_x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0_{\bullet} & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad H^{S_y} = \begin{bmatrix} -1 & -2 & -1 \\ -0 & 0_{\bullet} & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{2.16}$$

Der Sobel-Operator setzt sich aus quadratischen Filterkernen der Größe $3x3$ zusammen, was einen Vorteil im Bezug auf die Anfälligkeit gegenüber Bildstörungen im Vergleich zu einzeiligen Differenzfiltern mit sich bringt [10]. Weitere typische Gradientenfilter sind der Roberts-Operator und der Prewitt-Operator. Genauere Informationen zu den einzelnen Filtern kann man in [69] nachlesen.

Laplace-Filter Für die Detektion lokaler Intensitätsänderungen auf Basis der zweiten Ableitung setzt man Differenzfilter ein, die den Laplace-Operator approximieren [15]. Der Laplace-Operator ist die Summe der partiellen Ableitungen zweiter Ordnung und ist definiert als [40]:

$$\nabla^2 I(x,y) = \frac{\partial^2 I(x,y)}{\partial x^2} + \frac{\partial^2 I(x,y)}{\partial y^2} \tag{2.17}$$

[8] Die zuvor vorgestellten Differenzfilter finden vorrangig Intensitätsänderungen, die in horizontaler Richtung auftreten.

Der Laplace-Operator arbeitet in alle Richtungen gleich und ist daher rotationsinvariant. Er beinhaltet, anders als der Gradient, Informationen über die Stärke der Intensitätsänderung, nicht aber über deren Richtung [69].

Die Ableitungen zweiter Ordnung können im diskreten Fall wiederum durch Differenzbildung mittels Faltung approximiert werden, beispielsweise durch [66]:

$$\frac{\partial^2 I(x,y)}{\partial x^2} \approx I \otimes H^{\nabla_x^2} = I \otimes [1 \; -2\,\mathbf{.} \; 1] \tag{2.18}$$

$$\frac{\partial^2 I(x,y)}{\partial y^2} \approx I \otimes H^{\nabla_y^2} = I \otimes \begin{bmatrix} -1 \\ \mathbf{2.} \\ 1 \end{bmatrix} \tag{2.19}$$

Ein Filter, der den Laplace-Operator approximiert, kann dann aus diesen beiden Filtern zusammengesetzt werden. Beispiele für Laplace-Filter der Größe $3x3$ sind folgende Filterkerne [69]:

$$H^{\nabla^2} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & \mathbf{-4.} & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad H^{\nabla^2} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & \mathbf{-8.} & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{2.20}$$

Aufgrund der zweimaligen Ableitung die beim Laplace-Filter durchgeführt wird, reagiert dieser extrem empfindlich auf Rauschen in Bildern. In Abbildung 2.7b ist dieser Effekt illustriert.

2.2.3 Gaußsche Ableitungsfilter

Differenzfilter haben den ungewollten Effekt neben Intensitätsänderungen auch Bildrauschen hervorzuheben [72]. Um diesem Problem entgegenzuwirken, wird bei der Approximation von Ableitungen häufig zuerst eine Glättung durchgeführt und danach erst ein Differenzfilter angewendet [40].

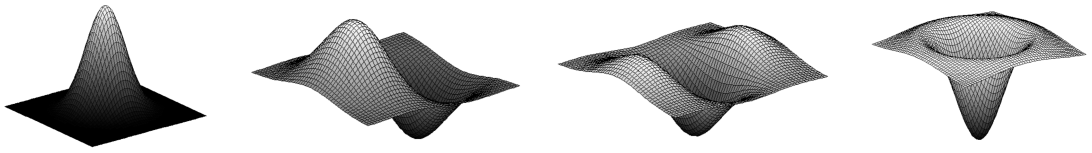
Da man im Allgemeinen eine richtungsunabhängige Detektion von Intensitätsänderungen anstrebt, liegt es nahe, zu diesem Zweck einen symmetrischen Glättungsfilter[9] einzusetzen. Der Gauß-Filter ist der einzige separierbare symmetrische Glättungsfilter [72]. Sowohl Differenzierung als auch Glättung sind lineare Operationen und daher kommutativ, also vertauschbar [72]. Die Operationen zur Berechnung des Bildgradienten beziehungsweise des Laplace-Operators mit vorgeschalteter Gauß-Glättung entsprechen daher folgenden Faltungen [69]:

$$\nabla \left[H^{G(\sigma)} \otimes I(x,y) \right] = [\nabla H^{G(\sigma)}] \otimes I(x,y) \tag{2.21}$$

$$\nabla^2 \left[H^{G(\sigma)} \otimes I(x,y) \right] = [\nabla^2 H^{G(\sigma)}] \otimes I(x,y) = H^{LoG(\sigma)} \otimes I(x,y) \tag{2.22}$$

Dabei wird die Kombination des Laplace-Operators mit der Gauß-Funktion $[\nabla^2 H^{G(\sigma)}]$ als *Laplacian Of Gaussian* (kurz: LoG) bezeichnet.

[9] das heißt einen Filter, der in alle Richtungen gleich arbeitet



(a) (b) (c) (d)

Abbildung 2.8: Darstellung der 3D-Form: (a) Gauß-Funktion $G(x, y)$; (b) partielle Ableitung 1. Ordnung der Gauß-Funktion in x-Richtung $\frac{\partial G(x,y)}{\partial x}$; (c) partielle Ableitung 2. Ordnung in x-Richtung $\frac{\partial^2 G(x,y)}{\partial x^2}$; (d) LoG-Operator $\frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2}$

Es ist somit möglich, Glättung und Bildung der Ableitung in einem einzigen Schritt durchzuführen, indem Filter verwendet werden deren Gewichte den Ableitungen der Gauß-Funktion (Gleichung 2.5) entsprechen. Wir bezeichnen im Folgenden Filter der Form $H^{G_{im}(\sigma)}$ nach [43] als Gaußsche Ableitungsfilter, wobei m die Ordnung der Ableitung und i die Richtung der Ableitung angibt. *Gaußsche Ableitungsfilter* berechnen gewichtete Differenzen von Intensitätswerten in einer Filterregion und können als verallgemeinerte Form von Differenzfiltern angesehen werden [43]. Sie werden häufig bei der Detektion von Interest-Points eingesetzt.

Die Filterkoeffizienten der Filter $H^{G_{im}(\sigma)}$ werden mit Hilfe von abgeleiteten Gauß-Funktionen berechnet. Die partiellen Ableitungen erster und zweiter Ordnung der zweidimensionalen Gauß-Funktion (siehe Gleichung 2.5) sind gegeben durch [69]:

$$\frac{\partial G(x,y)}{\partial x} = -\frac{x}{2\pi\sigma^4} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} \qquad \frac{\partial gG(x,y)}{\partial y} = -\frac{y}{2\pi\sigma^4} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)}$$

$$\frac{\partial^2 G(x,y)}{\partial x^2} = \frac{x^2-\sigma^2}{2\pi\sigma^6} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} \qquad \frac{\partial^2 G(x,y)}{\partial y^2} = \frac{y^2-\sigma^2}{2\pi\sigma^6} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)}$$

Die Filterkoeffizienten des LoG-Filters $H^{LoG(\sigma)}$ werden durch den LoG-Operator bestimmt, der folgendermaßen definiert ist [69]:

$$LoG = \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2} = \frac{x^2+y^2-2\sigma^2}{2\pi\sigma^6} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} \tag{2.23}$$

Abbildung 2.8 zeigt die Form der Gauß-Funktion und ihrer Ableitungen. Die Form dieser Funktionen steht bei der Detektion von Interest-Points in engem Zusammenhang mit der Struktur der lokalen Features die vom jeweiligen Detektor identifiziert wird. So wird der LoG-Filter beispielsweise häufig zur Detektion blob-ähnlicher Strukturen eingesetzt (Abschnitt 3.1.2). In Abschnitt 4.1.3 wird näher darauf eingegangen.

2.3 Gaußsche Skalenräume

Betrachtet man ein Objekt in seiner Umgebung dann ändern sich die erkennbaren Details mit dem Abstand den man zum Objekt einnimmt. Sieht man sich beispielsweise einen Tisch aus der Nähe an, so kann man dessen Maserung detailliert erkennen. Betrachtet man denselben Tisch aus größerer Entfernung, kann man keine feinen Details mehr erkennen [32].

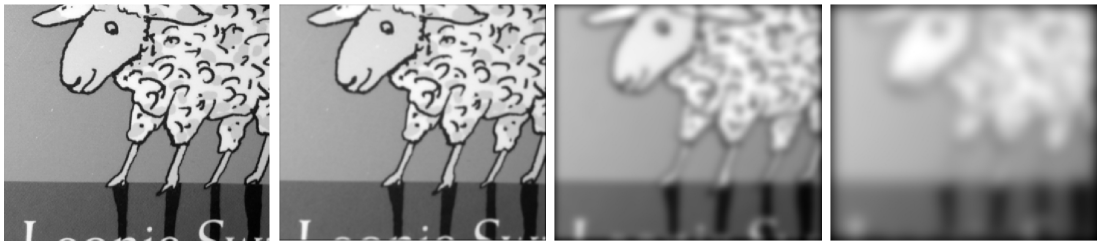
Erzeugt man nun ein Fotografie eines Objekts, so kann man denselben Effekt beobachten: Ist der Abstand zwischen Kamera und Objekt bei der Aufnahme gering, so kann man in dem erstellten Bild viele Details erkennen. Wenn der Abstand zwischen Kamera und Objekt hingegen groß ist, kann man kaum noch Details erkennen. In Abhängigkeit von den erkennbaren Details kann man auch sagen, dass ein Bild fein oder grob skaliert ist [32].

In der computergestützten Verarbeitung von Bildern möchten wir in der Lage sein, gleiche oder ähnliche Objekte in verschiedenen Aufnahmen durch einen Vergleich von Bildern zu erkennen (Abschnitt 1). Das wird dann problematisch, wenn die Vergleichsobjekte und die daraus extrahierten lokalen Features in unterschiedlichen Skalierungen beziehungsweise Auflösungen vorliegen. Skalenräume bieten uns die Möglichkeit, Bilder (und somit auch die enthaltenen Features) in verschiedenen Skalierungen zu repräsentieren[10] und helfen so, diesem Problem entgegenzuwirken. Sie werden bei vielen Interest-Point-Detektoren eingesetzt, um Robustheit der Verfahren gegenüber Skalierungen zu ermöglichen [73].

Grundsätzlich wird bei der Konstruktion eines Skalenraums aus einem Eingangsbild, das in einer bestimmten Skalierung vorliegt, ein Stapel von Bildern erzeugt. Diese Bilder repräsentieren dann verschiedene, gröbere Skalierungen des Eingangsbildes. Es wird also versucht, den Effekt des sich immer weiter Entfernens, also den Übergang von einer feinen zu einer gröberen Auflösungsstufe nachzubilden. Dazu benötigt man ein Verfahren, das immer mehr Details beziehungsweise Strukturen in Bildern verschwinden lässt. Das Verfahren soll von einem sogenannten Skalenparameter abhängig sein, der die simulierte Distanz zwischen Kamera und Objekt steuert. Je größer der Wert des Skalenparameters ist, desto mehr Strukturen sollen verschwinden [62]. Wir wissen, dass ein solcher Effekt durch Faltung mit einem Glättungsfilter realisiert werden kann (Abschnitt 2.2.2.1).

Glättungsfilter, die zum Aufbau von Skalenräumen geeignet sind, müssen dabei zwei grundlegende Anforderungen erfüllen [25, 62]: das Minimum-Maximum-Prinzip und die Halbgruppeneigenschaft. Das *Minimum-Maximum-Prinzip* besagt, dass durch die Faltung mit dem Glättungsfilter keine neuen Strukturen entstehen sollen. Der Informationsgehalt im Bild soll mit steigendem Skalenparameter abnehmen. Die *kommutative Halbgruppeneigenschaft* besagt, dass n aufeinander folgende Glättungsoperationen zum selben Resultat führen sollen wie eine Glättung mit einem Filterkern, dessen Größe der Summe aller n Filterkerne entspricht. Die Glättungsoperationen sollen zudem in beliebiger Reihenfolge ausführbar sein.

[10] Sie werden daher auch als Multiskalen-Repräsentationen bezeichnet [43].



(a) (b) (c) (d)

Abbildung 2.9: Bilder auf verschiedenen Skalenebenen: (a) Originalbild mit $\sigma = 0$ (b) Glättung mit $\sigma = 1$ (c) Glättung mit $\sigma = 5$ (d) Glättung mit $\sigma = 10$

Das heißt es soll gelten [43]:

$$I(x,y) \otimes H^{G(\sigma_1)} \otimes ... \otimes H^{G(\sigma_n)} = I(x,y) \otimes H^{G(\sigma_1+...+\sigma_n)} \tag{2.24}$$

Es wurde gezeigt [31, 32], dass Filterkerne auf Basis der Gauß-Funktion (siehe Abschnitt 2.2.2.1) oder auf Basis ihrer Ableitungen (siehe Abschnitt 2.2.3) in der Praxis als einzige für die Konstruktion von Skalenräumen geeignet sind [43]. Die verschiedenen Ebenen des Skalenraums[11] können daher durch wiederholte Faltung des Bildes mit immer größeren Gauß-Filtern gebildet werden [29]:

$$L(x,y,\sigma) = I(x,y) \otimes H^{G(\sigma)} \tag{2.25}$$

Die so erzeugten Skalenräume werden auch als Gaußsche Skalenräume bezeichnet. Gaußsche Skalenräume $L(x,y,\sigma)$ sind nach [25] dreidimensionale Datenstrukturen, deren Dimensionen den Ortskoordinaten des Bildes und dem Skalenparameter σ entsprechen. Sie bestehen aus mehreren Ebenen, wobei sich auf jeder der Ebenen eine durch Glättung mit einem gauß-basierten Filter veränderte Version des Eingangsbildes befindet. Die Standardabweichung σ des gauß-basierten Filterkerns $H^{G(\sigma)}$ steuert dabei den Grad der Glättung auf jeder Ebene und entspricht somit dem Skalenparameter. Strukturen die wesentlich kleiner sind als σ werden weg-geglättet [40]. Abbildung 2.9 zeigt einige Ebenen eines solchen Skalenraums. Das Ausgangsbild $I(x,y)$ in seiner ursprünglichen Skalierung befindet sich in einem Skalenraum auf der Ebene σ_0. Die Standardabweichung für die verschiedenen Skalenebenen kann durch

$$\sigma_n = k^n \sigma_0 \tag{2.26}$$

bestimmt werden, wobei n die Ebene des Skalenraums bezeichnet und k die Änderung der Größe der eingesetzten Gauß-Filter zwischen zwei aufeinander folgenden Skalenebenen festlegt [29].

Prinzipiell werden nach Mikolayczyk [43] unter dem Begriff Skalenraum häufig zwei verschiedene Arten von Multiskalen-Repräsentationen zusammengefasst: Bei den sogenannten *Bildpyramiden* wird durch Glättung des Bildes auf einer feineren Skalenebene ein

[11] d.h. die einzelnen Bilder des Stapels

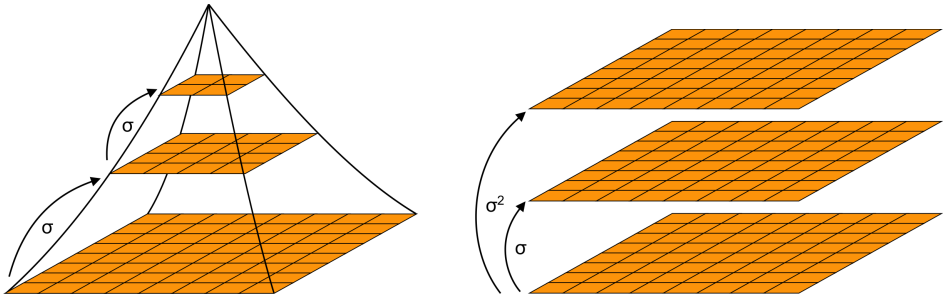


Abbildung 2.10: Aufbau von Skalenräumen: (a) Bildpyramide (b) linearer Skalenraum [43]

Bild mit gröberer Skalierung erzeugt (siehe Abbildung 2.10a). Dieser Vorgang wird dann auf der so entstandenen gröberen Skalenebene wiederholt und so weiter. Beim Übergang von einer feineren auf eine gröbere Skalenebene wird zudem eine Reduktion der Auflösung des Bildes durchgeführt [43]. Diese Vorgehensweise ermöglicht eine Minimierung des Rechenaufwandes bei der Konstruktion der Skalenräume [25]. *Lineare Skalenräume* (siehe Abbildung 2.10b) werden hingegen erzeugt, indem das Ursprungsbild nach und nach mit immer größer werdenden Filterkernen geglättet wird. Diese Methode ist zwar rechnerisch aufwändiger, hat aber den Vorteil, dass sich die Bilder des Skalenraums nicht in der Größe unterscheiden. Somit können die Bilder auf unterschiedlichen Skalenebenen leichter miteinander verglichen werden [43]. Falls nicht anders erwähnt sind im Folgenden lineare Skalenräume gemeint, wenn von Skalenräumen gesprochen wird.

Zur Detektion von lokalen Features werden häufig ableitungsbasierte Funktionen über einem Skalenraum mit Hilfe von Gaußschen Ableitungsfiltern berechnet. Weitere Informationen dazu finden sich in Abschnitt 4.2.1. Eine ausführliche Erläuterung zu Skalenräumen wurde von Lindeberg verfasst und kann in [35] gefunden werden.

3 Literaturüberblick

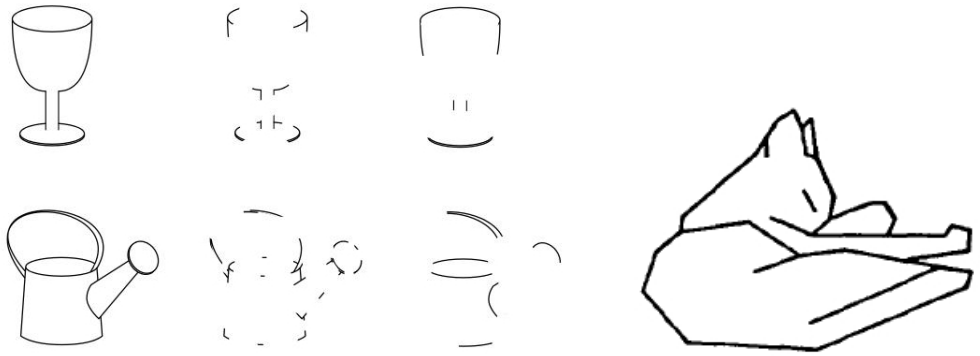
Die Idee zur Entwicklung von Interest-Point-Detektoren basiert auf Beobachtungen zum menschlichen Sehprozess und dem Versuch, diesen für die computergestützte Verarbeitung von Bildern zu adaptieren. Wie in Abschnitt 2.1 ausgeführt, ist bekannt, dass das menschliche Auge auf bestimmte Punkte beziehungsweise Strukturen unbewusst mehr reagiert als auf andere [70]. Man hat mitunter festgestellt, dass das menschliche Auge sehr gut auf Diskontinuitäten in seinem Blickfeld reagiert [27]. Als Diskontinuitäten bezeichnet man abrupte Intensitätsänderungen, die beispielsweise das Resultat von Tiefenunterschieden oder sich ändernden Materialeigenschaften sind.

Weiters wurde schon in den fünfziger Jahren von Attneave [1] und später von Biederman [6] gezeigt, dass für uns beim Erkennen von Objektformen vor allem auf Objektkonturen liegende Punkte mit hoher Krümmung eine wichtige Rolle spielen [75]. Dies wird in Abbildung 3.1 verdeutlicht [1, 6]: Werden aus den dargestellten Linienbildern Punkte entfernt (Abbildung 3.1a) oder ersetzt (Abbildung 3.1b), die eine niedrige Krümmung aufweisen, so sind wir dennoch in der Lage die dargestellten Objekte zu erkennen. Entfernt man hingegen Konturpunkte mit hoher Krümmung, so können Objektformen nur mehr schwer identifiziert werden (Abbildung 3.1a rechts).

Die ersten Interest-Point-Detektoren wurden bereits in den späten 1970er Jahren vorgestellt [75]. Die Entwicklung von Verfahren zur Extraktion von interessanten lokalen Features war und ist eine grundlegende Thematik in der computergestützten Verarbeitung von Bildern. Dementsprechend umfangreich und vielfältig ist die in diesem Kontext veröffentlichte Literatur.

Im Folgenden wird in Abschnitt 3.1 ein Überblick über die am häufigsten eingesetzten *Methoden zur Interest-Point-Detektion* gegeben, über die dazu vorhandene Literatur und die Entwicklung der Detektoren. Es werden einige Aspekte vorgestellt, anhand derer verschiedene Kategorien von Interest-Point-Detektoren unterschieden werden können. Eine umfassende Zusammenfassung existierender Verfahren zur Interest-Point-Detektion kann in [75] gefunden werden. In Abschnitt 3.2 werden kurz gängige *Methoden zur Evaluierung von Interest-Point-Detektoren* vorgestellt. Eine ausführliche Übersicht der Techniken, die zur Evaluierung von Feature-Detektoren eingesetzt werden können und der bereits durchgeführter Evaluierungen findet sich in [64] und [75].

Für eine geeignete Weiterverarbeitung von detektierten Interest-Points spielen Feature-Deskriptoren eine wichtige Rolle. Diese sind allerdings nicht Gegenstand der vorliegenden Arbeit. Informationen zu verschiedenen Feature-Deskriptoren sind beispielsweise in [72] und [45] zu finden. Die nachfolgenden Abschnitte orientieren sich an [75].



(a) (b)

Abbildung 3.1: Darstellung der Bedeutung von Konturpunkten mit starker Krümmung für die Objekterkennung: (a) Links: Linienzeichnung im Original; Mitte: nach Entfernung von Konturpunkten mit niedriger Krümmung; Rechts: nach Entfernung von Konturpunkten mit hoher Krümmung [6] (b) Zeichnung einer Katze bei der Punkte niedriger Krümmung durch gerade Linien ersetzt wurden [1]

3.1 Methoden zur Interest-Point-Detektion

Bei Verfahren zur Detektion von Interest-Points kann man nach [63] und [73] grundsätzlich zwischen intensitätsbasierten-, konturbasierten- und modellbasierten Verfahren unterscheiden. *Konturbasierte Verfahren* [23] extrahieren Objektkonturen aus Bildern und identifizieren dann beispielsweise Stellen mit hoher Krümmung als Interest-Points. Bei *modellbasierten Methoden* [57] werden bestimmte Bildstrukturen mit Hilfe eines parametrischen Modells identifiziert. Dabei simuliert das jeweilige Modell die Struktur des lokalen Features, das gefunden werden soll. Modellbasierte Methoden werden vor allem dann eingesetzt, wenn ganz spezielle Bildstrukturen identifiziert werden sollen [73, 63].

Intensitätsbasierte Verfahren lokalisieren Interest-Points direkt auf der Basis von lokalen Intensitätsänderungen im Bildsignal [73]. Sie verarbeiten Grauwertbilder. Intensitätsbasierte Verfahren werden in der Praxis am häufigsten eingesetzt, weil ihre Anwendung keinerlei besonderer Voraussetzungen bedarf und daher auf verschiedene Bildtypen möglich ist [75]. Aus diesem Grund wird in der vorliegenden Arbeit ausschließlich auf intensitätsbasierte Verfahren eingegangen. In letzter Zeit wurden auch Methoden zur Identifikation von Interest-Points entwickelt, die Farbbilder verarbeiten können [50, 18, 76, 70]. Nach [75] sind diese Verfahren jedoch für gewöhnlich nur Erweiterungen intensitätsbasierter Verfahren. Farbinformationen werden lediglich verwendet, um die Stabilität der Detektoren zu verbessern.

Die Vorgehensweise bei intensitätsbasierten Interest-Point-Detektoren ist häufig ähnlich: Es wird eine *Interest-Funktion* für jedes Bildelement des betrachteten Eingangsbildes berechnet und so ein neues Bild erzeugt. Einzelne Detektoren unterscheiden sich dann im Wesentlichen durch die jeweils eingesetzte Interest-Funktion. Wir verstehen im Nach-

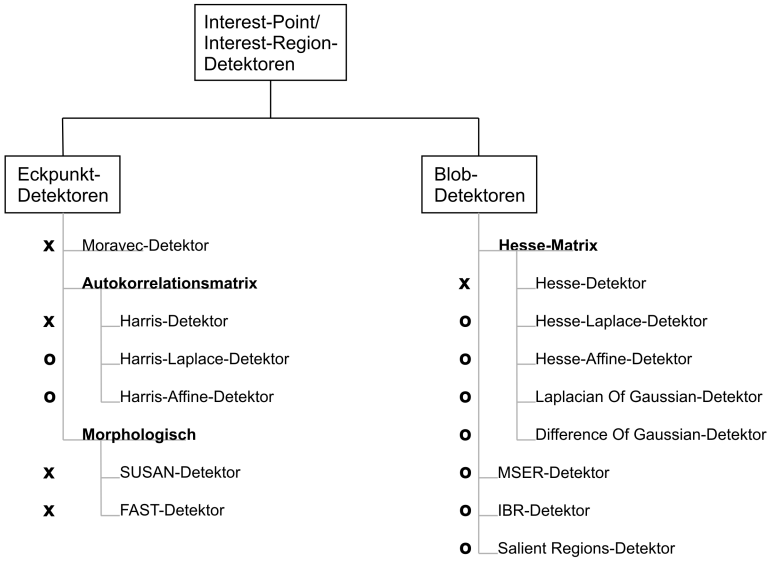


Abbildung 3.2: Übersicht von Verfahren zur Detektion von lokalen Features. Das Zeichen vor dem jeweiligen Detektor gibt an ob es sich um einen Interest-Point-Detektor (**x**) oder einen Interest-Region-Detektor (**o**) handelt [4]

folgenden unter einer Interest-Funktion eine Funktion die so definiert ist, dass sie ein lokales Extremum an den Stellen im erzeugten Bild annimmt, an denen das gesuchte lokale Feature vorliegt. Sie liefert also ein Maß für die Wahrscheinlichkeit des Vorliegens der gewünschten lokalen Struktur an jeder Bildposition. In Abhängigkeit von der lokalen Struktur, die vom jeweiligen Detektor identifiziert werden soll, wird zwischen *Eckpunkt-* und *Blob-Detektoren* unterschieden (siehe Abbildung 3.2) [75].

Die angewendeten Interest-Funktionen zur Identifikation der lokalen Features sind häufig ableitungsbasiert. Interest-Funktionen, die zur Detektion von Eckpunkten eingesetzt werden, sind zumeist als *Kombination partieller Ableitungen erster Ordnung* definiert und werden auf Basis der sogenannten *Autokorrelationsmatrix* (Abschnitt 4.1.2) berechnet. Für *Blob-Detektoren* hingegen werden häufig *Kombinationen partieller Ableitungen zweiter Ordnung* aus der *Hesse-Matrix* (Abschnitt 4.1.3) eingesetzt.

Heute liegt der Schwerpunkt vor allem auf der Entwicklung von Interest-Point-Detektoren die invariant gegenüber verschiedenen geometrischen Transformationen sind[1] (Abschnitt 2.1.1). Dabei lässt sich zunächst ein Trend zu skalierungsinvarianten und später zu affin-invarianten Detektoren erkennen [24]. Diese Methoden sind oftmals Erweiterungen einfacher ableitungsbasierter Detektoren. Im Nachfolgenden wird zunächst auf einfache Eckpunkt-Detektoren (Abschnitt 3.1.1) und Blob-Detektoren (Abschnitt 3.1.2) eingegangen. In Abschnitt 3.1.3 werden dann Methoden zusammengefasst, die robust

[1]Obwohl in der Literatur zumeist der Begriff *invariant* verwendet wird, wäre der korrekte Begriff hier eigentlich kovariant (siehe [75]).

beziehungsweise invariant gegenüber Skalierungsveränderungen und affinen Bildtransformationen sind. Der Fokus in den nachfolgenden Kapiteln liegt auf denjenigen Detektoren, die auch in die entwickelte Applikation integriert wurden.

3.1.1 Eckpunkt-basierte Verfahren

Die ersten Verfahren zum Auffinden von Interest-Points waren Eckpunkt-Detektoren, das heißt sie waren darauf ausgelegt, Eckpunkte in Bildern zu identifizieren. In diesem Zusammenhang versteht man unter dem Begriff Eckpunkt Stellen im Bild, an denen eine *signifikante Intensitätsänderung in mindestens zwei Richtungen* auftritt [63].

Ursprüngliches Ziel bei der Interest-Point-Detektion war es, robuste, exakt lokalisierbare Stellen in Bildern zu finden. Diese waren beispielsweise für den Einsatz bei der Objektverfolgung oder als Referenzpunkte zur Kalibrierung von Kamerasystemen gedacht [10, 75]. Eckpunkte sind solche Stellen. Sie sind die am höchsten strukturierten Orte im Bild [71] und sind stabil unter bestimmten geometrischen und photometrischen Transformationen [10]. Eckpunkte sind zudem exakt lokalisierbar, da sie durch einen einzelnen Punkt identifiziert werden können [75].

Wir wissen, dass für die automatisierte Verarbeitung von Bildern oftmals ein Bildabgleich die Grundlage bildet (Abschnitt 1). Abbildung 3.3 zeigt zwei Bilder derselben Szene, die sich durch die jeweiligen Bedingungen bei der Aufnahme unterscheiden [72]. Im Bild links sind drei unterschiedliche Bildausschnitte markiert, die zur Durchführung eines Bildabgleichs der beiden dargestellten Bilder eingesetzt werden sollen. Die Bildausschnitte sollten daher so beschaffen sein, dass sie gut in der Abbildung rechts lokalisiert werden können. Die ausgewählten Bildausschnitte sind in der Mitte der Abbildung vergrößert dargestellt und zeigen eine homogene Region, eine Kante und einen Eckpunkt. Es ist intuitiv klar, dass der Bildausschnitt der den Eckpunkt zeigt am besten für den Bildabgleich geeignet ist, da er als einziger exakt in beiden Bildern von Abbildung 3.3 lokalisiert werden kann [72].

Einer der ersten Eckpunkt-Detektoren war der 1977 vorgestellte *Moravec-Detektor* [51]. Dieser, ursprünglich zum Navigieren mobiler Roboter entwickelte Detektor, identifiziert Eckpunkte indem untersucht wird, wie sich die Intensität in verschiedene Richtungen in der Nachbarschaft eines betrachteten Bildelements ändert (siehe Abschnitt 4.1.1). Grundlage zur Berechnung der Intensitätsänderungen ist eine Differenzbildung definiert durch das mittlere Abstandsquadrat.

Im Jahr 1988 stellten Harris und Stephens eine Weiterentwicklung des Moravec-Detektors vor, den so genannten *Harris-Detektor* [20] (siehe Abschnitt 4.1.2). Beim Harris-Detektor werden Intensitätsänderungen auf Basis der Autokorrelationsmatrix bestimmt. Dabei wird aus der Autokorrelationsmatrix eine gradientenbasierte Interest-Funktion berechnet, die eine richtungsunabhängige Detektion von Eckpunkten ermöglicht. Ähnliche Verfahren wurden von Förstner [16] und Shi u.a. [67] entwickelt. Es hat sich gezeigt, dass die mit Hilfe des Harris-Operators identifizierten Eckpunkte nicht nur invariant gegenüber Rotationen, sondern auch gegenüber Translationen und Veränderungen der Beleuchtungsbedingungen in kleinerem Ausmaß sind [75]. Der Harris-Detektor ist einer der am häufigsten eingesetzten Interest-Point-Detektoren. Er wurde bereits mehrfach

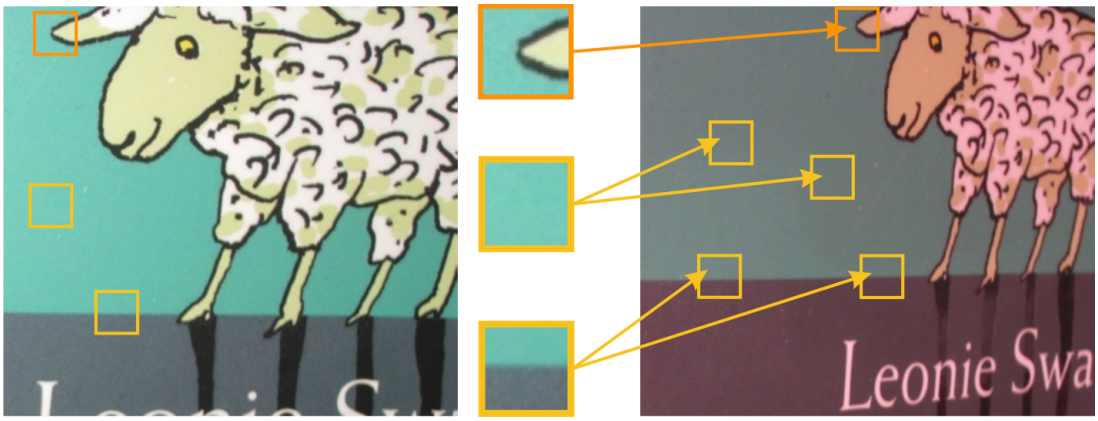


Abbildung 3.3: Vergleich der Eignung verschiedener Bildstrukturen zur Durchführung eines Bildabgleichs der beiden Bilder links und rechts. Bildstrukturen vergrößert in der Mitte dargestellt: oben: Eckpunkt; unten: Kante; Mitte: homogene Region [72]

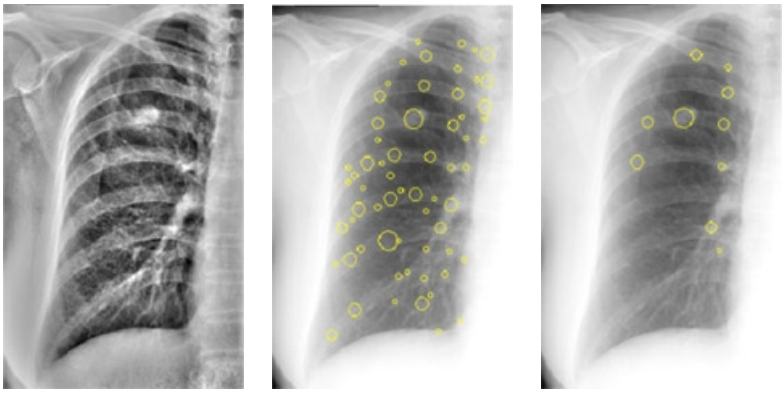
weiterentwickelt, um ihn robust gegenüber Änderungen der Skalierungen und affinen Transformationen zu machen (Abschnitt 3.1.3).

Der SUSAN- und der FAST-Detektor basieren nicht auf der Ermittlung von Intensitätsänderungen durch Bildung von Differenzen, sondern auf morphologischen Operatoren (Abschnitt 2.2.1). Der *SUSAN-Detektor* wurde 1997 von Smith und Brady vorgestellt [68] (siehe Abschnitt 4.1.4). Der *FAST-Detektor* ist ein speziell im Hinblick auf Schnelligkeit entwickelter Nachfolger des SUSAN-Detektors und wurde 2005 von Rosten vorgestellt [59, 60] (Abschnitt 4.1.5).

3.1.2 Blob-basierte Verfahren

Als Blobs bezeichnet man zusammenhängende Regionen im Bild die sich aus Bildelementen zusammensetzen, deren Intensitätswerte entweder heller oder dunkler sind als die Intensitätswerte der Bildelemente in ihrer direkten Umgebung [75, 24]. Sie entsprechen also nahezu *homogenen Regionen*, an deren *Grenze* eine *Intensitätsänderung* auftritt. Blob-Detektoren identifizieren solche Strukturen, beziehungsweise die Schwerpunkte solcher Strukturen in Bildern. Abbildung 3.4 zeigt ein Beispiel solcher detektierten Blobs [61]. Da die Position von Blobs nicht so exakt lokalisierbar ist wie die von Eckpunkten, werden Blobs eher für Anwendungen eingesetzt, bei denen die Kenntnis der genauen Position des Features nicht so relevant ist, wie beispielsweise zur Objekterkennung [75]. Eckpunkt- und Blob-Detektoren werden häufig in Kombination eingesetzt, da so unterschiedliche lokale Gegebenheiten von Bildern repräsentiert werden können.

Den Ausgangspunkt für viele Blob-Detektoren bildet eine Matrix von partiellen Ableitungen zweiter Ordnung, die als Hesse-Matrix bezeichnet wird (Abschnitt 4.1.3). Sowohl die Determinante als auch die Spur der Hesse-Matrix nehmen für blob-ähnliche Strukturen ein Maximum an. Einer der ersten Blob-Detektoren war der 1978 von Beaudet



(a) (b) (c)

Abbildung 3.4: Blob-Detektion zur Erkennung von Knoten in Röntgenbildern aus [61]: (a) Originalbild mit normalisiertem Kontrast; (b) detektierte Blobs; (c) Blobs die am wahrscheinlichsten den gesuchten Knoten entsprechen

vorgestellte *Determinant Of Hessian-Detektor* [3] (Abschnitt 4.1.3). Blobs werden dabei mit Hilfe einer Interest-Funktion identifiziert, die auf der Determinante der Hesse-Matrix basiert.

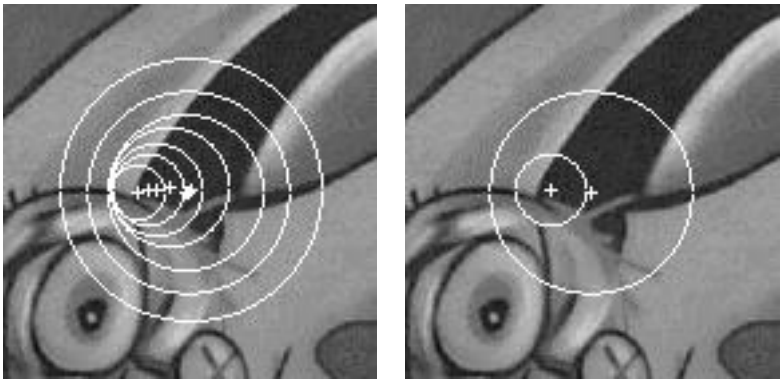
Die Größe von blob-ähnlichen Strukturen kann, im Gegensatz zur Größe von Eckpunkten, sehr gut bestimmt werden und gibt gleichzeitig einen guten Hinweis auf die Skalierung in der die vorhandenen lokalen Features vorliegen [75] (Abschnitt 4.2.1). Aus diesem Grund sind die von Blob-Detektoren identifizierten lokalen Features im Allgemeinen zumindest robust gegenüber Skalierungsveränderungen und werden daher im nächsten Abschnitt vorgestellt.

3.1.3 Skalierungsinvariante und affin-invariante Verfahren

Die in den vorigen Abschnitten angesprochenen Detektoren identifizieren Interest-Points nur auf einer einzelnen Auflösungsstufe von Bildern. Aus diesem Grund reagieren diese Verfahren äußerst empfindlich auf Skalierungsveränderungen. Die detektierten lokalen Features bilden somit keine stabile Grundlage für den Bildabgleich, in Anwendungen in denen sich die Zielobjekte in den jeweiligen Bildern durch ihre Größe unterscheiden [29].

Durch *Kombination ableitungsbasierter Methoden* zur Detektion von Interest-Points mit den in Abschnitt 2.3 vorgestellten *Skalenräumen*, können lokale Features auf allen Ebenen eines Skalenraums und somit für mehrere Auflösungsstufen eines Bildes identifiziert werden. Auf diese Weise kann Invarianz oder zumindest Robustheit der Verfahren gegenüber Skalierungsveränderungen erreicht werden. Grundsätzlich wird hier nach [43] zwischen zwei Ansätzen unterschieden, den sogenannten Multiskalen-Detektoren und den skalierungsinvarianten Detektoren.

Multiskalen-Ansätze [43] sind eine einfache Möglichkeit um Interest-Point-Detektoren robust gegenüber Skalierungsveränderungen zu machen. Dabei wird für das betrachtete Eingangsbild eine dreidimensionale Skalenraum-Repräsentation generiert (siehe Abbil-



(a) (b)

Abbildung 3.5: Vergleich der Ergebnisse des Multiskalen-Harris-Detektors (a) und des skalierungsinvarianten Harris Laplace-Detektors (b); Die Größe der Kreise weist auf die Skalenebene hin, auf der die lokalen Features detektiert wurden. Bilder aus [43]

dung 2.10) [77, 32, 31, 43]. Auf jeder Ebene des Skalenraums wird unabhängig voneinander ein Interest-Point-Detektor angewendet, um lokale Features zu identifizieren [40]. Für die Anwendung im Skalenraum werden die jeweils eingesetzten Interest-Funktionen in Bezug auf die Skalierung normalisiert. Auf diese Normalisierung wird in Abschnitt 4.2.1 eingegangen. Ein Beispiel für einen Multiskalen-Detektor ist der sogenannte *Multiskalen-Harris-Detektor*, der im Jahr 2000 von Dufournaud und Schmid veröffentlicht wurde (siehe [14]). Bei dieser Erweiterung des Harris-Detektors werden Features als lokale Maxima einer im Skalenraum berechneten, normalisierten Harris-Interest-Funktion bestimmt [75]. Da bei Multiskalen-Ansätzen die Detektion auf allen Skalenebenen unabhängig voneinander erfolgt entsteht hier häufig das Problem, dass zu viele lokale Features auf unterschiedlichen Ebenen identifiziert werden, die alle aus derselben Bildstruktur resultieren [43, 47]. Dadurch kann das Durchführen eines robusten Bildabgleichs erschwert werden [73] (siehe Abbildung 3.5a).

Bei *skalierungsinvarianten Ansätzen* [43] wird dieses Problem gelöst, indem nur lokale Features ausgewählt werden, die sowohl in Richtung der Ortskoordinaten x, y als auch in Richtung der Skalierung σ charakteristisch sind [40] (siehe Abbildung 3.5b). Dazu wird wiederum eine Skalenraum-Repräsentation $L(x, y, \sigma)$ des Eingangsbildes konstruiert in der dann nach Stellen gesucht wird, an denen die jeweilige skalennormalisierte Interest-Funktionen ein Extremum annimmt (Abschnitt 2.3) [43]. Die identifizierten Extrema entsprechen Interest-Points an der Position x, y gemeinsam mit ihrer sogenannten charakteristischen Skalierung σ (siehe Abschnitt 4.2.1). Die Grundlage zur Bestimmung der charakteristischen Skalierung von lokalen Features bildet die von Lindeberg [37] vorgestellte automatische Skalenselektion (4.2.1). Bei skalierungsinvarianten Ansätzen kann somit das Problem einer mehrfachen Detektion derselben Struktur auf mehreren Skalenebenen verringert und ein robuster Bildabgleich ermöglicht werden (siehe Abbildung 3.5b) [75]. In Abschnitt 4.2 werden einige skalierungsinvariante Erweiterungen ableitungsba-

sierter Interest-Point-Detektoren vorgestellt. Diese unterscheiden sich im Wesentlichen durch die skalennormalisierten Interest-Funktionen, die zur Identifizierung der Positionen x, y und der charakteristischen Skalierung σ der gesuchten lokalen Features eingesetzt werden.

Beispiele für ableitungsbasierte, skalierungsinvariante Methoden sind der von Mikolayzyk und Schmid 2001 vorgestellte *Harris Laplace-Detektor*, der zur Detektion von Eckpunkten in Skalenräumen eingesetzt wird und der *Hesse Laplace-Detektor*, der zur Identifizierung blob-ähnlicher Strukturen verwendet wird (Abschnitt 4.2.1.1) [44, 47]. Es handelt sich dabei um skalierungsinvariante Erweiterungen des Harris-Detektors beziehungsweise des Determinant Of Hessian-Detektors. Weitere Verfahren, die zur skalierungsinvarianten Detektion von Blobs eingesetzt werden können, sind der von Lindeberg [33, 37] vorgestellte *Laplacian Of Gaussian-Detektor* und der *Difference Of Gaussian-Detektor* (Abschnitt 4.2.1.2). Der Difference Of Gaussian-Detektor [12, 38] wird zudem zur Detektion sogenannter Schlüsselpunkte in dem von Lowe für den Einsatz bei der Objekterkennung entwickelten SIFT-Detektor eingesetzt (siehe [38, 39]).

Multiskalen-Ansätze und skalierungsinvariante Ansätze liefern als Ergebnis zumeist kreisförmige Regionen, die sogenannten *Interest-Regions* (Abschnitt 4.2). Diese Interest-Regions stellen sowohl Informationen über die Position, als auch Informationen über die Skalierung der detektierten lokalen Features bereit [40, 75].

Damit dieselben Features in zwei Bildern derselben Szene identifiziert werden können, auch wenn diese mit unterschiedlichem Betrachtungswinkel aufgenommen wurden, müssen die eingesetzten Detektoren invariant gegenüber *affinen Transformationen* sein [29]. Affine Invarianz von Detektoren wird häufig erreicht, indem sowohl Informationen über Position und Größe der identifizierten lokalen Features als auch eine Schätzung der affinen Form des Features extrahiert wird [75, 24]. Das Ergebnis der Detektion ist, wie auch schon bei den skalierungsinvarianten Detektoren, eine Interest-Region. Diese beschreibt die detektierten lokalen Features und hat meistens die Form einer Ellipse. Da die Gruppe der affinen Transformationen uniforme Skalierungen umfasst, sind affin-invariante Detektoren auch skalierungsinvariant [75].

Beispiele für affin-invariante Verfahren sind der *Harris Affine-* und der *Hesse Affine-Detektor*. Die beiden ableitungsbasierten Detektoren sind Erweiterungen des Harris Laplace- beziehungsweise Hesse Laplace-Detektors und wurden ebenfalls von Mikolayczyk und Schmid entwickelt [45, 47]. Ausgehend von den durch den Harris Laplace- beziehungsweise Hesse Laplace-Detektor ermittelten lokalen Features mit ihrer charakteristischen Skalierung wird hier ein iterativer Algorithmus von Lindeberg [36] zur Schätzung der affinen Verzerrung der vorliegenden Bildstruktur angewendet [75]. Das Bild kann dann neu abgetastet werden, um diese Verzerrung auszugleichen.

Der sogenannte *Salient Regions-Detektor* von Kadir und Brady basiert auf Konzepten aus dem Gebiet der Informationstheorie [27]. Der 2001 vorgestellte Detektor identifiziert Blobs als Stellen im Bild, die einen besonders hohen Informationsgehalt aufweisen [24]. Der Informationsgehalt an der Stelle wird beispielsweise durch die Entropie der Verteilung der Intensitätswerte in der Nachbarschaft eines Bildelements gemessen. Der Detektor arbeitet auf mehreren Skalen und ist dadurch skalierungsinvariant [24]. In [28] wurde

eine erweiterte Version des Detektors vorgestellt, die zudem invariant gegenüber affinen Transformationen ist.

Zwei weitere affin-invariante Detektoren die häufig blob-ähnliche Strukturen auffinden, sind der MSER-Detektor und der IBR-Detektor. Da sie auch homogene Bildregionen mit unregelmäßiger Form identifizieren die weder Blobs noch Eckpunkten entsprechen, werden sie in [75] als Regionen-Detektoren bezeichnet. Der *MSER-Detektor* (Abschnitt 4.2.2) detektiert lokale Features auf Basis eines watershed-ähnlichen Segmentierungsverfahrens (siehe [69]). Beim *IBR-Detektor* (Intensity Based Regions-Detektor) werden zunächst Intensitätsextrema, d.h. lokale Minima oder Maxima, im Bild identifiziert. Ausgehend von diesen Extrema werden kreisförmig Strahlen ausgesandt und die Intensitätsprofile entlang dieser Strahlen untersucht, um diejenigen Stellen zu finden, an denen ein Extremum im Intensitätsprofil vorliegt. Auf diese Weise können die Grenzen homogener Regionen mit irregulärer Form identifiziert werden (vgl. [74]).

3.2 Methoden zur Evaluierung

Es existiert eine große Anzahl verschiedener Methoden zur Detektion von lokalen Features. Daher ist es wichtig, diese gezielt miteinander vergleichen zu können, beispielsweise um geeignete Verfahren für bestimmte Anwendungen zu finden [75]. In der Literatur finden sich einige Arbeiten, die die Evaluation verschiedener Detektoren anhand unterschiedlicher Kriterien zum Thema haben. Dabei beruhen frühe Evaluierungen verschiedener lokaler Feature-Detektoren zumeist auf Sichtkontrolle oder Verifizierung einer Ground Truth (siehe [64, 56]). Heute wird der Vergleich verschiedener Interest-Point-Detektoren zumeist auf Basis zweier von Schmid u.a. im Jahr 1998 vorgestellter Evaluationskriterien durchgeführt [73]. Die zwei Kriterien um die es sich handelt sind die sogenannte Wiederholungsrate (engl.: Repeatability Rate) und der Informationsgehalt [63, 64].

Bei Methoden, die auf *Sichtkontrolle* basieren, werden zunächst Kriterien festgelegt, anhand derer die Güte der jeweiligen Methoden beurteilt werden soll. Beispielsweise könnten die Detektoren anhand der Positionierung oder Verteilung der identifizierten Interest-Points bewertet werden [56]. Die von einem Detektor gelieferten Ergebnisse werden im einfachsten Fall visuell durch den Menschen beurteilt. Derartige Methoden hängen vorrangig von den evaluierenden Person ab und sind daher sehr subjektiv [64]. Für Methoden, die auf *Verifizierung einer Ground Truth* basieren, muss zunächst manuell diese Ground Truth (Grundwahrheit) festgelegt werden. Diese ist abhängig von der menschlichen Interpretation des Bildes und somit ebenfalls subjektiv [64]. Nach Ausführung des Detektors wird durch einen Vergleich der Ergebnisse mit der definierten Grundwahrheit die Qualität des eingesetzten Interest-Point-Detektors beurteilt.

Der *Informationsgehalt* ist ein Maß für die Unterscheidungskraft beziehungsweise Besonderheit eines Interest-Points und wird auf Basis der Entropie eines Bildausschnitts um den Interest-Point ermittelt (siehe [64]). Mit Hilfe der *Wiederholungsrate* kann ermittelt werden, wie robust ein Detektor gegenüber verschiedenen geometrischen und photometrischen Transformationen ist (Abschnitt 2.1.1). Im Gegensatz zu anderen Evaluationskriterien wird die Wiederholungsrate zwischen Bildpaaren berechnet [64]. Es wird der Anteil

derjenigen lokalen Features ermittelt, die in beiden Bildern des betrachteten Bildpaares detektiert wurden und die zu jeweils korrespondierenden Objekten gehören (Abschnitt 2.1). Ein wesentlicher Vorteil dieser beiden Evaluierungskriterien, der Wiederholungsrate und des Informationsgehaltes, liegt darin, dass sie automatisiert ausgewertet werden können [56].

Die Wiederholungsrate eines Detektors ist ein ausschlaggebendes Kriterium zur Beurteilung für viele Anwendungen im Bereich der Computer Vision, beispielsweise für die Objekterkennung. Vergleiche gängiger Interest-Point-Detektoren auf Grundlage der Wiederholungsrate finden sich beispielsweise in [63, 64, 65, 47, 49, 48, 17, 53]. Für die Evaluation von Verfahren zur Detektion lokaler Features entwickelten Mikolajczyk u.a. eine Sammlung von Testbildserien und stellten diese zur freien Verfügung[2]. Eine solche Serie von Testbildern setzt sich jeweils aus einem Referenzbild und schrittweise transformierten Versionen dieses Referenzbildes zusammen. Zusätzlich ist eine Homographie gegeben, die es ermöglicht, Bildpositionen in den transformierten Bildern zu Positionen im Referenzbild in Beziehung zu setzen. Diese Daten werden heute standardmäßig zum Testen neu entwickelter Detektoren eingesetzt [75]. Von den Autoren wird zudem MATLAB-Code zur Ermittlung der Wiederholungsrate bereitgestellt.

[2]Siehe http://www.robots.ox.ac.uk/~vgg/research/affine/ (zuletzt abgerufen am 11.11.2011)

4 Detektoren

In den nachfolgenden Abschnitten wird die Funktionsweise gängiger Verfahren zur Detektion von lokalen Features im Detail beschrieben. Dabei handelt es sich um den Harris-, Harris Laplace-, Determinant Of Hessian-, Hesse Laplace-, Laplacian Of Gaussian-, Difference Of Gaussian-, FAST- und den MSER-Detektor. Diese Detektoren wurden auch in die entwickelte Applikation zum Vergleich integriert. Zum besseren Verständnis wird zudem der Moravec-Detektor beschrieben, der ein direkter Vorgänger des Harris-Detektors ist und der SUSAN-Detektor, auf dem der FAST-Detektor aufbaut.

Bei den jeweiligen Detektoren wird zwischen Interest-Point-Detektoren (Abschnitt 4.1) und Interest-Region-Detektoren (Abschnitt 4.2) unterschieden (vgl. [73]). *Interest-Point-Detektoren* sind Verfahren, die Blobs oder Eckpunkte identifizieren und als Resultat Punkte (Interest-Points) liefern, welche die Position des lokalen Features beschreiben. Die von diesen Methoden detektierten lokalen Features sind nicht invariant gegenüber uniformen Skalierungen oder affinen Transformationen. Zu den *Interest-Region-Detektoren* werden Verfahren gezählt, die als Ergebnis sogenannte Interest-Regions liefern. Diese Interest-Regions bieten zusätzliche Informationen über die detektierten lokalen Features, beispielsweise über die Größe des betrachteten Features. Zu den Detektoren dieser Gruppe zählen skalierungsinvariante und auch affin-invariante Detektoren.

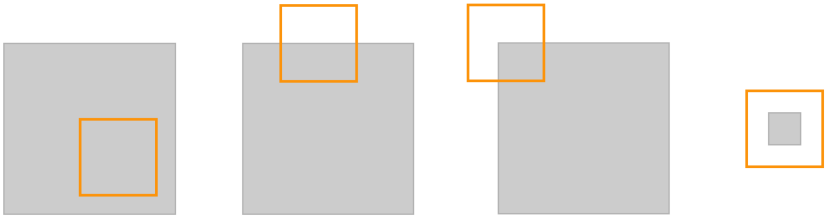
4.1 Interest-Point-Detektoren

Im Folgenden werden zunächst Interest-Point-Detektoren vorgestellt, die Intensitätsänderungen auf Basis von Differenzbildung beziehungsweise Berechnung von Ableitungen identifizieren. Die Detektoren sind der *Moravec-*, *Harris-*, und der *Determinant Of Hessian-Detektor*. Für Methoden, die Eckpunkte in Bildern detektieren, spielt dabei die Bildung von partiellen Ableitungen erster Ordnung und die daraus berechnete Autokorrelationsmatrix (siehe Abschnitt 4.1.2) eine wichtige Rolle. Ableitungsbasierte Blob-Detektoren bauen häufig auf Kombinationen partieller Ableitungen zweiter Ordnung aus der Hesse-Matrix als Interest-Funktion auf (siehe Abschnitt 4.1.3).

Des weiteren werden zwei Detektoren vorgestellt, die Änderungen von Intensitätswerten auf der Basis von nichtlinearen Operatoren ermitteln. Sowohl der SUSAN-Detektor als auch der FAST-Detektor können zur Detektion von Eckpunkten in Bildern eingesetzt werden.

4.1.1 Der Moravec-Detektor

Beim Moravec-Detektor (siehe [52]) wird ein kleines Fenster um ein aktuelles Bildelement betrachtet. Man geht davon aus, dass sich die Intensitätswerte in dem durch das Fenster



(a) (b) (c) (d)

Abbildung 4.1: Fallunterscheidung beim Moravec-Detektor nach Lage des betrachteten Fensters: (a) über homogener Region; (b) über Kante; (c) über Eckpunkt; (d) über isoliertem Punkt (siehe Fußnote 1)

festgelegten Bildausschnitt bei leichten Verschiebungen des Fensters verändern. In Abhängigkeit von der Art der Veränderung kann unterschieden werden ob eine homogene Region, eine Kante oder ein Eckpunkt im betrachteten Bildausschnitt vorliegt [73].

Um Eckpunkte in einem Eingangsbild I zu identifizieren wird wie folgt vorgegangen [40, 73]: Es wird zunächst ein Fenster H über dem aktuell betrachteten Bildelement $p = (x, y)$ platziert. Das lokale Fenster wird dann um einen kleinen Betrag in verschiedene Richtungen (horizontal, vertikal und in Richtung der Diagonalen) verschoben. Durch Berechnung des *mittleren Abstandsquadrats* (engl.: Sum of Squared Differences) werden die Intensitätsänderungen $SSD(u, v)$ in der Nachbarschaft des betrachteten Bildelements bestimmt, die durch die Verschiebung des Fensters $H(x, y)$ in die jeweilige Richtung $s = (u, v)$ entstehen [20, 40]. Das mittlere Abstandsquadrat ist folgendermaßen definiert [20, 40]:

$$SSD(u, v) = \sum_{x,y} H(x, y) \left[I(x + u, y + v) - I(x, y) \right]^2 \tag{4.1}$$

$$s(u, v) \in \{(\ 0,\ 1), (\ 0, -1), (\ 1,\ 0), (-1,\ 0), (\ 1,\ 1), (-1, -1), (\ 1, -1), (-1,\ 1)\}$$

Das Fenster $H(x, y)$ nimmt innerhalb der betrachteten quadratischen Region den Wert 1 an und 0 außerhalb. Es hat also die Form eines linearen Mittelwertfilters ohne vorgeschaltetem Skalierungsfaktor (siehe Gleichung 2.4). In Abhängigkeit von der Beschaffenheit des Bildausschnitts innerhalb des betrachteten Fensters ergeben sich drei verschiedene Fälle [20] (Abbildung 4.1)[1]:

1. Befindet sich das Fenster über einer farblich nahezu homogenen Fläche, so bewirken Verschiebungen des Fensters nur geringe Intensitätsänderungen (Abbildung 4.1a).

2. Befindet sich das Fenster über einer Kante, so entstehen bei Verschiebungen des Fensters entlang der Kante kaum Änderungen. Verschiebungen normal zur Kante resultieren hingegen in einer großen Veränderung der Intensitätswerte (Abbildung 4.1b).

[1] http://kiwi.cs.dal.ca/~dparks/CornerDetection/moravec.htm (zuletzt abgerufen am 13.11.2011)

3. Befindet sich das Fenster über einem Eckpunkt oder über einem isolierten Punkt, so resultieren alle betrachteten Verschiebungen in einer großen Intensitätsänderung (Abbildung 4.1c, Abbildung 4.1d). Eckpunkte können daher als Stellen identifiziert werden, an denen die minimale Intensitätsänderung die bei Verschiebungen des Fensters entsteht groß ist. Die Moravec Interest-Funktion IF_{Mor} für ein betrachtetes Bildelement p wird daher definiert als [20]:

$$IF_{Mor}(x, y) = min(SSD(u, v)) \tag{4.2}$$

Die Funktion wird für alle Bildelemente (x, y) des Eingangsbildes I berechnet. Im so entstandenen Ergebnisbild stellt der Intensitätswert an jeder Position ein Maß für die Wahrscheinlichkeit des Vorliegens eines Eckpunktes dar. Je höher der Intensitätswert, desto wahrscheinlicher ist es, dass ein Eckpunkt an der entsprechenden Position gefunden wurde. Es wird daher an denjenigen Stellen ein Eckpunkt identifiziert, an denen die Funktion IF_{Mor} ein lokales Maximum oberhalb eines definierten Schwellwerts T bildet, d.h. wenn:

$$min(SSD(u, v)) > T$$

Durch den Vergleich mit dem Schwellwert soll zudem gewährleistet werden, dass "falsche" Eckpunkte, wie beispielsweise durch Bildrauschen entstandene Störungen, nicht zu einfach in das Endresultat aufgenommen werden [20].

Der Moravec-Detektor war einer der ersten entwickelten Interest-Point-Detektoren und weist nach [20] einige Probleme auf. So ist beispielsweise das Ergebnis der Interest-Funktion IF_{Mor} richtungsabhängig, weil nur diskrete Verschiebungen des Fensters berücksichtigt werden. Die Wiederholbarkeit der Detektion bei Rotation eines Bildes ist somit nicht gegeben. Zudem ist das Ergebnis aufgrund des verwendeten binären rechteckigen Fensters H oft verrauscht und es werden immer noch zu leicht "falsche" Eckpunkte in die Ergebnismenge aufgenommen. Alle diese Probleme werden bei dem im nachfolgenden Abschnitt vorgestellten Harris-Detektor gelöst.

4.1.2 Der Harris-Detektor

Beim Harris-Detektor [20] wird im Wesentlichen die selbe Idee wie beim Moravec-Detektor verfolgt. Die Grundlage zur Berechnung der Interest-Funktion bildet hier allerdings die Autokorrelationsmatrix und nicht das mittlere Abstandsquadrat. Die *Autokorrelationsmatrix* (auch als Strukturtensor bezeichnet) ist eine aus partiellen Ableitungen erster Ordnung aufgebaute Matrix. Eine ausführliche Beschreibung der Eigenschaften und Anwendungen dieser Matrix ist in [25] zu finden. Die Autokorrelationsmatrix liefert eine Beschreibung der Gradientenverteilung [75] in der Umgebung eines betrachteten Bildelements $p = (x, y)$ und ist definiert als [14, 10]:

$$\mu(x, y, \sigma) = H^{G(\sigma)} \otimes \begin{pmatrix} I_x^2(x, y) & I_x I_y(x, y) \\ I_x I_y(x, y) & I_y^2(x, y) \end{pmatrix} = \begin{pmatrix} A & C \\ C & B \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \tag{4.3}$$

Zum Aufbau der Autokorrelationsmatrix wird wie folgt vorgegangen [75, 29]: Es werden zunächst die partiellen Ableitungen erster Ordnung $I_x(x, y)$ und $I_y(x, y)$ in horizontale

und vertikale Richtung bestimmt. Diese können durch lineare Faltung mit beliebigen Gradientenfiltern H^{∇_x} und H^{∇_y} approximiert werden (Abschnitt 2.2.2.2):

$$I_x(x,y) = \frac{\partial I(x,y)}{\partial x} \approx I(x,y) \otimes H^{\nabla_x}$$

$$I_y(x,y) = \frac{\partial I(x,y)}{\partial y} \approx I(x,y) \otimes H^{\nabla_y}$$

In der ursprünglichen Version des Harris-Detektors wird der Sobel-Operator (siehe Gleichung 2.16) zur Approximation der partiellen Ableitungen eingesetzt [69]. Neuere Varianten des Harris-Detektors verwenden dafür gewöhnlich Gaußsche Ableitungsfilter (Abschnitt 2.2.3), die aufgrund der Eigenschaften der Gauß-Funktion besser geeignet sind [72]. Die einzelnen Komponenten I_x^2, I_y^2 und I_xI_y der Autokorrelationsmatrix werden berechnet und die Ergebnisse werden mit einem Gauß-Filter $H^{G(\sigma)}$ über die Nachbarschaft des betrachteten Bildelements p gemittelt. Die Standardabweichung σ des verwendeten Gauß-Filters legt dabei die Größe der berücksichtigten Nachbarschaft fest [73]. Durch die Glättung wird der Verstärkung von Bildrauschen durch die eingesetzten Gradientenfilter entgegengewirkt.

Die Matrix μ kann in eine Diagonalmatrix übergeführt werden (siehe [10]). Auf diese Weise können die Eigenwerte λ_1, λ_2 der Autokorrelationsmatrix μ bestimmt werden, die eine richtungsunabhängige Beschreibung von μ bilden [20, 10]. Die Relation der Eigenwerte zueinander lässt darauf schließen, ob und in wie viele Richtungen Intensitätsänderungen in der Nachbarschaft des betrachteten Bildelements p auftreten und folglich ob eine homogene Region ($\lambda_1 \approx \lambda_2$), eine Kante ($|\lambda_1 - \lambda_2| \gg 0$) oder ein Eckpunkt ($\lambda_1 \gg 0 \wedge \lambda_2 \gg 0$) vorliegt (Abbildung 4.1a - 4.1c)[29].

Aufbauend auf diesen Beobachtungen definieren Harris und Stephens die Interest-Funktion IF_{Harr}, die beim Harris-Detektor zur Identifizierung von Eckpunkten in Bildern eingesetzt wird. Die Funktion IF_{Harr} wird durch Kombination von Spur[2] (Gleichung 4.5) und Determinante (Gleichung 4.6) der Autokorrelationsmatrix μ berechnet als [29]:

$$IF_{Harr}(x,y) = Det(\mu) - \alpha Tr^2(\mu) \tag{4.4}$$

$$Tr(\mu) = \lambda_1 + \lambda_2 = A + B \tag{4.5}$$

$$Det(\mu) = \lambda_1\lambda_2 = AB - C^2 \tag{4.6}$$

Die Determinante der Matrix μ entspricht dem Produkt ihrer Eigenwerte (siehe Gleichung 4.5) und die Spur der Matrix der Summe ihrer Eigenwerte (siehe Gleichung 4.6). Aus diesem Grund können bei Verwendung der Harris-Funktion IF_{Harr} die durch die Eigenwerte enthaltenen Informationen über die lokale Bildstruktur genutzt werden, ohne

[2]Die Spur (engl.: Trace) einer quadratischen Matrix entspricht der Summe der Koeffizienten in der Hauptdiagonale [7, 10].

die Eigenwerte tatsächlich berechnen zu müssen [29]. Die Konstante α steuert die Empfindlichkeit des Harris-Detektors [10]. Durch Veränderung von α kann das Verhalten des Detektors so verändert werden, dass er sowohl zur Identifizierung von Eckpunkten als auch von Kanten eingesetzt werden kann [40]. Je größer der Wert für α gewählt wird, desto unempfindlicher wird der Detektor und es werden weniger Eckpunkte gefunden. Typischerweise werden Werte zwischen $0.04 - 0.06$ für α gewählt [10].

Wie schon beim Moravec-Detektor, kann durch Auswertung der Funktion $IF_{Harr}(x, y)$ die Beschaffenheit der lokalen Bildstruktur um das betrachtete Bildelement p ermittelt werden [20]:

1. Ist $IF_{Harr} \approx 0$, so liegt eine uniforme Bildregion ohne signifikante Intensitätsänderungen vor.

2. Wenn $IF_{Harr} < 0$ ist, dann tritt nur in eine Richtung eine signifikante Intensitätsänderung auf, das heißt es liegt eine Kante vor.

3. Ist der Wert beider Eigenwerte groß und somit $IF_{Harr} > 0$, so liegt ein Eckpunkt vor. Es tritt eine signifikante Intensitätsänderung in mehr als eine Richtung auf.

Schlussendlich wird an Stellen ein Eckpunkt identifiziert, an denen die Interest-Funktion IF_{Harr} ein lokales Maximum annimmt und oberhalb eines festgelegten Schwellwerts T liegt, das heißt falls $IF_{Harr}(x, y) > T$.

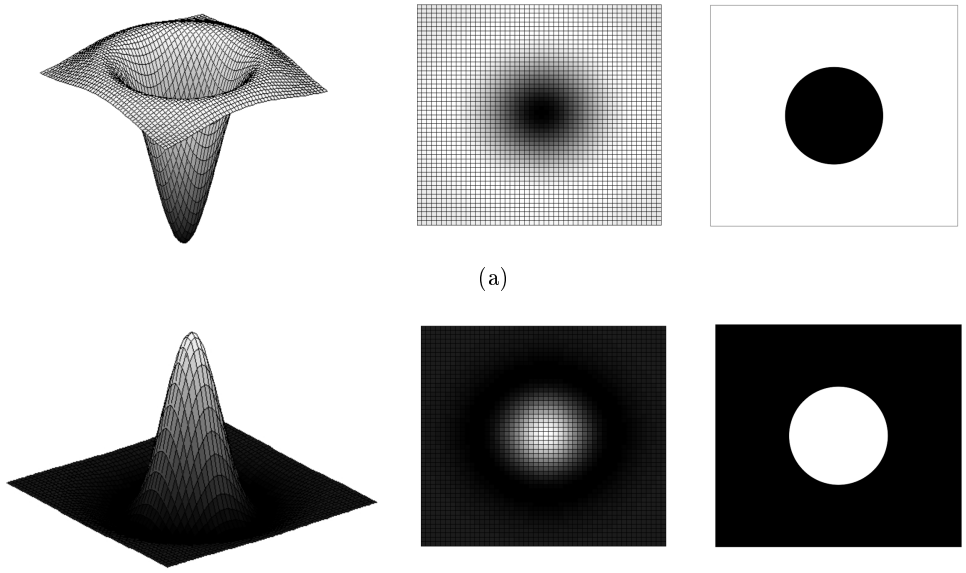
Das Verfahren nach Harris und Stephens löst die in Abschnitt 4.1.1 vorgestellten Probleme des Moravec-Detektors (vgl. [20]). Durch die Verwendung der Autokorrelationsmatrix anstelle einer Menge von diskreten Verschiebungen eines Fensters wird die richtungsunabhängige Identifizierung von Eckpunkten ermöglicht. Die Wiederholbarkeit der mit Hilfe des Harris-Detektors detektierten lokalen Features ist somit auch bei Rotationen von Bildern gegeben [75]. Durch den Einsatz des Gauß-Filters anstelle eines Mittelwertfilters ist der Harris-Detektor zudem weniger anfällig für Bildstörungen. Der Nachteil des Harris-Detektors liegt darin, dass er in dieser Form nicht invariant gegenüber Veränderungen der Skalierung ist. Eine skalierungsinvariante Erweiterung des Harris-Detektors wird in Abschnitt 4.2.1.1 vorgestellt.

Aus der Autokorrelationsmatrix lassen sich auch andere Interest-Funktionen ermitteln, die für die Identifizierung von Eckpunkten in Bildern geeignet sind (siehe [75]). Solche Interest-Funktionen wurden beispielsweise von Shi & Thomasi [67] und Noble [55] vorgestellt.

4.1.3 Der Determinant Of Hessian-Detektor

Vor allem zur Detektion von blob-ähnlichen Strukturen in Bildern werden häufig Interest-Funktionen eingesetzt die aus der Hesse-Matrix berechnet werden. Für ein Bildelement an der Stelle $p = (x, y)$ im Bild $I(x, y)$ ist die Hesse-Matrix definiert als [73]:

$$\eta(x, y, \sigma) = \begin{pmatrix} I_{xx}(x, y, \sigma) & I_{xy}(x, y, \sigma) \\ I_{xy}(x, y, \sigma) & I_{yy}(x, y, \sigma) \end{pmatrix} = \begin{pmatrix} A & C \\ C & B \end{pmatrix} \tag{4.7}$$



(a)

(b)

Abbildung 4.2: Vergleich der Struktur der Determinante der Hesse-Matrix in Abbildung (a) und der Spur der Hesse-Matrix in Abbildung (b); Beide Funktionen können zur Detektion blob-ähnlicher Strukturen eingesetzt werden (siehe Fußnote 3)

Die Komponenten $I_i(x, y)$ der Matrix η entsprechen dabei den partiellen Ableitungen zweiter Ordnung des Eingangsbildes I in Richtung i und werden durch Faltungen mit entsprechenden Gaußschen Ableitungsfiltern der Größe σ approximiert (siehe Abschnitt 2.2.3) [73], beispielsweise:

$$I_{xx}(x, y, \sigma) = \frac{\partial^2 I(x, y)}{\partial^2 x} \approx I(x, y) \otimes H^{G_{xx}(\sigma)}$$

$$I_{xy}(x, y, \sigma) = \frac{\partial^2 I(x, y)}{\partial x \partial y} \approx I(x, y) \otimes H^{G_{xy}(\sigma)}$$

Die Hesse-Matrix ist, ähnlich der Autokorrelationsmatrix, eine symmetrische, quadratische Matrix aus der sich Informationen über die lokale Struktur in der Nachbarschaft eines Bildelements p ableiten lassen. Dazu werden, wie schon beim Harris-Detektor, die Eigenwerte λ_1, λ_2 der Matrix herangezogen [75, 73].

Beim sogenannten Determinant Of Hessian-Detektor (*DoH-Detektor*) nach Beaudet (siehe [3]) werden blob-ähnliche Strukturen an Stellen im Eingangsbild I identifiziert, an denen das Produkt der Eigenwerte der berechneten Hesse-Matrix groß ist. Das Produkt der Eigenwerte λ_1, λ_2 entspricht der Determinante der Hesse-Matrix η. Die beim DoH-Detektor zur Detektion von lokalen Features eingesetzte Interest-Funktion IF_{DoH} ist somit definiert als [19]:

$$
\begin{aligned}
IF_{DoH}(x,y,\sigma) &= Det(\eta) = \lambda_1\lambda_2 = AB - C^2 \\
&= I_{xx}(x,y,\sigma)I_{yy}(x,y,\sigma) - (I_{xy}(x,y,\sigma))^2
\end{aligned} \tag{4.8}
$$

Blobs werden dann identifiziert, wenn die Interest-Funktion ein lokales Maximum annimmt und über einem festgelegten Schwellwert T liegt, d.h. $IF_{DoH}(x,y,\sigma) > T$.

Die Spur der Hesse-Matrix η, berechnet mit Hilfe von Gaußschen Ableitungsfiltern, ist definiert als [73]:

$$
\begin{aligned}
Tr(\eta) &= \lambda_1 + \lambda_2 = A + B \\
&= I_{xx}(x,y,\sigma) + I_{yy}(x,y,\sigma)
\end{aligned} \tag{4.9}
$$

Die Funktion $Tr(\eta)$ entspricht dem in Abschnitt 2.2.3 vorgestellten Laplacian Of Gaussian-Operator [75].

Sowohl die Determinante als auch die Spur der Hesse-Matrix nehmen ein Extremum für blob-ähnliche Strukturen in Bildern an [75]. Dies liegt in der Form der Funktionen begründet [73]. Abbildung 4.2a links zeigt die 3D-Form der LoG-Funktion. Zur Berechnung der Funktion $Tr(\eta)$ wird das Eingangsbild I mit einem Filterkern $H^{LoG(\sigma)}$ gefaltet, dessen Koeffizienten durch die LoG-Funktion bestimmt werden (Abschnitt 2.2.3). Die lineare Faltung kann auch als Bildvergleich verstanden werden, bei dem jede Position (x,y) des Eingangsbild mit einem Filterkern verglichen wird. Den Filterkern kann man sich dabei als eine Art Schablone vorstellen. Das Resultat der Faltungsoperation ist dann an denjenigen Stellen am stärksten, an denen die gewählte Schablone der aktuell betrachteten Filterregion am ähnlichsten ist [66]. Im Fall des LoG-Operators wird ein Filterkern mit ähnlichem Aussehen wie Abbildung 4.2a Mitte als Schablone eingesetzt[3]. Bei Faltung eines Bildes mit einem solchen Filterkern ergibt sich ein starker Ausschlag an den Stellen, an denen blob-ähnliche Strukturen vorliegen. Bei einem Vergleich des Blobs in Abbildung 4.2a rechts, würde sich bei Faltung mit einem LoG-Filter der stärkste Ausschlag ergeben, wenn der Filter über dem Blob platziert wird und die Größe der Blob-Struktur mit der Ausdehnung des Filters korreliert. Aus diesem Grund gibt die Größe der zur Detektion von Blobs eingesetzten gauß-basierten Filter[4] auch gleichzeitig einen Hinweis auf die Ausdehnung der detektierten lokalen Struktur. Die Form der Funktion $Det(\eta)$ sieht der eines auf den Kopf gestellten LoG-Operators sehr ähnlich. Daher ist auch das Verhalten von $Det(\eta)$ im Bezug auf blob-ähnliche Strukturen ähnlich dem des LoG-Operators (siehe 4.2b).

Sowohl die Determinante als auch die Spur der Hesse-Matrix können somit zur Blob-Detektion eingesetzt werden. Der LoG-Operator wird zudem häufig zur automatischen Skalenselektion eingesetzt (Abschnitt 4.2.1). Der Nachteil von Blob-Detektoren auf Basis der Hesse-Matrix liegt darin, dass sie häufig Interest-Points in der nähe von Kanten

[3] http://www.cs.unc.edu/~lazebnik/spring11/lec08_blob.pdf (zuletzt abgerufen am 12.11.2011)

[4] Diese wird im Fall des LoG-Operators und auch bei anderen gauß-basierten Filterkernen durch die Standardabweichung σ bestimmt.

identifizieren. Diese sind sehr anfällig für Bildrauschen und daher nicht sehr robust [75]. Es existieren zahlreiche skalierungsinvariante Detektoren, die auf der Hesse-Matrix und daraus berechneten Interest-Funktionen basieren. Einige davon werden in Abschnitt 4.2 vorgestellt.

4.1.4 Der SUSAN-Detektor

Das Akronym SUSAN steht für *Smallest Univalue Segment Assimimilating* und beschreibt ein Verfahren, das sowohl zur Detektion von Eckpunkten als auch zur Detektion von Kanten und zur Unterdrückung von Bildrauschen eingesetzt werden kann [75, 68]. Im Folgenden wird darauf eingegangen, wie mit Hilfe des SUSAN-Detektors Eckpunkte identifiziert werden können. Für nähere Informationen bezüglich der anderen Funktionalitäten siehe [68].

Zum Auffinden von eckpunkt-ähnlichen Strukturen in Bildern wird wie folgt vorgegangen [75, 68]: Ein kreisförmiges Fenster M mit fixem Radius r wird mit seinem Zentrum über dem aktuell betrachteten Bildelement $p = (x, y)$ positioniert. Das Bildelement p wird als Nukleus bezeichnet. Anschließend wird der Intensitätswert $I(p)$ des Nukleus mit den Intensitätswerten $I(m)$ aller Bildelemente m innerhalb der Nachbarschaft M verglichen [68]:

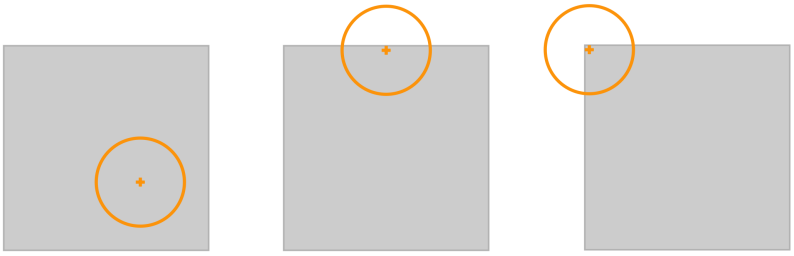
$$C(m,p) = \begin{cases} 1 & wenn \ \ |I(m) - I(p)| \leq T_v \\ 0 & wenn \ \ |I(m) - I(p)| > T_v \end{cases}$$

Auf diese Weise können diejenigen Bildelemente in der Umgebung des Nukleus bestimmt werden die gleiche, beziehungsweise ähnliche Intensitätswerte wie der Nukleus selbst aufweisen. Wie stark sich die Intensitätswerte der betrachteten Bildelemente unterscheiden dürfen, um als ähnlich klassifiziert zu gelten, wird durch den Schwellwert T_v gesteuert [68].

Alle Bildelemente mit ähnlichem Intensitätswert innerhalb des Fensters M bilden eine Fläche. Diese Fläche wird als USAN bezeichnet. Die Größe des zugehörigen USANs für einen Nukleus p wird berechnet als [68]:

$$IF_{SUS}(p) = \sum_{m \epsilon M} C(m,p)$$

Das Verhältnis der Größe des USANs zur Größe des Fensters M gibt Aufschluss über die lokalen Strukturen in der betrachteten Umgebung. Abbildung 4.3 zeigt drei kreisförmige Fenster in orange, die jeweils über unterschiedlichen Bildstrukturen liegen. In der Mitte des Fensters liegt der gekennzeichnete Nukleus. Das zugehörige USAN ist durch die graue Fläche innerhalb des Fensters gegeben. Liegt das betrachtete Fenster über einer eher homogenen Bildregion (Abbildung 4.3a), so erstreckt sich die USAN-Fläche nahezu über das gesamte Fenster. Wenn das betrachtete Fenster in der Nähe einer Kante liegt (Abbildung 4.3b), nimmt das USAN ungefähr die Hälfte der Fläche innerhalb des Fensters ein. In der Nähe von Eckpunkten sinkt die Größe der USAN-Fläche auf ein Viertel der Fenstergröße [75, 68].



(a) (b) (c)

Abbildung 4.3: Fallunterscheidung beim SUSAN-Detektor: (a) Fenster liegt über homogener Bildregion; (b) über einer Kante; (c) über einem Eckpunkt. Die grau durchscheinenden Stellen entsprechen Bildelementen, die einen ähnlichen Intensitätswert wie der Nukleus (+) in der Mitte des Fensters aufweisen [68]

Eckpunkte lassen sich mit Hilfe des SUSAN-Detektors folglich als Orte in Bildern bestimmen, an denen die Anzahl der Bildelemente mit ähnlichem Intensitätswert wie der des betrachteten Bildpunktes p ein lokales Minimum einnimmt und unter einem bestimmten Schwellwert T_g liegt. Für die Ergebnismenge werden die Bildelemente ausgewählt, für die sich die kleinsten USANs (smallest USAN = SUSAN) ergeben durch $IF_{SUS}(p) < T_g$ [75, 68].

4.1.5 Der FAST-Detektor

Ursprünglich für die Detektion von Eckpunkten in Echtzeitanwendungen geschaffen, ist der FAST-Detektor (*Features from Accelerated Segment Test*) vor allem auf Schnelligkeit ausgerichtet [59]. Der FAST-Detektor basiert auf einer ähnlichen Idee wie sein Vorgänger, der SUSAN-Detektor.

Zur Identifikation von Eckpunkten wird wie folgt vorgegangen [59, 60, 75]: Zunächst wird ein Bresenham-Kreis mit fixem Radius r über dem aktuell betrachteten Bildelement $p = (x, y)$ positioniert (siehe [21]). Der Intensitätswert $I(p)$ des Bildelements p wird mit den Intensitätswerten $I(m)$ jener Bildelemente m verglichen, die auf dem definierten Bresenham-Kreis liegen. Durch den Vergleich können diejenigen Bildelemente m identifiziert werden, die wesentlich dunkler als p sind, d.h. $I(m) \leq I(p) - T$, oder wesentlich heller als p sind, d.h. $I(m) > I(p) + T$ [60]. Durch den vordefinierten Schwellwert T wird wiederum gesteuert, wie groß die Differenz der betrachteten Intensitätswerte sein muss, um als unterschiedlich eingestuft zu werden. Ein Eckpunkt wird dann an Stellen identifiziert, an denen eine Folge von n benachbarten, auf dem Bresenham-Kreis liegenden Bildelementen m, einen wesentlich helleren oder dunkleren Intensitätswert aufweist als p. Der Intensitätswert des Bildelements p muss sich also in mehrere Richtungen ausreichend von seiner Umgebung unterscheiden, was der in Abschnitt 3.1.1 eingeführten Definition von Eckpunkten entspricht.

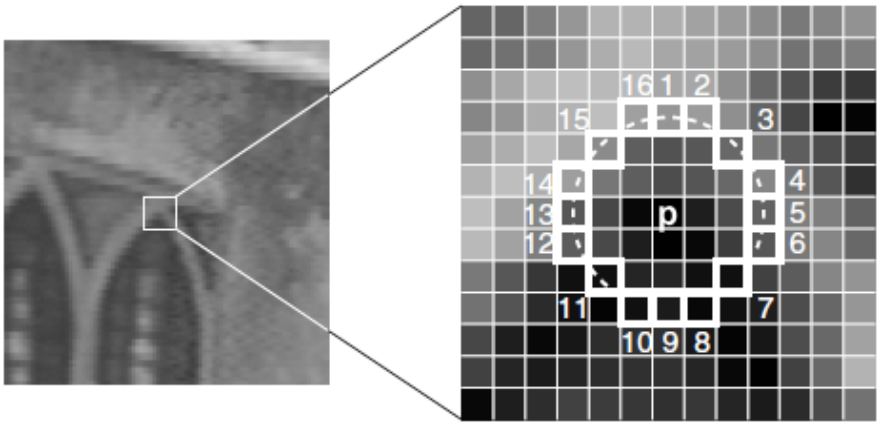
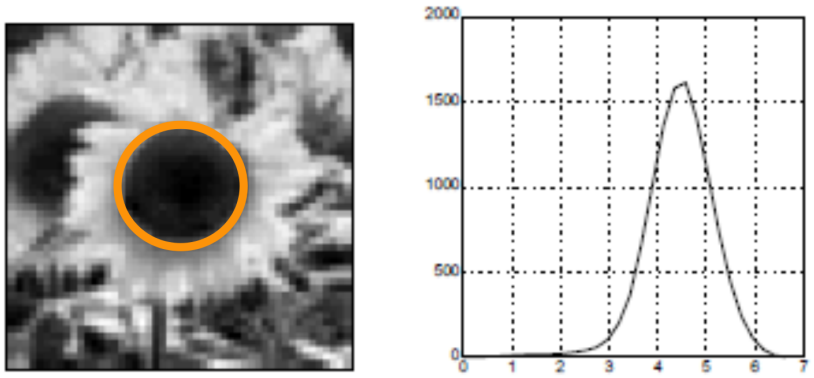


Abbildung 4.4: Darstellung der untersuchten Bildpunkte beim FAST-Detektor für eine aktuelle Bildposition p und einen Bresenham-Kreis mit Radius 3. Die Intensitätswerte von mindestens 12 benachbarten Bildelementen (gestrichelte Linie) müssen sich ausreichend von $I(p)$ unterscheiden, damit ein Eckpunkt detektiert wird [59]

In seiner ursprünglichen Version [59] wurde der FAST-Detektor mit $r = 3$ und $n = 12$ implementiert (siehe Abbildung 4.4). In dieser Konfiguration liegen insgesamt 16 zu untersuchende Bildpunkte auf dem Kreis um das aktuell betrachtete Bildelement p [75]. Bei Vorliegen eines Eckpunktes müssen sich nun mindestens 12 benachbarte, auf dem Kreis liegende, Bildelemente stark von p unterscheiden. Die Effizienz des FAST-Detektors kann in dieser Konfiguration nach [60] weiter gesteigert werden, indem zuerst nur die Bildpunkte $1, 9, 5$ und 13 untersucht werden. Ist an der betrachteten Stelle ein Eckpunkt vorhanden, so müssten zumindest drei dieser Bildelemente einen wesentlich helleren oder dunkleren Intensitätswert als p aufweisen. Ist dies nicht der Fall, so kann die betrachtete Stelle sofort ausgeschlossen werden. Erweiterungen des Detektors, bei denen unter anderem die Parameter r und n variiert werden können, wurden in [60] vorgeschlagen.

4.2 Interest-Region-Detektoren

Nachfolgend wird zunächst auf die Vorgehensweise zum Auffinden von lokalen Features mit Hilfe des Harris Laplace-, Hesse Laplace-, Laplacian Of Gaussian- und des Difference Of Gaussian-Detektors eingegangen. Bei diesen Detektoren handelt es sich um skalierungsinvariante Erweiterungen der im vorigen Abschnitt vorgestellten ableitungsbasierten Interest-Point-Detektoren. Die Skalierungsinvarianz wird durch Anwendung von skalennormalisierten Interest-Funktionen in einem Skalenraum in Kombination mit der automatischen Skalenselektion nach Lindeberg erreicht. Bei der Detektion der lokalen Features wird bei den jeweiligen Verfahren ähnlich vorgegangen, der Unterschied liegt vorrangig in den eingesetzten Interest-Funktionen. Abschließend wird der sogenannte MSER-Detektor vorgestellt. Dieser Detektor unterscheidet sich grundlegend von den an-



(a) (b)

Abbildung 4.5: Automatische Skalenselektion nach Lindeberg: Für die blob-ähnliche Struktur in (a) soll die charakteristische Skalierung σ gefunden werden. In Abbildung (b) ist die "Skalenspur" für diese Struktur abgebildet [43]. Dabei wird ein ableitungsbasierter Blob-Detektor auf verschiedenen Skalenebenen angewendet. Die Stärke der Antwort des eingesetzten Detektors (y-Achse) wird gegen die Skalierung σ (x-Achse) aufgetragen. Für die Skalenebene, an der die verwendete Interest-Funktion und die blob-ähnliche Struktur die größte Ähnlichkeit haben, wird ein Maximum über die Skalierungen angenommen. Dieses Maximum entspricht der charakteristischen Skalierung und ist im Ursprungsbild orange markiert. Bilder aus [37]

deren Verfahren, da er nicht auf Bildung von Ableitungen basiert sondern auf einer Art Segmentierung des Bildes. Alle in diesem Abschnitt vorgestellten Detektoren liefern Interest-Regions als Ergebnis.

4.2.1 Skalierungsinvariante Erweiterungen ableitungsbasierter Verfahren

Eine Vielzahl von intensitätsbasierten Verfahren zur Detektion von lokalen Features basiert auf der Verwendung von ableitungsbasierten Interest-Funktionen, d.h. auf Kombinationen verschiedener partieller Ableitungen [75]. Beispiele für solche Verfahren sind der bereits vorgestellte Harris-Detektor, der partielle Ableitungen erster Ordnung kombiniert (Abschnitt 4.1.2) und der Determinant Of Hessian-Detektor, bei dem partielle Ableitungen zweiter Ordnung eingesetzt werden (Abschnitt 4.1.3). Diese beiden Detektoren sind zwar nicht skalierungsinvariant, können aber durch Anwendung über Gaußschen Skalenräumen dahingehend erweitert werden.

Verfahren, die *ableitungsbasierte Interest-Funktionen* einsetzen und die invariant gegenüber Skalierungsveränderungen sind, ermitteln neben der Position auch die sogenannte

charakteristische Skalierung eines lokalen Features [75]. Die charakteristische Skalierung bestimmt die Größe einer skalierungsinvarianten Interest-Region um jedes Feature [43]. Um die Position eines lokalen Features gemeinsam mit der charakteristischen Skalierung zu ermitteln, wird ein Skalenraum $L(x, y, \sigma)$ für das betrachtete Eingangsbild I konstruiert (Abschnitt 2.3). Auf jeder Ebene des Skalenraums wird dann eine skalennormalisierte Interest-Funktion $IF^n(x, y, \sigma)$ berechnet. Durch Auffinden von Stellen, die ein Extremum in diesem dreidimensionalen Skalenraum annehmen, d.h. sowohl in Richtung der Ortskoordinaten x, y als auch in Richtung der Skalierung σ, können dann lokale Features gemeinsam mit ihrer charakteristischen Skalierung identifiziert werden [73]. Die Vorgehensweise wird in den folgenden Abschnitten genauer ausgeführt. Die Abschnitte orientieren sich dabei an den Arbeiten von Mikolajczyk [43] und Lindeberg [34].

Gaußsche Skalenräume werden durch Faltung des Eingangsbildes I mit immer größeren Gauß-Filtern $H^{G(\sigma)}$ erzeugt als $L(x, y, \sigma) = I(x, y) \otimes H^{G(\sigma)}$ (Abschnitt 2.3). Die Größe des Gauß-Filters wird durch σ gesteuert. Wir bezeichnen im Folgenden *Ableitungen,* die *über einem Gaußschen Skalenraum* $L(x, y, \sigma)$ berechnet werden, mit $L_{i_m}(x, y, \sigma)$. Dabei beschreibt m die Ordnung der Ableitung und i die Richtung der Ableitung in den Ortskoordinaten x, y (vgl. [43]). Die Ableitung erster Ordnung nach x in einem Skalenraum wird beispielsweise als $L_x(x, y, \sigma)$ bezeichnet. Im Allgemeinen werden Ableitungen im Skalenraum realisiert, indem das Eingangsbild $I(x, y)$ sukzessive mit entsprechenden Gaußschen Ableitungsfiltern $H^{G_{i_m}(\sigma)}$ mit immer größerem σ gefaltet wird [40], d.h. [34]:

$$L_{i_m}(x, y, \sigma) = I(x, y) \otimes H^{G_{i_m}(\sigma)} \tag{4.10}$$

Der Grad der Glättung des Eingangsbildes auf den einzelnen Ebenen des Skalenraums wird durch den Parameter σ der angewendeten Gauß-Filter gesteuert. Im Eingangsbild vorhandene Bildstrukturen werden mit zunehmendem Skalenparameter immer glatter (siehe 2.3). Auch die Amplitude der im Skalenraum berechneten Ableitungen wird mit steigendem σ geringer [73]. Aus diesem Grund müssen die berechneten Ableitungen auf den unterschiedlichen Skalenebenen in Bezug auf die Skalierung normalisiert werden [34]. Nur so kann ein vergleichbares Ergebnis auf verschiedenen Skalenebenen berechnet und Skalierungsinvarianz erreicht werden. Lindeberg zeigte, dass *skalennormalisierte Ableitungen* der Ordnung m durch Multiplikation mit einem Faktor σ^m realisiert werden können als [43]:

$$L^n_{i_m}(x, y, \sigma) = \sigma^m I(x, y) \otimes H^{G_{i_m}(\sigma)} \tag{4.11}$$

Ausführliche Informationen zur Berechnung von skalennormalisierten Ableitungen finden sich in [34, 37].

Die gewünschten skalennormalisierten Interest-Funktionen IF^n können nun als Kombinationen dieser gauß-basierten, skalennormalisierten Ableitungen realisiert werden. Zur Detektion von lokalen Features werden dann auf jeder Ebene des Skalenraums entsprechende skalennormalisierte Interest-Funktionen berechnet als $IF^n(x, y, \sigma)$. Das Ergebnis ist ein dreidimensionaler Skalenraum, bei dem jede Ebene der berechneten Interest-Funktion für eine andere Auflösungsstufe des Eingangsbildes entspricht. Lindeberg führte das Konzept der automatischen Skalenselektion ein, das besagt, dass Extrema in Richtung

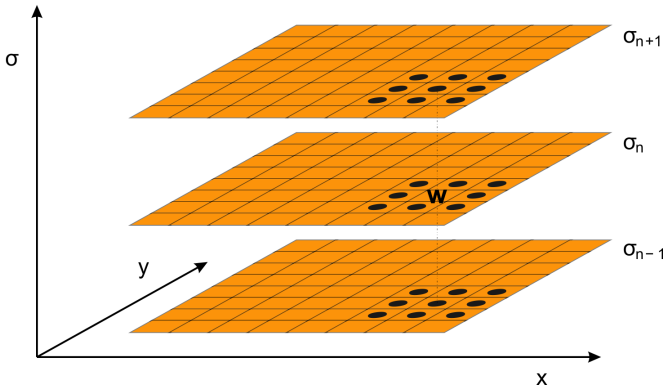


Abbildung 4.6: Suche nach Extrema im mittels einer Interest-Funktion $IF^n(x, y, \sigma)$ aufgebauten dreidimensionalen Skalenraum: Ein Punkt w bildet ein Maximum, wenn sein Wert größer ist als der seiner benachbarten Bildelemente auf derselben Skalenebene σ_n und der Bildelemente auf benachbarten Skalenebenen σ_{n+1} und σ_{n-1}; die benachbarten Bildelemente des Bildpunktes w sind in der Darstellung durch schwarze Punkte gekennzeichnet (siehe [44])

der Skalierung von im Skalenraum berechneten normalisierten Ableitungen der charakteristischen Skalierung eines lokalen Features entsprechen (siehe Abbildung 4.5). Lokale Features können bei ableitungsbasierten Verfahren aus diesem Grund gemeinsam mit ihrer charakteristischen Skalierung σ bestimmt werden, indem im Skalenraum $IF^n(x, y, \sigma)$ nach Stellen (x, y, σ) gesucht wird, die ein lokales Extremum sowohl in Richtung der Ortskoordinaten x, y als auch in Richtung der Skalierung σ annehmen (siehe Abbildung 4.6) [43, 75].

Ausführliche Informationen zur automatischen Skalenselektion können in [43, 37] gefunden werden. In den nachfolgenden Unterabschnitten werden einige skalierungsinvariante Detektoren vorgestellt, die auf der Bildung von Ableitungen im Skalenraum und automatischer Skalenselektion basieren. Die einzelnen Detektoren unterscheiden sich vorwiegend durch die eingesetzten Interest-Funktionen. Bei dem im nächsten Abschnitt vorgestellten Harris Laplace- und Hesse Laplace-Detektor werden dabei zwei unterschiedliche Interest-Funktionen zur Identifizierung von Extrema in Richtung der Ortskoordinaten x, y und in Richtung der Skalierung σ eingesetzt.

4.2.1.1 Der Harris Laplace-Detektor und der Hesse Laplace-Detektor

Der Harris Laplace-Detektor und der Hesse Laplace-Detektor [47] sind skalierungsinvariante Erweiterungen des Harris-Detektors (Abschnitt 4.2.1.1) und des Hesse-Detektors (Abschnitt 4.1.3). Beide Verfahren detektieren Interest-Points gemeinsam mir ihrer charakteristischen Skalierung basierend auf der Bildung von Ableitungen. Dazu werden zunächst mittels einer skalennormalisierten Interest-Funktion Positionen als Interest-Points auf jeder Ebene eines Skalenraums identifiziert, die ein Extremum in Richtung der Orts-

koordinaten bilden. Für diese wird dann mit Hilfe des skalennormalisierten Laplace-Operators die charakteristische Skalierung bestimmt [75]. Als Ausgabe liefern beide Detektoren skalierungsinvariante Interest-Regions. Zur Ermittlung der Extrema in Richtung der Ortskoordinaten x, y werden die beim Harris-Detektor eingesetzte Autokorrelationsmatrix (Gleichung 4.3) und die Hesse-Matrix (Gleichung 4.7) unter Verwendung Gaußscher Ableitungsfilter so umformuliert, dass eine Berechnung im Skalenraum ermöglicht wird [73].

Die *skalennormalisierte Autokorrelationsmatrix* μ^n ist gegeben durch [44]:

$$\mu^n(x, y, \sigma_I, \sigma_D) = \sigma_D^2 H^{G(\sigma_I)} \otimes \begin{bmatrix} L_x^2(x, y, \sigma_D) & L_x L_y(x, y, \sigma_D) \\ L_x L_y(x, y, \sigma_D) & L_y^2(x, y, \sigma_D) \end{bmatrix}$$

Der Parameter σ_D wird als Differentiationsskala bezeichnet und legt die Größe des gaußbasierten Filters fest, der zur Berechnung der Ableitungen L_{i_m} im Skalenraum der Ordnung m in Richtung i herangezogen wird.

Wie schon beim Harris-Detektor werden die berechneten Ableitungen in einer Nachbarschaft um das betrachtete Bildelement (x, y) durch Faltung mit einem Gauß-Filter $H^{G(\sigma_I)}$ geglättet. Die Größe der zur Glättung betrachteten Nachbarschaft wird durch die sogenannte Integrationsskala σ_I festgelegt. Durch den Faktor σ_D^2 wird eine Normalisierung der Ergebnisse in Bezug auf die Skalierung nach Lindeberg realisiert [47].

Der *skalennormalisierte Harris-Operator* IF_{Harr}^n wird aus der Matrix μ^n bestimmt und ist folgendermaßen definiert [45]:

$$IF_{Harr}^n(x, y, \sigma_n) = Det(\mu^n(x, y, \sigma_n)) - \alpha Tr^2(\mu^n(x, y, \sigma_n))$$

Zur Identifikation von Interest-Points wird nun die Funktion IF_{Harr}^n auf allen Ebenen σ_n der Skalenraum-Repräsentation des Eingangsbildes I angewendet. Die skalennormalisierte Harris-Funktion IF_{Harr}^n ist, wie schon die einfache Harris-Funktion (Gleichung 4.4), dazu geeignet, eckpunkt-ähnliche Strukturen als Interest-Points in Bildern zu identifizieren.

Nach der Berechnung der Funktion IF_{Harr}^n für jede Auflösungsstufe des Eingangsbildes können Interest-Points an Stellen $w = (x, y)$ identifiziert werden, an denen ein lokales Maximum innerhalb einer Nachbarschaft M angenommen wird, das über einem bestimmten Schwellwert T_r liegt. Beim Harris Laplace-Detektor wird dazu der Wert jedes Bildelements w mit den Werten seiner acht benachbarten Bildelemente w_M auf derselben Skalenebene verglichen und überprüft, ob folgende zwei Bedingung erfüllt sind [44]:

$$\begin{aligned} IF_{Harr}^n(w, \sigma_n) &> IF_{Harr}^n(w_M, \sigma_n) \;\; \forall w_M \in M \\ IF_{Harr}^n(w, \sigma_n) &> T_r \end{aligned} \tag{4.12}$$

Die *skalennormalisierte Hesse-Matrix* $\eta^n(x, y)$ basiert auf Bildung partieller Ableitungen zweiter Ordnung und ist definiert als [45]:

$$\eta^n(x, y, \sigma_D) = \sigma_D^2 \begin{pmatrix} L_{xx}(x, y, \sigma_D) & L_{xy}(x, y, \sigma_D) \\ L_{xy}(x, y, \sigma_D) & L_{yy}(x, y, \sigma_D) \end{pmatrix} \tag{4.13}$$

Der skalennormalisierte Determinant Of Hessian-Operator IF^n_{Hess} wird als Determinante der Matrix η^n berechnet und ist definiert als:

$$IF^n_{Hess} = Det(\eta^n(x, y, \sigma_n))$$

Analog zum Harris Laplace-Detektor werden Interest-Points identifiziert, indem die skalennormalisierte Determinant Of Hessian-Funktion IF^n_{Hess} auf jeder Ebene σ_n eines Gaußschen Skalenraums angewendet wird und dann nach lokalen Extrema in einer Nachbarschaft M in Richtung der Ortskoordinaten gesucht wird (siehe Gleichung 4.12). Die identifizierten Extrema entsprechen blob-ähnlichen Strukturen im Bild.

Sowohl beim Harris Laplace-Detektor als auch beim Hesse Laplace-Detektor werden die identifizierten Interest-Points, die Extrema in Richtung x, y bilden, zum Auffinden der charakteristischen Skalierung mit Hilfe des skalennormalisierten Laplacian Of Gaussian-Operators weiter untersucht. Dieser wird folgendermaßen berechnet [44]:

$$|IF^n_{LoG}(x, y, \sigma_n)| = \sigma_n^2 |L_{xx}(x, y, \sigma_n) + L_{yy}(x, y, \sigma_n)|$$

Die gesuchten Interest-Regions werden dann als Punkte x, y mit Skalierung σ_n bestimmt, für die der Operator $|IF^n_{LoG}|$ ein Maximum in Richtung der Skalierung σ annimmt, das über einem bestimmten Schwellwert T_s liegt. Es wird also überprüft, ob folgende Bedingungen erfüllt sind [44]:

$$\begin{aligned} |IF^n_{LoG}(x, y, \sigma_n)| &> |IF^n_{LoG}(x, y, \sigma_{n-1})\ | \wedge\ |IF^n_{LoG}(x, y, \sigma_n)| > |IF^n_{LoG}(x, y, \sigma_{n+1})| \\ |IF^n_{LoG}(x, y, \sigma_n)| &> T_s \end{aligned}$$

Die Verwendung des Harris Laplace-Detektors beziehungsweise des Hesse Laplace-Detektors ermöglicht die Detektion von Interest-Regions, die robust gegenüber Drehungen, Verschiebungen und Skalierungsveränderungen von Bildern sind [47]. Mikolayczyk u.a. schlugen Erweiterungen der beiden Detektoren vor, bei denen iterativ die ermittelte Position und Skalierung der lokalen Features verfeinert wird bis das Ergebnis konvergiert. Ausführliche Informationen zu diesen Verfahren können in [43, 47] gefunden werden.

4.2.1.2 Der Laplacian Of Gaussian-Detektor und der Difference Of Gaussian-Detektor

Sowohl der Laplacian Of Gaussian-Detektor (LoG-Detektor) als auch der Difference Of Gaussian-Detektor (DoG-Detektor) sind skalierungsinvariante Verfahren zur Identifizierung von blob-ähnlichen Strukturen in Bildern. Blobs werden bei beiden Verfahren zusammen mit ihrer charakteristischen Skalierung identifiziert, indem eine Skalenraum-Repräsentation des Eingangsbildes mit Hilfe einer skalennormalisierten Interest-Funktion aufgebaut wird und nach Stellen gesucht wird, die gleichzeitig ein lokales Extremum in Richtung der Ortskoordinaten und in Richtung der Skalierung annehmen. Die Detektoren unterscheiden sich lediglich durch die über dem Skalenraum eingesetzte Interest-Funktion [75].

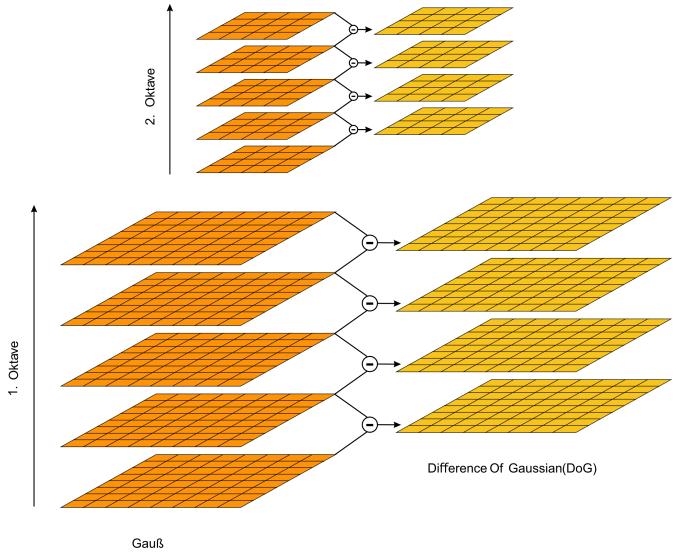


Abbildung 4.7: Aufbau eines DoG-Skalenraums (rechts) durch Differenzbildung zwischen benachbarten Skalenebenen eines Gaußschen Skalenraums (links) nach Lowe [39]

Die eingesetzte Interest-Funktion für den *LoG-Detektor* IF_{LoG} entspricht dem skalennormalisierten LoG-Operator und somit der Spur der skalennormalisierten Hesse-Matrix (Gleichung 4.13). Sie ist folgendermaßen definiert [37, 44]:

$$\begin{aligned} IF^n_{LoG}(x,y,\sigma) &= \sigma^2(L_{xx}(x,y,\sigma) + L_{yy}(x,y,\sigma)) \\ &= Tr(\eta^n(x,y,\sigma_n)) \end{aligned}$$

Die für den *DoG-Detektor* eingesetzte Interest-Funktion IF^n_{DoG} entspricht einer Approximation des LoG-Operators. Der DoG wird durch Differenzbildung zwischen benachbarten Skalenebenen in einem Gaußschen Skalenraum realisiert und ist definiert als [39]:

$$IF^n_{DoG}(x, y\sigma) = L(x,y,k\sigma) - L(x,y,\sigma)$$

Die Beziehung des DoG-Operators IF^n_{DoG} zum skalennormalisierten Laplace-Operator ist dabei gegeben durch [39]:

$$IF^n_{DoG} \approx (k-1)IF^n_{LoG}$$

Dieser Zusammenhang lässt sich nach [75] dadurch erklären, dass die Anwendung des LoG-Operators im Skalenraum einer Ableitung des Bildes in Richtung der Skalierung σ entspricht. Wie in Abschnitt 2.2.2.2 ausgeführt, können partielle Ableitungen in verschiedene Richtungen durch Bildung von Differenzen zwischen benachbarten Bildelementen in die jeweilige Richtung angenähert werden. Somit kann die Ableitung in Richtung

der Skalierung durch Bildung der Differenz zwischen zwei benachbarten Skalenebenen $L(x, y, k\sigma)$, $L(x, y, \sigma)$ approximiert werden [39].

Abbildung 4.7 veranschaulicht, wie beim Aufbau eines Skalenraums mit Hilfe der DoG-Funktion vorgegangen werden kann [39]: Durch sukzessive Faltung des Eingangsbildes I mit Gauß-Filtern $H^{G(\sigma)}$ wird eine Skalenraum-Repräsentation aufgebaut. Die Standardabweichung σ der Gauß-Filter, mit denen die einzelnen Ebenen des Skalenraums berechnet werden, unterscheiden sich jeweils durch einen konstanten Faktor k. Der Skalenraum wird in sogenannte Oktaven unterteilt, wobei jede Oktave einer Verdopplung von σ entspricht. Um den Rechenaufwand zu verringern, wird am Übergang zwischen einzelnen Oktaven jeweils ein Downsampling durchgeführt [39]. Dabei wird das Bild neu abgetastet und verkleinert (Abschnitt 2.3) [40]. Es wird nur jedes zweite Bildelement einer Zeile oder Spalte in das neu erzeugte Bild übernommen. Aus der so erhaltenen Repräsentation des Eingangsbildes (Abbildung 4.7 links) wird die Differenz zweier Bilder auf jeweils benachbarten Skalenebenen gebildet und so die DoG-Repräsentation (Abbildung 4.7 rechts) erzeugt.

Lokale Features werden beim LoG- und beim DoG-Detektor gemeinsam mit ihrer charakteristischen Skalierung ermittelt, indem Stellen ausgewählt werden, die ein Extremum im aufgebauten Skalenraum annehmen. Es wird dazu der Wert eines Bildelements $w = (x, y)$ mit den Werten seiner 8 Nachbarn auf derselben Skalenebene und denen der insgesamt 18 benachbarten Bildelemente auf den angrenzenden Skalenebenen verglichen [39].

In der Praxis wird der DoG-Detektor häufig dem LoG-Detektor vorgezogen, da er eine gute Approximation desselben liefert, aber viel effizienter berechenbar ist, da keine Ableitungen ermittelt werden müssen [75]. Der DoG-Detektor wird zudem für die Identifizierung von Kandidatenpunkten in dem von Lowe vorgestellten SIFT-Algorithmus eingesetzt [39]. Lowe stellt in seiner Arbeit außerdem einige Schritte zur Nachbearbeitung der mittels des DoG-Detektors identifizierten lokalen Features vor, die aber auch für den LoG-Detektor eine Verbesserung der Ergebnisse bewirken können (siehe [39]).

Bei der Nachbearbeitung wird wie folgt vorgegangen [39]: Es werden zunächst die Positionen x, y und Skalierungen σ der identifizierten Extrema verfeinert. Da zur Ermittlung dieser Extrema nur diskrete Stellen betrachtet werden, könnte das "echte" Extremum auch an irgendeiner Zwischenstelle liegen. Daher wird die Interest-Funktion in der Umgebung eines Extremums durch eine quadratische Taylorreihe angenähert. So kann ermittelt werden, ob das tatsächliche Extremum eigentlich näher an einem anderen als der identifizierten Bildposition liegt und dementsprechend verschoben werden muss. Der Vorgang wird iterativ wiederholt. Zusätzlich werden Extrema, deren Absolutwert unter einem bestimmten Schwellwert liegt zurückgewiesen, da sie anfällig gegenüber Bildstörungen sind [39]. Durch Berechnung der Hesse-Matrix für die verbleibenden Punkte und Auswertung der Konfiguration der Eigenwerte können schließlich identifizierte Extrema ausgeschlossen werden, die auf einer Kante liegen [75] (Abschnitt 4.1.3). Diese können nur schlecht lokalisiert werden und wären daher der Wiederholbarkeit des Detektors abträglich (siehe Abschnitt 3.1.1).

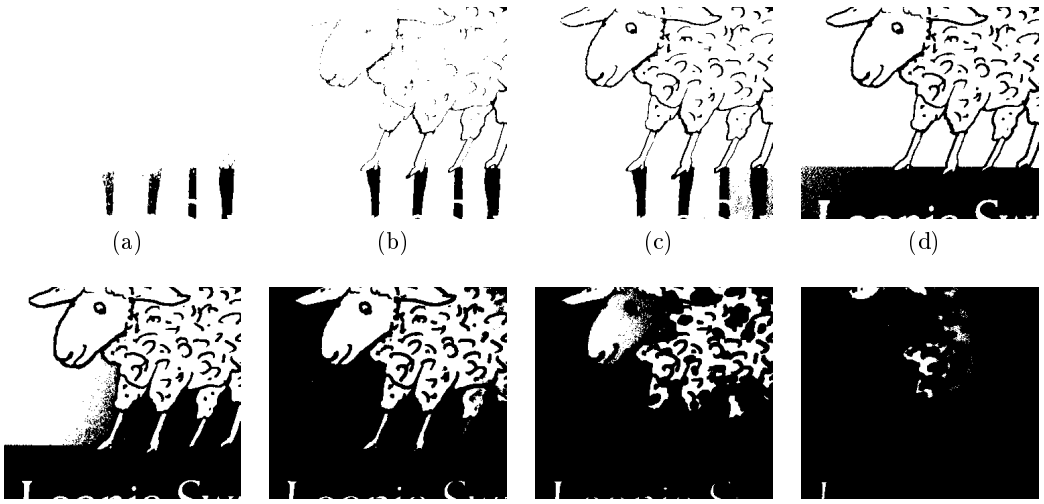


Abbildung 4.8: Erstellen verschiedener Schwellwertbilder beim MSER-Detektor: Das Eingangsbild wird mit immer größerem Schwellwert T in eine Folge von Binärbildern übergeführt

4.2.2 MSER-Detektor

MSER steht für *Maximally Stable Extremal Regions* und bezeichnet ein von Matas u.a. entwickeltes Verfahren zur Detektion affin-invarianter Interest-Regions, das auf einer Art von Segmentierung beruht [42]. Der MSER-Detektor wählt zusammenhängende[5] Regionen Q in Bildern als lokale Features aus, die heller oder dunkler als ihre direkte Umgebung ∂Q sind und innerhalb derer sich die Intensitätswerte nur minimal ändern [75]. Die Menge ∂Q wird als äußere Grenze der Region Q bezeichnet. Zur äußeren Grenze zählen dabei alle Bildelemente, die zu zumindest einem Bildelement der Menge Q benachbart sind, aber aufgrund ihrer Intensitätswerte nicht zu Q gehören [42].

Zur Bestimmung dieser Regionen Q homogener Intensitätswerte wird der Reihe nach jeder Intensitätswert im Eingangsbild einmal als Schwellwert T verwendet[6], um ein Schwellwertbild zu erzeugen [42]. In einem solchen Schwellwertbild bekommt jedes Bildelement $p = (x, y)$ den Wert 0 oder 1 zugeordnet, je nachdem, ob sein Intensitätswert $I(x, y)$ über oder unter dem festgelegten Schwellwert T liegt. Für jeden betrachteten Schwellwert T wird somit ein Binärbild aus dem Eingangsbild erzeugt.

Matas u.a. erklären diesen Vorgang sehr anschaulich mit Hilfe einer Film-Analogie, die hier auch kurz erläutert werden soll (vgl. [42]): Man stelle sich einen Film vor, bei dem jedes der gezeigten Einzelbilder (engl.: Frames) einem Schwellwertbild I_T entspricht, das mit einem bestimmten Schwellwert T erzeugt wurde. Der Wert von T durchläuft nacheinander alle Intensitätswerte I des Eingangsbildes in aufsteigender Reihenfolge. Zu Beginn des Films sieht man daher zunächst nur ein weißes Bild. Mit der Zeit erscheinen

[5] Für die Überprüfung des Zusammenhangs wird eine 4rer Nachbarschaft verwendet (siehe [10]).
[6] Für ein Grauwertbild mit 8 Bit ist $T \epsilon \{0, ..., 255\}$.

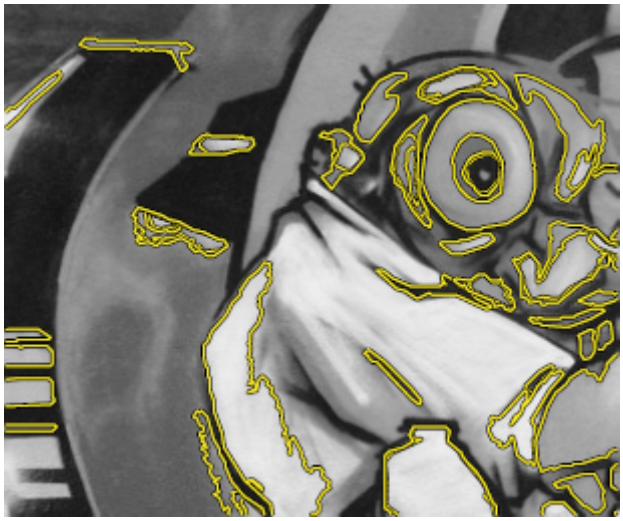


Abbildung 4.9: Vom MSER-Detektor identifizierte Interest-Regions in der ursprünglichen Form; Diese unregelmäßigen Regionen werden zumeist durch Ellipsen approximiert; Bild aus [75]

immer mehr dunkle Stellen im Bild. Diese dehnen sich allmählich weiter aus und formen zusammenhängende Regionen, so lange, bis die dunklen Regionen gegen Ende des Films zusammenwachsen und schlussendlich ein schwarzes Bild entsteht. Dieser Prozess wird in Abbildung 4.8 anhand eines Beispielbildes illustriert. Im Wesentlichen entspricht der Vorgang einer Watershed-Transformation [5], bei dem das betrachtete Bild ausgehend von Stellen, deren Intensitätswerte lokalen Minima im Bild entsprechen, "geflutet" wird. Jede Erhöhung des Schwellwerts T entspricht dann einem Ansteigen der Wassersäule.

Die Menge aller in den Schwellwertbildern gefundenen, zusammenhängenden Regionen bildet die Menge der Extremal Regions. Das Adjektiv Extremal weist hier darauf hin, dass sich die Intensitätswerte der Bildelemente innerhalb der Regionen Q deutlich von denen der Bildelemente ∂Q außerhalb der Grenze der Region unterscheiden [75], d.h. dass für alle Bildelemente $p \epsilon Q,\ q \epsilon \partial Q$ gilt $I(p) > I(q)$ oder $I(p) < I(q)$ [42]. Als Maximally Stable Extremal Regions werden schließlich diejenigen Extremal Regions ausgewählt, die über eine große Anzahl von Schwellwertbildern stabil sind, d.h. die ihre Form über eine längere Zeit nicht verändern. Zum Identifizieren solcher Regionen werden die gefundenen Extremal Regions sortiert, sodass $Q_1 \subset \cdots \subset Q_i \subset Q_{i+1} \subset Q_n$ gilt. Für das Vorliegen einer Maximally Stable Extremal Region Q_{i*} muss der Ausdruck $q(i) = |Q_{i+\triangle} \setminus Q_{i-\triangle}|/|Q_i|$ ein lokales Minimum für $i*$ annehmen. Dabei bezeichnet $|Q_{i*}|$ die Anzahl der Bildelemente in Q_{i*} [42].

Die vom MSER-Detektor identifizierten Features können nach [75] blob-ähnlichen Strukturen entsprechen aber auch anderen Strukturen mit komplexeren Objektgrenzen. Der MSER-Detektor liefert als Ergebnis daher Interest-Regions, die beliebige Formen annehmen können (siehe Abbildung 4.9). Meistens wird ihre Form jedoch durch Ellipsen approximiert, um Speicherkosten zu reduzieren und weitere Bearbeitungsschritte zu ver-

einfachen [73]. Der MSER-Detektor ist invariant gegenüber affinen Transformationen. Er liefert die besten Resultate, wenn er für Bilder eingesetzt wird, die deutlich voneinander abgegrenzte Strukturen aufweisen [75].

5 Umsetzung

Im Rahmen der vorliegenden Arbeit wurde eine Applikation entwickelt, die den visuellen Vergleich der Ergebnisse ausgewählter Interest-Point-Detektoren ermöglicht. Die Applikation wird im Folgenden als VisIP (Visual Interest-Points) bezeichnet. In diesem Kapitel wird beschrieben, wie VisIP realisiert wurde. Auf die zur Entwicklung von VisIP eingesetzte Umgebung und verwendete Zusatzfunktionen wird in Abschnitt 5.1 eingegangen. In Abschnitt 5.2 wird erklärt, welche Detektoren für die Integration in VisIP ausgewählt wurden und warum. Anschließend wird auf die verwendeten Implementierungen der Verfahren zur Interest-Point-Detektion eingegangen (siehe Abschnitt 5.3). Den Abschluss dieses Kapitels bildet eine Beschreibung von VisIP selbst, der Funktionalität und der Bedienung (Abschnitt 5.4).

5.1 Entwicklungsumgebung und Zusatzfunktionen

VisIP wurde unter Microsoft-Windows in *MATLAB* (R2010B) implementiert. MATLAB ist eine von MathWorks entwickelte Software, die eine Umgebung für die effiziente Berechnung mathematischer Aufgaben bietet[1]. Das Akronym MATLAB steht für MAtrix LABratory und weist darauf hin, dass Operationen in MATLAB zumeist auf Matrizen oder Vektoren durchgeführt werden. MATLAB eignet sich sehr gut für die Verarbeitung von bildbasierten Daten, da digitale Bilder nichts anderes als Matrizen sind, deren Koeffizienten Farbwerten entsprechen.

Neben nützlichen Funktionen zur Verarbeitung und Darstellung von Daten werden in MATLAB anwendungsspezifische Funktionen in Form sogenannter Toolboxen bereitgestellt. Für die Realisierung der vorliegenden Arbeit waren vor allem die in der *Image Processing Toolbox* enthaltenen Funktionen hilfreich, wie beispielsweise lineare Faltung und Funktionen zum Erstellen unterschiedlicher Filter.

Die graphische Benutzeroberfläche von VisIP wurde mittels *GUIDE* realisiert, einem in MATLAB integrierten graphischen Layout-Editor. Auf der Website des MATLAB-Herstellers findet sich zudem eine Plattform für den Austausch ergänzender MATLAB-Funktionen, die von verschiedenen MATLAB-Nutzern programmiert wurden[2]. Für VisIP wird die auf dieser Austauschplattform zur Verfügung gestellte Funktion *DRAGZOOM* verwendet. Diese ermöglicht unter anderem das interaktive Zoomen in dargestellten Bildern[3]. Die Implementierungen der in VisIP integrierten Interest-Point-Detektoren stam-

[1] http://www.mathworks.de/products/matlab/ (zuletzt abgerufen am 13.11.2011)
[2] http://www.mathworks.com/matlabcentral/fileexchange/ (zuletzt abgerufen am 13.11.2011)
[3] http://www.mathworks.com/matlabcentral/fileexchange/29276-dragzoom-drag-and-zoom-tool (zuletzt abgerufen am 13.11.2011)

Art	Detektor	Eckp.	Blob	Region	Rot	Skal	Affin	Grundlage
Point	Harris	+			+			1. Ableitung
	DoH		+		+			2. Ableitung
	FAST	+			+			Morphologie
Region	Harris Lap	+	(+)		+	+		1. Ableitung
	Hesse Lap	(+)	+		+	+		2. Ableitung
	LoG		+		+	+		2. Ableitung
	DoG		+		+	+		2. Ableitung
	MSER		(+)	+	+	+	+	Segmentierung

Tabelle 5.1: Übersicht der in VisIP integrierten Detektoren und ihrer Einteilung: Unterscheidung zwischen Interest-Point-Detektoren und Interest-Region-Detektoren, Eckpunkt-, Blob-, oder Regionen-Detektoren. Die Detektoren unterscheiden sich außerdem im Grad ihrer Robustheit gegenüber Bildtransformationen [75]

men aus unterschiedlichen Quellen. Nähere Informationen dazu finden sich in Abschnitt 5.3.

5.2 Auswahl der Detektoren

In der Praxis werden zur Detektion lokaler Features zumeist intensitätsbasierte Verfahren eingesetzt, da sie, im Gegensatz zu modellbasierten- oder konturbasierten Verfahren, für den Einsatz auf unterschiedlichen Arten von Bildern und für ein breites Spektrum von Anwendungen geeignet sind (Abschnitt 3). Bei den in VisIP integrierten Methoden handelt es sich aus diesem Grund ausschließlich um intensitätsbasierte Verfahren.

VisIP ermöglicht die Detektion und Darstellung lokaler Features mittels des Harris-, Harris Laplace-, Determinant Of Hessian-, Hesse Laplace-, Laplacian Of Gaussian-, Difference Of Gaussian-, FAST- und MSER-Detektors. Diese Detektoren gehören zu den in der Praxis am häufigsten eingesetzten. Zudem wurde darauf geachtet, Vertreter unterschiedlicher Kategorien von Detektoren auszuwählen. In Tabelle 5.1 ist die getroffene Einteilung der selektierten Detektoren dargestellt (vgl. [75]). In VisIP sind sowohl Blob- als auch Eckpunkt-Detektoren integriert, die einen jeweils unterschiedlichen Grad an Robustheit gegenüber Rotationen, Skalierungsveränderungen und affinen Transformationen aufweisen. Die Grundlage zur Identifizierung von lokalen Features mit Hilfe der ausgewählten Detektoren bilden Kombinationen partieller Ableitungen erster Ordnung (Harris, Harris Laplace) oder partieller Ableitungen zweiter Ordnung (DoH, Hesse Laplace, LoG, DoG), morphologischen Operatoren (FAST) oder einer Art von Segmentierung (MSER) (Abschnitt 4). In Abhängigkeit von der Art der Ausgabe des jeweiligen Detektors wird wird zwischen Interest-Point- und Interest-Region-Detektoren unterschieden (Abschnitt 4).

Detektor	Quelle
Harris	MATLAB
DoH	Laptev
FAST	Rosten
Harris Laplace	LIP-VIREO
Hesse Laplace	LIP-VIREO
LoG	LIP-VIREO
DoG	LIP-VIREO
MSER	LIP-VIREO

Tabelle 5.2: Übersicht der Quellen der in VisIP eingesetzten Implementierungen der einzelnen Detektoren

5.3 Implementierungen der Detektoren

Implementierungen von Verfahren zur Detektion von lokalen Features sind in verschiedenen Ausführungen von unterschiedlichen Entwicklern verfügbar. Nachdem die Detektoren festgelegt waren, welche in VisIP berechenbar sein sollten, wurde nach geeigneten Implementierungen dieser Verfahren gesucht. Grundsätzlich sollten die Implementierungen folgende Kriterien erfüllen:

- Die Implementierung sollte in MATLAB unter Windows verwendbar sein (Abschnitt 5.1). Oftmals werden die implementierten Algorithmen nur für den Einsatz unter Linux zur Verfügung gestellt. Solche Anwendungen kamen für die Verwendung in VisIP nicht in Frage[4].

- Damit eine möglichst unkomplizierte Bedienung von VisIP ermöglicht wird, sollten die ausgewählten Detektoren einheitlich steuerbar sein, das heißt es sollten gleiche oder ähnliche Parameter übergeben werden können. Dies war problematisch, da bei vielen der untersuchten Implementierungen oft entweder gar keine Eingabeparameter übergeben werden konnten und somit das Ergebnis des Detektors in keiner Weise beeinflussbar war, oder zu verschiedenartige Eingabeparameter erforderlich waren. Für alle in VisIP integrierten Detektoren kann nun die Anzahl `n` der lokalen Features übergeben werden, die detektiert werden sollen. Durch den jeweiligen Detektor werden dann, soweit möglich, die `n` lokalen Features zurückgeliefert, die den höchsten Wert für die jeweils eingesetzte Interest-Funktion aufweisen, das heißt die am stärksten ausgeprägten Eckpunkte oder Blobs.

- Die ausgewählten Implementierungen sollten vergleichbare Informationen als Ausgabe zur Darstellung der detektierten lokalen Features liefern. Alle für VisIP ausgewählten Implementierungen der Detektoren liefern die Positionen der detektierten

[4]Ein Beispiel dafür ist die Sammlung verschiedener Interest-Point-Detektoren und -Deskriptoren von Gyuri Dorkó, zu finden unter http://lear.inrialpes.fr/people/dorko/downloads.html (zuletzt abgerufen am 01.11.2011).

lokalen Features als Ausgabe. Für die skalierungs- und affin-invarianten Detektoren werden zudem Informationen geliefert, die das Darstellen einer kreisförmigen- oder ellipsenförmigen Interest-Region zu jedem Feature ermöglichen.

- Damit die Ergebnisse der einzelnen Detektoren nachvollzogen werden können, sollten sich die verwendeten Implementierungen so nahe wie möglich an den in Abschnitt 4 vorgestellten Bearbeitungsschritten zur Berechnung der lokalen Features orientieren.

Basierend auf den genannten Auswahlkriterien wurden die in Tabelle 5.2 rechts gelisteten Implementierungen der Detektoren für die Integration in VisIP ausgewählt. Diese stammen aus unterschiedlichen Quellen (MATLAB, Laptev, Rosten oder LIP-Vireo) und werden nachfolgend ausführlicher erklärt. Dabei wird primär auf die Parameter der Detektoren eingegangen, die in VisIP tatsächlich zum Einsatz kommen. Alle Detektoren identifizieren lokale Features in Grauwertbildern.

Harris-Detektor: Der in Abschnitt 4.1.2 vorgestellte Harris-Detektor wird in VisIP mit Hilfe der in MATLAB integrierten Funktion `corner` berechnet[5]. Die Funktion kann folgendermaßen aufgerufen werden:

```
c = corner(I, 'Harris', n)
```

Als Eingangsparameter werden das Bild `I` in dem lokale Features detektiert werden sollen und der Name ('`Harris`') des ausgewählten Detektors übergeben. Neben dem Harris-Detektor kann mit der Funktion `corner` auch der Detektor von Shi & Thomasi berechnet werden [67]. Weiters wird die maximale Anzahl (`n`) der lokalen Features übergeben, die zurückgeliefert werden sollen. Dabei werden die lokalen Features ausgewählt, die die höchsten Werte für die Harris-Interest-Funktion aufweisen (Gleichung 4.4).

Zusätzlich können über die Eingabeparameter der `corner` Funktion die Filterkoeffizienten des Gauß-Filters $H^{G(\sigma)}$ angegeben werden, der zur Glättung der berechneten Ableitungen eingesetzt wird (siehe Gleichung 4.3), sowie der Wert der Konstante α der die Empfindlichkeit des Harris-Detektors steuert (siehe Gleichung 4.4). Für die Umsetzung von VisIP werden die in MATLAB gesetzten Standardeinstellungen verwendet mit $\alpha = 0.04$ und $H^{G(\sigma)}$ ein $5x5$ Gauß-Filter mit $\sigma = 1.5$.

Als Ausgabe liefert die `corner` Funktion eine Matrix `c`, die die Positionen (`x,y`) der detektierten Interest-Points enthält. In der MATLAB-Online-Hilfe können genauere Informationen zu dieser und allen anderen in MATLAB integrierten Funktionen gefunden werden[6].

Determinant Of Hessian-Detektor: Zur Berechnung des Determinant Of Hessian-Detektors wird eine von Ivan Laptev zur Verfügung gestellte Sammlung von MATLAB-Funktionen eingesetzt[7]. Die darin enthaltene Funktion `intpointdet` realisiert Implementierungen mehrerer verschiedener Interest-Point-Detektoren, wie beispielsweise des

[5] http://www.mathworks.de/help/toolbox/images/ref/corner.html (zuletzt abgerufen am 27.10.2011)
[6] http://www.mathworks.de/help/index.html (zuletzt abgerufen am 27.10.2011)
[7] http://www.nada.kth.se/~laptev/code.html Datei: affintpoints.zip (zuletzt abgerufen am 28.10.2011)

Harris-Detektors, eines Detektors auf Basis des Laplace-Operators und auch des DoH-Detektors. Dieser wird in VisIP durch einen Aufruf der Funktion `intpointdet` in folgender Form berechnet:

```
pos = intpointdet(I, kparam, sxl2, sxi2, pointtype, n)
```

Der Funktion wird ein Eingangsbild `I` übergeben, der ausgewählte Detektor `pointtype`[8] sowie die Anzahl `n` der maximal zu detektierenden Interest-Points. Der Parameter `sxl2` steuert die Varianz des Gauß-Filters, der zur Berechnung der Ableitungen eingesetzt wird (siehe Gleichung 4.7). Der Wert der Varianz wurde für VisIP als 2.25 gewählt. Für die übrigen Parameter `kparam`, `sxi2` müssen zwar Werte übergeben werden, diese werden aber für den DoH-Detektor nicht benötigt.

Die Funktion `intpointdet` liefert eine Matrix `pos` zurück, in der jedes der detektierten lokalen Features durch eine Zeile mit folgender Form repräsentiert ist:

```
pos = [x y sxl2 c11 c12 c22]
```

Die Position der detektierten lokalen Features wird durch `x, y` beschrieben. Zusätzlich werden die zuvor übergebene Varianz `sxl2` und die Parameter `c11, c12, c22` retourniert, die eine Interest-Region um die lokalen Features beschreiben. Diese vier Werte werden benötigt, falls die detektierten Features als Startpunkte für die ebenfalls in der Sammlung von Ivan Laptev enthaltene Funktion `adaptintpointaffine` verwendet werden. Diese wendet einen iterativen Prozess an, um die charakteristische Skalierung und die affine Form der Features zu bestimmen. In VisIP wird der DoH-Detektor als Interest-Point-Detektor verwendet und nur für eine Skalierung berechnet. Von den zurückgegebenen Werten wird daher lediglich die Position der Features verwendet.

FAST-Detektor: Für den FAST-Detektor wird in VisIP eine Implementierung eingesetzt, die vom Entwickler des Verfahrens selbst stammt. Auf seiner Website stellt Rosten verschiedene Versionen des FAST-Detektors als Maschinen-generierte MATLAB-Funktionen zur Verfügung[9]. Die einzelnen Versionen heißen beispielsweise `fast7`, `fast9` oder `fast12` und unterscheiden sich durch die Länge der Folge v von Bildelementen, die zur Identifizierung eines Eckpunktes betrachtet werden. In VisIP wird die Funktion `fast12` verwendet, die der in Abschnitt 4.1.5 vorgestellten Vorgehensweise entspricht. Ein Eckpunkt wird dabei an Stellen identifiziert, an denen eine Folge von v benachbarten, auf dem Bresenham-Kreis liegenden Bildelementen, wesentlich heller oder dunkler als der aktuell betrachtete Bildpunkt ist.

Der Funktionsaufruf zur Berechnung des FAST-Detektors hat folgende Form:

```
[c, score] = fast12(I, T, nonmax)
```

Es werden das Eingangsbild `I` und ein Schwellwert `T` übergeben, der die Intensitätsdifferenz steuert, die zum Identifizieren eines Eckpunktes betrachtet wird. Durch die Variable `nonmax` kann gesteuert werden, ob eine Nicht-Maxima-Unterdrückung durchgeführt

[8] Für den DoH-Detektor ist `pointtype=3`

[9] http://www.edwardrosten.com/work/fast.html (zuletzt abgerufen am 02.11.2011)

werden soll (`1`) oder nicht (`0`). Bei der Nicht-Maxima-Unterdrückung wird für jeden potentiellen Interest-Point eine kleine Nachbarschaft betrachtet und es werden nur Punkte ausgewählt, die in der betrachteten Nachbarschaft ein Maximum annehmen.

Als Rükgabewert liefert der FAST-Detektor eine Matrix `c` mit den Positionen der identifizierten Eckpunkte (`x, y`). Die Funktion bietet nicht die Möglichkeit, die Anzahl der Features festzulegen die detektiert werden sollen. Dies liegt darin begründet, dass beim FAST-Detektor ursprünglich kein tatsächliches Interest-Maß berechnet wird, das angibt wie ausgeprägt ein identifiziertes lokales Feature ist. Die Entwickler des FAST-Detektors haben daher in neueren Implementierungen einen Ersatz für dieses Interest-Maß geschaffen. Durch Binärsuche wird für jeden Interest-Point der höchste Schwellwert `T` gefunden, für den der jeweilige Punkt noch als Eckpunkt gewertet werden würde. Je höher dieser Schwellwert ist, desto stärker ausgeprägt ist der gefundene Eckpunkt. Der so ermittelte Wert wird ebenfalls als Ausgabe geliefert (`score`). In VisIP wird die Beschränkung der Anzahl der zu identifizierenden Features dadurch ermöglicht dass die detektierten lokalen Features nach ihrem `score` sortiert werden und dann die gewünschte Anzahl von Features ausgewählt wird.

Der FAST-Detektor identifiziert tendenziell sehr viele lokale Features. Um die Anzahl der berechneten Features zu reduzieren ist der FAST-Detektor in VisIP so eingestellt, dass immer Nicht-Maxima-Unterdrückung (`nonmax = 1`) durchgeführt wird. Zudem wurde der eingesetzte Schwellwert als `T = 40` gewählt und ist somit höher als von Rosten vorgeschlagen (`T = 20`).

Lip-Vireo: LIP-Vireo (LIP: Local Interest Point)[10] ist ein von der VIREO-Forschungsgruppe der City University Of Hong Kong entwickelte Software zur Berechnung verschiedener Interest-Point-Detektoren und -Deskriptoren. Diese ist für Windows und weitere Plattformen verfügbar und umfasst effiziente Implementierungen des Harris Laplace-, Hesse Laplace-, Difference Of Gaussian-, Laplacian Of Gaussian- und des MSER-Detektors wie in Abschnitt 4.2 vorgestellt. Von den Entwicklern wird ein Archiv mit der Programmdatei `lip-vireo.exe` zur Verfügung gestellt, die über die Kommandozeile in folgender Form aufgerufen wird:

```
lip-vireo -img path -d name -kpdir path -c file.txt
```

Als Eingabeparameter werden der Name des ausgewählten Detektors (`-d name`) und Pfade zum Eingangsbild (`-img path`) sowie zu einem Verzeichnis für die Ausgabe der Berechnungen (`-kpdir path`) übergeben. Als Bildformate für das Eingangsbild werden die Formate PGM, BMP und JPG akzeptiert. Weiters muss dem Programm der Name einer Konfigurationsdatei bekanntgegeben werden (`-c file.txt`). Über einen Eintrag in der Konfigurationsdatei wird die Anzahl `n` der detektierten lokalen Features gesteuert. Weiters kann über Angaben in der Konfigurationsdatei unter anderem festgelegt werden, welche Informationen zu den detektierten lokalen Features ausgegeben werden sollen. Nähere Informationen können im LIP-Vireo Leitfaden gefunden werden (Fußnote 10).

[10] http://www.cs.cityu.edu.hk/~wzhao2/lip-vireo.htm (zuletzt abgerufen am 27.10.2011)

Als Ausgabe speichert das Programm eine Textdatei mit der Endung .keys in das angegebene Verzeichnis. In der Textdatei sind die Position der lokalen Features (`x, y`), der Wert des zur Berechnung eingesetzten Interest-Maßes (`funcVal`) an der Stelle und Variablen `a b c` enthalten, die eine Interest-Region für jedes lokale Feature beschreiben. Jedes lokale Feature wird im Ausgabefile durch eine Zeile mit folgender Form beschrieben:

```
x y   a b c  funcVal
```

Für die Berechnung der einzelnen Detektoren mittels LIP-Vireo in VisIP wird jeweils ein String erstellt und als Befehl an die Kommandozeile übergeben. Die in VisIP ausgewählte Anzahl der maximal zu detektierenden lokalen Features wird vor jeder Ausführung in das verwendete Konfigurationsdatei geschrieben. Ist die Identifizierung der Features erfolgt, so werden die Informationen aus der entsprechenden Ausgabedatei wieder in MATLAB eingelesen und dort weiterverarbeitet.

Auf der LIP-Vireo Website wird außerdem die MATLAB-Funktion `display_features` bereitgestellt, mit der die detektierten und in MATLAB eingelesenen Features in den Eingangsbildern dargestellt werden können. In VisIP werden die Teile der Funktion verwendet, die zum Darstellen der kreisförmigen oder ellipsenförmigen Interest-Regions, basierend auf den ausgegebenen Variablen `a`, `b`, `c`, dienen. Für die skalierungsinvarianten Detektoren Harris Laplace-, Hesse Laplace-, DoG und LoG ist die ausgegebene Interest-Region kreisförmig, für den affin-invarianten MSER-Detektor ellipsenförmig. Für den Harris-, DoH- und FAST-Detektor wird nur die Position der detektierten lokalen Features als Ergebnis gekennzeichnet.

5.4 Programmaufbau und Bedienung

Ziel bei der Entwicklung von VisIP war es, eine intuitive Benutzeroberfläche zu schaffen, die einen einfachen visuellen Vergleich der von unterschiedlichen Interest-Point-Detektoren identifizierten lokalen Features ermöglicht. Der Aufbau der VisIP Benutzeroberfläche ist in Abbildung 5.1 zu sehen. Auf der linken Seite der Oberfläche kann ein Eingangsbild zur Bearbeitung ausgewählt, verschiedene Einstellungen für die Detektion und Darstellung der Ergebnisse getroffen und schließlich die Berechnung der lokalen Features gestartet werden. Auf der rechten Seite der Benutzeroberfläche wird das Eingangsbild gemeinsam mit den ermittelten lokalen Features dargestellt. Damit die Ergebnisse aller acht integrierten Detektoren gleichzeitig betrachtet werden können gibt es in VisIP acht Anzeigeflächen (Abbildung 5.1 rechts, grau hinterlegt). Die einzelnen Anzeigeflächen sind miteinander verbunden. Vergrößern oder Verschieben des dargestellten Ergebnisbildes in einer der Anzeigeflächen bedingt die Vergrößerung beziehungsweise Verschiebung der Ergebnisbilder in allen anderen Anzeigeflächen im selben Ausmaß.

Alle Dateien, die zur Ausführung von VisIP in MATLAB benötigt werden, sind in einem Archiv (hier: `VisIP`) mit der in Abbildung 5.2 links abgebildeten Struktur enthalten. Im Unterordner `BILDER` ist eine kleine Sammlung von Bildern enthalten, die als Eingabebilder zur Durchführung des Vergleichs verwendet werden können. Bei den Bildern handelt es sich um Grauwertbilder aus einem Malbuch für Kinder (siehe Abbildung 5.3)

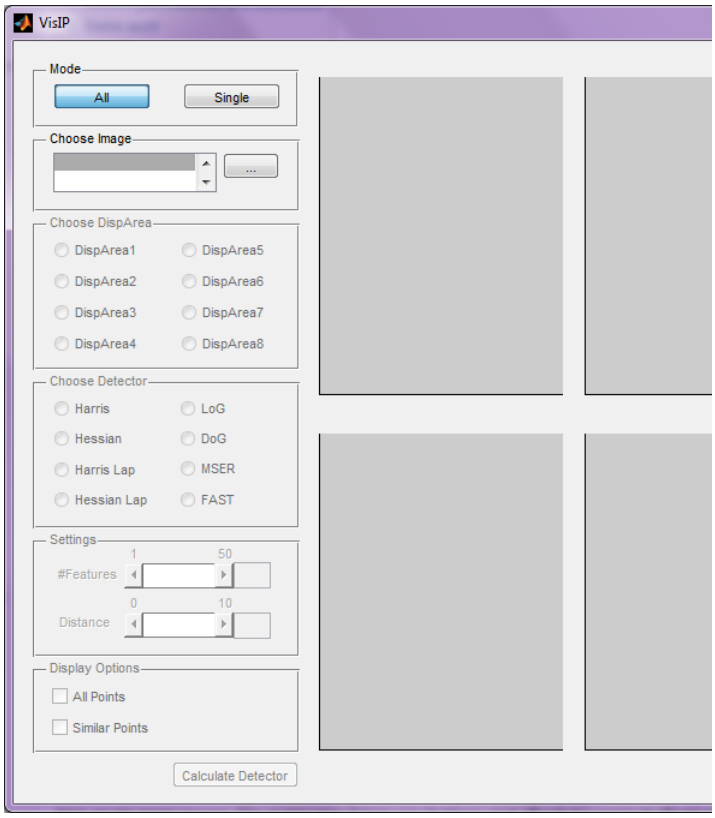


Abbildung 5.1: Benutzeroberfläche von VisIP beim Start des Programms

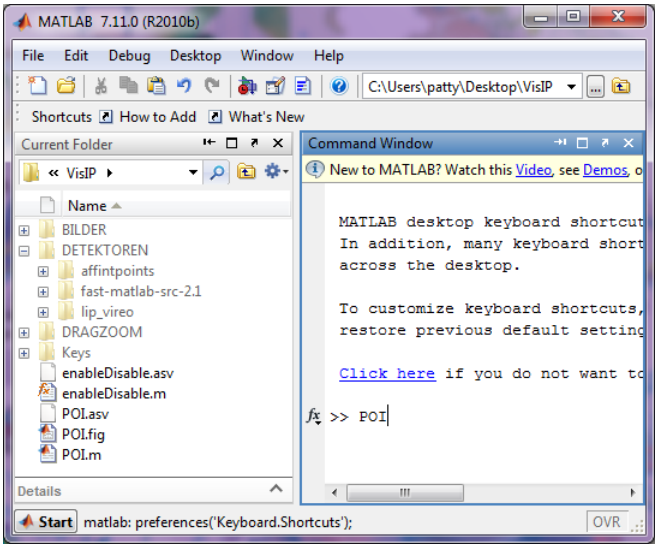


Abbildung 5.2: Verzeichnisstruktur von VisIP (links) und Aufruf von VisIP (rechts) durch Eingabe des Befehls POI in MATLAB

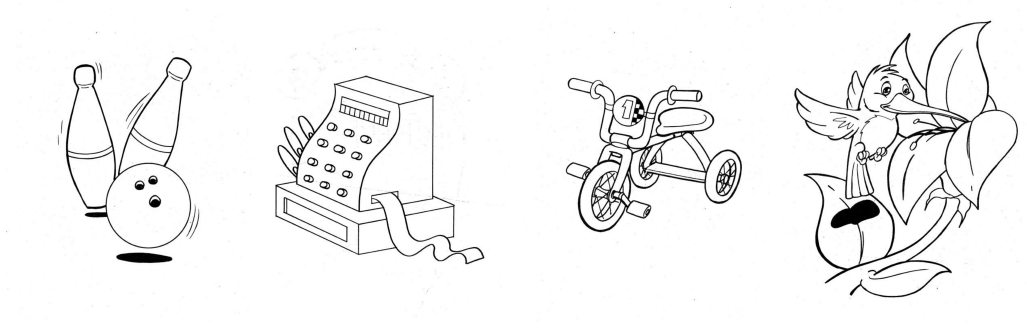


Abbildung 5.3: Beispiele aus der Sammlung von Eingangsbildern für die Durchführung des Vergleichs in VisIP [2]

[2]. Diese wurden aufgrund ihrer Einfachheit ausgewählt, da so gut erkennbar ist, welche Strukturen von welchem Detektor aufgefunden werden und ob ähnliche Strukturen von verschiedenen Detektoren identifiziert werden. Der Ordner `DETEKTOREN` enthält die Implementierungen der jeweiligen Detektoren aus Externen Quellen (LIP-Vireo, Rosten, Laptev). Die mittels LIP-VIREO berechneten lokalen Features werden im Unterverzeichnis `Keys` gespeichert und von dort für die Verwendung in MATLAB eingelesen. Im Ordner DRAGZOOM ist die bereits erwähnte Erweiterung für das Vergrößern in den Anzeigeflächen enthalten.

VisIP wird direkt aus MATLAB durch Eingabe des Befehls `POI` im MATLAB Command Window gestartet, wie in Abbildung 5.2 gezeigt. Es öffnet sich unmittelbar die in Abbildung 5.1 dargestellte Benutzeroberfläche. Um die integrierten Interest-Point-Detektoren auszuführen, müssen zunächst einige Einstellungen auf der linken Seite des Fensters vorgenommen werden. Diese ist in sechs verschiedene Bereiche unterteilt:

Mode: Es kann zwischen zwei verschiedenen Programmmodi in VisIP gewählt werden. Ist der Button `Single` ausgewählt, so können Ergebnisse mehrerer Detektoren nacheinander einzeln berechnet und dargestellt werden (siehe Abbildung 5.4). Es können verschiedene Detektoren oder auch ein und derselbe Detektor mit unterschiedlich gewählten Parametern miteinander verglichen werden. Beim Modus `All` wird für jeden der implementierten Detektoren eine Ergebnismenge erzeugt und dann nebeneinander dargestellt (siehe Abbildung 5.5). Bei einer Ausführung von VisIP im Modus `All` gelten für alle Detektoren dieselben Einstellungen. Standardmäßig ist der Modus `All` ausgewählt.

Choose Image: Durch einen Klick auf den Button mit der Beschriftung `'...'` im Bereich `Choose Image` wird ein Dialogfenster geöffnet, aus dem zunächst ein Eingangsbild aus den mitgelieferten Testbildern ausgewählt werden muss, in welchem die lokalen Features berechnet werden sollen.

Choose DispArea: Wenn als Modus `Single` ausgewählt ist, kann im Bereich `Choose DispArea` eine der acht Anzeigeflächen ausgewählt werden, auf der die Ergebnisse

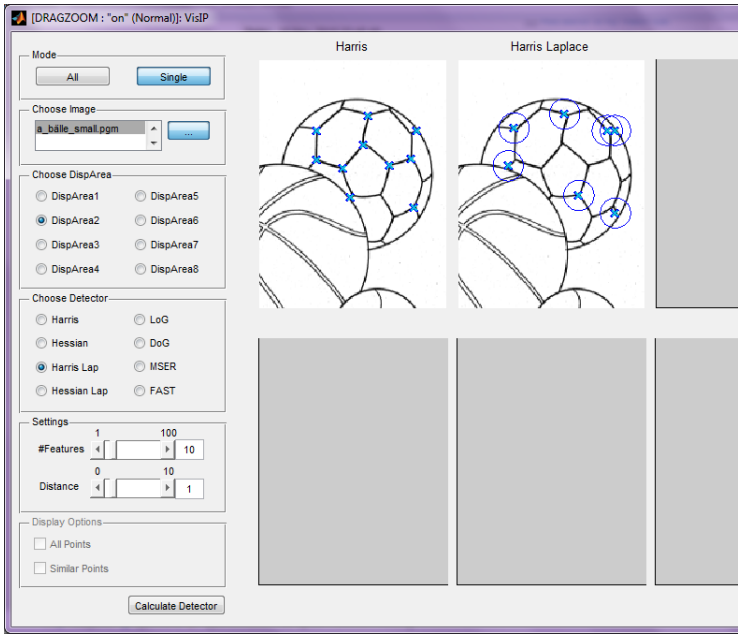


Abbildung 5.4: Beispiel für eine Ausführung von VisIP im Modus `Single`. Der Harris und der Harris Laplace-Detektor werden mit jeweils 10 detektierten lokalen Features nebeneinander dargestellt

dargestellt werden sollen. Die Anzeigeflächen sind von links oben nach rechts unten nummeriert. Im Modus `All` ist dieser Bereich nicht relevant und daher nicht editierbar.

Choose Detector: Im Bereich `Choose Detector` kann einer der acht Detektoren zur Berechnung ausgewählt werden. Die Auswahl ist wiederum nur im Modus `Single` zugänglich. Im Modus `All` werden alle acht Detektoren berechnet und in jeweils einer der Anzeigeflächen dargestellt.

Settings: Im Bereich `Settings` sind zwei Schieberegler platziert. Durch den Schieberegler `#Features` kann die maximale Anzahl der lokalen Features begrenzt werden, die bei der Ausführung zurückgeliefert werden sollen. In VisIP können ähnliche lokale Features farblich hervorgehoben dargestellt werden (siehe Abbildung 5.5). Ähnlich bedeutet dabei, dass ein Interest-Point von unterschiedlichen Detektoren an einer ähnlichen Position im Eingangsbild aufgefunden wurde. Über den Regler `Distance` kann dabei gesteuert werden, wie groß der Abstand zwischen zwei detektierten Features maximal sein darf damit sie als ähnlich gelten. Der Abstand wird mit der Euklidischen Distanz gemessen (siehe [69]).

Display Options: Wenn die Einstellungen für alle zuvor genannten Bereiche getroffen wurden kann die Ausführung eines oder aller Detektoren durch einen Klick auf den Button `Calculate Detector` gestartet werden. Die Ergebnisse werden in den jewei-

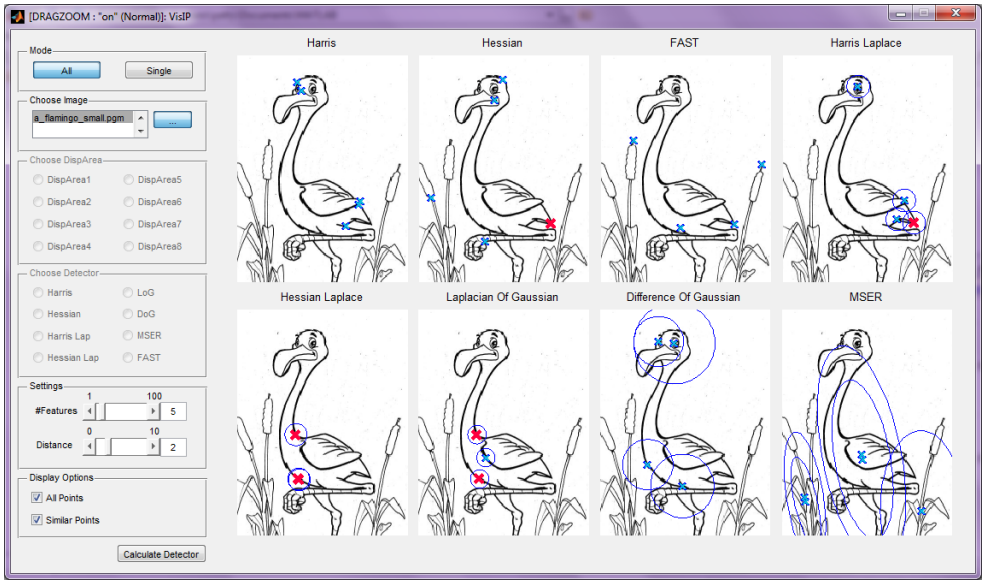


Abbildung 5.5: Beispiel für eine Ausführung von VisIP im Modus `All`. Es werden alle acht verfügbaren Detektoren auf einem Eingangsbild berechnet und die Ergebnisse dargestellt. Positionen, an denen von mehreren verschiedenen Detektoren Interest-Points gefunden wurden, können rot hervorgehoben dargestellt werden, wenn die Option `Similar Points` gesetzt ist

ligen Anzeigeflächen angezeigt. Dabei wird immer das gewählte Eingangsbild dargestellt und darüber die lokalen Features. Für die Interest-Point-Detektoren Harris, DoH und FAST wird dabei nur die Position x,y des lokalen Features als Punkt dargestellt. Für die skalierungsinvarianten Features wird zusätzlich eine kreisförmige Interest-Region dargestellt, für den affin-invarianten MSER-Detektor wird die Interest-Region durch eine Ellipse repräsentiert. Im Bereich `Display Options` kann durch Setzen der entsprechenden Checkboxen festgelegt werden, ob alle (`All Points`) oder nur die als ähnlich eingestuften lokalen Features (`Similar Points`) angezeigt werden sollen.

Wie bereits erwähnt können durch die Funktion DRAGZOOM die dargestellten Ergebnisse vergrößert werden oder die Bilder so verschoben werden, dass die relevanten Bereiche besser sichtbar sind. Dies kann durch folgende Mausbewegungen über einer der Anzeigeflächen durchgeführt werden:

```
Linke Maustaste drücken   +  Maus bewegen            =>  Anzeige verschieben
Mausrad drücken           +  Maus nach oben bewegen  =>  Anzeige vergrößern
Mausrad drücken           +  Maus nach oben bewegen  =>  Anzeige verkleinern
Doppelklick mit linker oder rechter Maustaste        =>  zur Ausgangslage zurück
```

6 Zusammenfassung und Schlussbetrachtungen

Ziel der vorliegenden Arbeit war es einerseits, einen gute Basis für den Einstieg in das Gebiet der Interest-Point-Detektion zu bieten und andererseits die Visualisierung und den Vergleich von Ergebnissen gängiger Interest-Point-Detektoren zu ermöglichen. Wir möchten im Folgenden die Inhalte des theoretischen Teils der Arbeit sowie gewonnene Erkenntnisse noch einmal kurz zusammenfassen (Abschnitt 6.1). Der praktische Teil der Arbeit bestand in der Realisierung von VisIP. In Abschnitt 6.2 betrachten wir Rahmenbedingungen bei der Entwicklung der Applikation und mögliche Anwendungen.

6.1 Die Theorie

Ein großer Teil der vorliegenden Arbeit beschäftigte sich mit den theoretischen Grundlagen, die für das Verständnis der Interest-Point-Detektoren nötig oder zumindest hilfreich sind. Interest-Point-Detektoren werden heute für eine Vielzahl von Anwendungen eingesetzt, wie beispielsweise für die Objekterkennung oder die Objektverfolgung. Für die meisten dieser Anwendungen ist die Durchführung eines Bildabgleichs in einem ersten Schritt vonnöten. Interest-Point-Detektoren identifizieren bestimmte lokale Features in Bildern welche sich als besonders geeignet zur Verwirklichung eines robusten Bildabgleichs erwiesen haben. Die Vorteile des Einsatzes von Verfahren, die auf Interest-Point-Detektion basieren, wurden in Kapitel 1 ausgeführt. Diese liegen beispielsweise in der kompakten Repräsentation des Bildinhalts, die durch den Einsatz von Interest-Points realisiert wird und die Invarianz dieser Verfahren gegenüber verschiedenen geometrischen und photometrischen Bildtransformationen.

Wir haben eine kleine Auswahl an möglichen Anwendungen für Interest-Points beschrieben. Dazu gehören beispielsweise das Stitching, das zum Erstellen von Panoramabildern eingesetzt wird und die inhaltsbasierte Bildsuche, die das Durchsuchen großer Bilddatenbanken basierend auf den Bildinhalten ermöglicht. Viele weitere Anwendungsszenarien werden in den Arbeiten von Maggio u.a. [40] und Szeliski [72] vorgestellt.

In Kapitel 2 haben wir den Begriff des Interest-Points näher untersucht. Wir haben festgestellt, dass in der Literatur viele unterschiedliche Begriffe für Interest-Points verwendet werden, wie beispielsweise saliente Punkte, Schlüsselpunkte oder lokale Features. Auch existieren verschiedene Definitionen für das Konzept des Interest-Points. Dabei werden den beschriebenen Punkten aber immer ähnliche Eigenschaften zugeschrieben. So sind Interest-Points zumeist als Stellen im Bild definiert, die eine wohldefinierte Position im Bild und einen hohen Informationsgehalt aufweisen. Zudem wird gewünscht, dass

sie robust beziehungsweise sogar invariant gegenüber verschiedenen Bildtransformationen wie beispielsweise Rotationen, uniformen Skalierungen und affinen Transformationen sind [73].

Damit die Funktionsweise von Interest-Point-Detektoren verstanden werden kann ist es nötig, die grundlegenden Bildverarbeitungsprozesse zu kennen, auf denen diese aufbauen. Viele Verfahren zur Detektion von Interest-Points basieren auf lokalen Operatoren, die auch Filter genannt werden. Diese Filter haben wir in Kapitel 2 ausführlich beschrieben. Interest-Point-Detektoren identifizieren lokale Features häufig an Stellen, an denen abrupte Intensitätsänderungen in mehreren Richtungen auftreten. Lineare Differenzfilter heben solche Intensitätsänderungen in Bildern hervor und können daher zum Auffinden der gesuchten Stellen eingesetzt werden. Lineare Glättungsfilter hingegen lassen Strukturen in Bildern unscharf erscheinen. Eine besonders wichtige Rolle spielen Gaußsche Glättungsfilter, deren Koeffizienten auf Basis der Gauß-Funktion berechnet werden und somit einige der besonderen Eigenschaften der Gauß-Funktion teilen. Eine ausführliche Beschreibung der Gauß-Funktion und ihren Eigenschaften ist in [58] zu finden. Durch Kombination von Differenzfiltern und Glättungsfiltern kann die Anfälligkeit von Differenzfiltern auf Bildrauschen ausgeglichen werden. Lineare Glättungsoperatoren bilden außerdem die Grundlage zum Aufbau von Skalenräumen. Skalenräume sind Repräsentationen eines Bildes und dessen Eigenschaften auf verschiedenen Auflösungsstufen in einem dreidimensionalen Raum. Sie bilden häufig die Basis skalierungsinvarianter oder affin-invarianter Interest-Point-Detektoren.

Um einen passenden Interest-Point-Detektor für eine bestimmte Aufgabenstellung auswählen zu können, ist es wichtig einen guten Überblick darüber zu haben, welche Techniken prinzipiell existieren und angewendet werden können. Aus diesem Grund haben wir in Kapitel 3 eine kurze Zusammenfassung der verschiedenen Verfahren zur Detektion von Interest-Points in der Literatur gegeben. Dabei wurde klar, dass in der Praxis vor allem intensitätsbasierte Methoden eingesetzt werden, welche Informationen über lokale Features direkt aus den Intensitätswerten der betrachteten Grauwertbilder extrahieren. Aus diesem Grund wurde der Schwerpunkt der Betrachtungen in weiterer Folge auf intensitätsbasierte Verfahren gelegt. Bei der Ausarbeitung des Literaturüberblicks wurde klar, dass bei intensitätsbasierten Verfahren zumeist grob zwischen zwei Gruppen von Detektoren unterschieden wird, nämlich zwischen Eckpunkt-Detektoren und Blob-Detektoren. Die Unterscheidung erfolgt dabei in Abhängigkeit von der Struktur der lokalen Features, die vom jeweiligen Detektor hauptsächlich identifiziert werden. In beiden Detektor-Gruppen kann man eine deutliche Entwicklung erkennen die von einfachen Verfahren, welche robust gegenüber Translationen und eventuell Rotationen sind, über skalierungsinvariante Methoden bis zu Verfahren geht, die invariant gegenüber affinen Transformationen sind.

Wir haben in Kapitel 3 zudem einen Überblick zu Arbeiten gegeben, die sich mit der Evaluierung von Interest-Point-Detektoren auseinandersetzen. In diesen Arbeiten wurden vor allem Effizienz oder Robustheit verschiedener Verfahren gegenübergestellt.

In Kapitel 4 haben wir die Vorgehensweise bei der Berechnung der gängigen Interest-Point-Detektoren ausführlich beschrieben. Es wurden dabei diejenigen Detektoren ausgewählt, die in der Praxis häufig eingesetzt werden oder die einen wichtigen Meilenstein in der Entwicklung von Interest-Point-Detektoren repräsentieren. Die ausgewählten Detektoren sind der Moravec-, Harris-, Determinant Of Hessian-, SUSAN-, FAST-, Harris Laplace-, Hesse Laplace-, Laplacian Of Gaussian-, Difference Of Gaussian- und den MSER-Detektor. Bei der Beschreibung der Detektoren wurde zwischen Detektoren unterschieden, die lediglich die Position eines lokalen Features identifizieren (Interest-Point-Detektoren) und Detektoren, die zudem die charakteristische Skalierung oder affine Form des lokalen Features an der Position schätzen (Interest-Region-Detektoren). Wir haben festgestellt, dass für die Identifizierung von Eckpunkten häufig Kombinationen von Ableitungen erster Ordnung aus der Autokorrelationsmatrix eingesetzt werden. Für das Auffinden blob-ähnlicher Strukturen spielen andererseits Kombinationen von Ableitungen zweiter Ordnung, berechnet aus der Hesse-Matrix, eine wichtige Rolle.

6.2 Die Praxis - VisIP

Der praktische Teil der vorliegenden Arbeit bestand in der Realisierung einer Applikation für den Vergleich von Interest-Point-Detektoren. Wie zuvor erwähnt, wurden in der Vergangenheit bereits zahlreiche Vergleiche verschiedener Detektoren durchgeführt (Abschnitt 3.2). Der Schwerpunkt bei aktuelleren Vergleichen lag dabei zumeist auf der Untersuchung der Robustheit einzelner Verfahren gegenüber unterschiedlichen Bildtransformationen auf Basis der Wiederholungsrate.

Wenn wir unsere Umgebung betrachten so werden sofort, auf unbewusster Ebene, wichtige Bereiche in unserem Sichtfeld detektiert [27, 70, 54]. Dabei stuft unser Wahrnehmungssystem Regionen als wichtig ein, die einen besonders hohen Informationsgehalt tragen und daher wichtig für eine rasche Entscheidungsfindung sind. Diese Regionen entsprechen lokalen Strukturen, wie beispielsweise Eckpunkten, Blobs oder Kanten, die mit Diskontinuitäten[1] einhergehen [27, 70]. Die Grundidee zur Interest-Point-Detektion war es, die Mechanismen auf denen der Sehprozess bei Menschen basiert nachzubilden. Interest-Point-Detektoren wurden entwickelt, um computergestützt solche wichtigen lokalen Strukturen in Bildern zu identifizieren. Wie beim menschlichen Vorbild sollte es dadurch möglich sein, gleich zu Beginn des Verarbeitungsprozesses von digitalen Bildern den Fokus auf wichtige Regionen im Bild zu richten. Aus diesen sollten einerseits besonders aussagekräftige Informationen zur Weiterverarbeitung und andererseits sehr kompakte Repräsentationen des Bildinhalts gewonnen werden können (Abschnitt 3) [75, 24].

Ein großer Teil unserer Interaktion mit der Umwelt und der Entscheidungen die wir treffen, basiert auf der Analyse visueller Informationen [11]. Unser visuelles Wahrnehmungssystem vermag außergewöhnliche Leistungen zu vollbringen. Wir sind beispielsweise in der Lage, das eigene Fahrrad in einem Meer von Fahrrädern auf einem Fahrradabstellplatz wieder zu finden. Zudem können wir, wenn wir einmal gelernt haben wie

[1] d.h. mit irgendwelchen Änderungen, beispielsweise der Materialeigenschaften oder Helligkeiten [27]

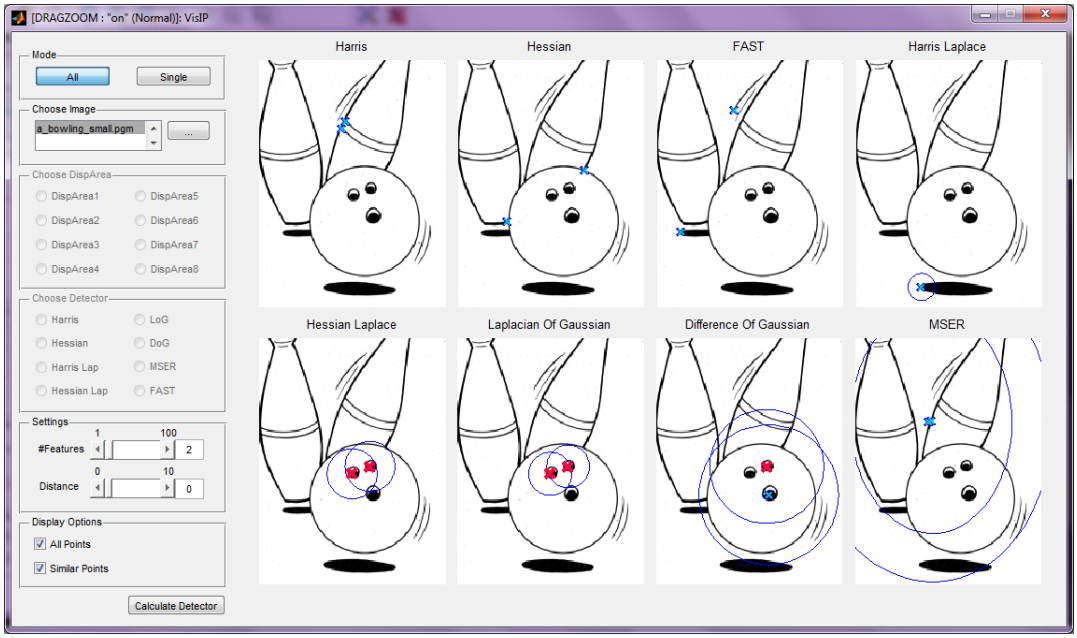


Abbildung 6.1: Beispielausführung von VisIP; Für den Hesse-Laplace, LoG- und DoG-Detektor wurden Interest-Points an ähnlichen Positionen identifiziert und sind rot hervorgehoben

ein Fahrrad aussieht, jedes Fahrrad auch als solches identifizieren. Dabei ist es ganz egal, welche Farbe es hat oder aus welcher Richtung wir es gerade ansehen [73]. Wir können also problemlos die Objekte in unserer Umgebung vergleichen, erkennen und auch kategorisieren und somit Aufgaben visuell lösen, an denen moderne Computersysteme noch immer scheitern (siehe Abschnitt 3). Die im Rahmen der vorliegenden Arbeit entwickelte Applikation basiert auf der Idee, diese Überlegenheit des menschlichen visuellen Wahrnehmungssystems gegenüber Computersystemen für den Vergleich und zur Beurteilung der Ergebnisse von Interest-Point-Detektoren auszunutzen.

Die Applikation trägt den Namen VisIP und ermöglicht den visuellen Vergleich von Interest-Point-Detektoren. In VisIP sind der Harris-, Determinant Of Hessian-, FAST-, Harris Laplace-, Hesse Laplace-, Laplacian Of Gaussian-, Difference Of Gaussian- und der MSER-Detektor integriert. VisIP wurde primär für den Einsatz zu Demonstrationszwecken und als visuelle Unterstützung für Lehrveranstaltungen entwickelt. Die ausgewählten Detektoren sollten zu diesem Zweck auf sehr einfach strukturierte Bilder angewendet und die Ergebnisse schließlich nebeneinander dargestellt werden können. Dadurch sollte die Möglichkeit geschaffen werden zu zeigen, auf welchen Bildstrukturen die jeweiligen Detektoren Interest-Points identifizieren. Wir können dies nun mit Hilfe von VisIP beispielsweise untersuchen, indem wir uns nur die am stärksten ausgeprägten Features anzeigen lassen. In VisIP kann für alle Detektoren die maximale Anzahl der lokalen Features festgelegt werden die zurückgeliefert werden sollen. Die identifizierten Features werden dabei immer in absteigender Reihenfolge, sortiert nach dem errechneten Maß für die Stärke der

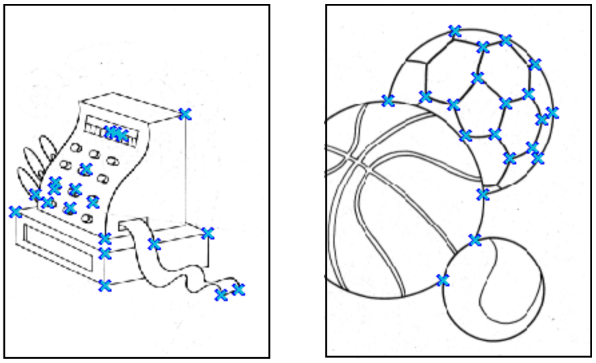


Abbildung 6.2: Ergebnis der Detektion des Determinant Of Hessian-Detektors auf zwei unterschiedlichen Bildern; die auf nur einer Auflösungsstufe des Eingangsbildes ermittelten Interest-Points entsprechen dabei eher eckpunkt- als blob-ähnlichen Strukturen

vorliegenden Struktur ausgegeben.

Weiters sollte in VisIP die Option geboten sein, Gemeinsamkeiten und Unterschiede in der Menge der lokalisierten Features zu untersuchen. In VisIP können für alle integrierten Detektoren auf einmal Ergebnisse ermittelt und dargestellt werden. Wir können uns dabei lokale Features die von mehreren Detektoren an einer ähnlichen Stelle identifiziert wurden farblich hervorgehoben darstellen lassen. Ein Beispiel für eine entsprechende Ausführung von VisIP wird in Abbildung 6.1 gezeigt, wobei hier für den Hesse Laplace-, den LoG- und den DoG-Detektor Interest-Points an ähnlichen Positionen gefunden wurden.

Interest-Point-Detektoren sind für den Einsatz in einer Vielzahl von Anwendungen geeignet und erzielen gute Resultate (Abschnitt 3.2). Durch den Einsatz von VisIP kann man aber gut erkennen, dass die in der Praxis identifizierten Strukturen häufig nicht den Strukturen entsprechen, die man aufgrund der theoretischen Konzepte hinter den Detektoren erwarten würde. So wird beispielsweise der in Abschnitt 4.1.3 vorgestellte Determinant Of Hessian-Detektor häufig als Blob-Detektor kategorisiert (siehe [75]). Wenn der Detektor jedoch nur auf einer Auflösungsstufe des Eingangsbildes berechnet wird werden für gewöhnlich sehr kleine Filter eingesetzt (siehe Gleichung 2.20). Aufgrund der Form dieser Filter werden in der Praxis eher eckpunkt-ähnliche Strukturen identifiziert und keine Blobs (siehe Abbildung 6.2).

In Kapitel 5 haben wir beschrieben, wie bei der Umsetzung von VisIP vorgegangen wurde, wie die Bedienung der Applikation erfolgt und auch welche Faktoren für die Auswahl von geeigneten Implementierungen der Detektoren berücksichtigt wurden. Eine eigenständige Implementierung der Detektoren war bei der Entwicklung von VisIP nicht vorgesehen. Einerseits gibt es bereits eine ausreichende Anzahl verschiedener Implementierungen von Interest-Point-Detektoren, die von den Entwicklern zur Verfügung gestellt werden und andererseits wäre damit ein erheblicher Mehraufwand verbunden gewesen, der den Rahmen der Diplomarbeit gesprengt hätte.

Für mögliche Weiterentwicklungen von VisIP wäre es dennoch von Vorteil, eine selbständige Implementierung der Verfahren vorzunehmen. Häufig wird nämlich bei den

verfügbaren Implementierungen ein ausführbares Programm, nicht aber der zugehörige Quellcode und nur spärliche Dokumentation bereitgestellt. Es gestaltete sich daher schwierig, geeignete Implementierungen für VisIP zu finden. Es kann zudem nicht genau nachvollzogen werden, wie die Ergebnisse der Berechnung entstehen. Dieses Problem wäre bei einer selbständigen Implementierung nicht gegeben. Zudem könnte ein umfassenderer Vergleich der Detektoren durchgeführt werden, indem mehrere variable Einstellungsmöglichkeiten geboten werden, beispielsweise um die Beschaffenheit der eingesetzten Filter festzulegen. Für den Einsatz zu Demonstrationszwecken wäre es zudem hilfreich, wenn wichtige Zwischenschritte des Detektionsprozesses dargestellt werden könnten.

Fast alle der verwendeten Implementierungen können als Ausgabe ein Maß für die Wahrscheinlichkeit des Vorliegens der gesuchten lokalen Struktur liefern. Zur besseren Beurteilung der Ergebnisse der einzelnen Interest-Point-Detektoren könnte eine alternative Visualisierung geboten werden, in der die gefundenen Interest-Points in Abhängigkeit von diesem berechneten Maß in unterschiedlichen Größen dargestellt werden.

Da die Invarianz von Interest-Point-Detektoren gegenüber verschiedenen geometrischen Transformationen ein wichtiges Kriterium ist, vor allem bei Anwendungen wie der inhaltsbasierten Bildsuche, könnte man VisIP dahingehend erweitern, dass auch ähnliche Interest-Points in unterschiedlich transformierten Versionen eines bestimmten Eingangsbildes erkannt werden können.

VisIP ist eine Applikation, die in der Lage ist, durch den Einsatz verschiedener Interest-Point-Detektoren “interessante” Strukturen in Bildern zu detektieren. Anhand der Beschaffenheit der identifizierten Strukturen können nun die tatsächlichen Leistungen von Interest-Point-Detektoren nachvollzogen werden.

Literaturverzeichnis

[1] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193, May 1954.

[2] Bambino. Das lustige malheft 6. Martin Kelter Verlag GmbH & Co. KG.

[3] P. R. Beaudet. Rotationally invariant image operators. In *Proceedings of the 4th International Joint Conference on Pattern Recognition*, pages 579–583, Kyoto, Japan, November 1978.

[4] J. Bernal, F. Vilariño, and J. Sánchez. Feature detectors and feature descriptors: Where we are now. Technical report, Computer Vision Center and Computer Science Department UAB Campus UAB, Edifici O, 08193, Bellaterra, Barcelona, Spain, 2010.

[5] S. Beucher and C. Lantuejoul. Use of Watersheds in Contour Detection. In *International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation, Rennes, France.*, September 1979.

[6] I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

[7] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik, 25.* 1991.

[8] M. Brown and D. G. Lowe. Recognising panoramas. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, volume 2 of *ICCV '03*, pages 1218–1225, Washington, DC, USA, 2003. IEEE Computer Society.

[9] H. Bässmann and J. Kreyß. *Bildverarbeitung Ad Oculos.* Springer, 4., aktualisierte auflage edition, 2004.

[10] W. Burger and M. J. Burge. *Digitale Bildverarbeitung.* X. media. press Series. Springer, 2006.

[11] L. F. Costa and R. M. Cesar. *Shape analysis and classification: theory and practice.* Image processing series. CRC Press, 2001.

[12] J. L. Crowley and A. C. Parker. Representation for shape based on peaks and ridges in the difference of low-pass transform. Technical Report CMU-RI-TR-83-04, Robotics Institute, Pittsburgh, PA, May 1983.

[13] I. Dimitrovski and S. Loskovska. Content-based retrieval system for x-ray images. In *2nd International Congress on Image and Signal Processing, CISP '09*, pages 1–5, 2009.

[14] Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *In Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South*, pages 612–618, 2000.

[15] A. Erhardt. *Einführung in die digitale Bildverarbeitung: Grundlagen, Systeme und Anwendungen ; mit 35 Beispielen und 44 Aufgaben.* Vieweg + Teubner, 2008.

[16] W. Forstner. A feature based correspondence algorithm for image matching. pages III: 150–166, 1986.

[17] F. Fraundorfer and H. Bischof. Evaluation of local detectors on non-planar scenes. In *In Proc. 28th workshop of the Austrian Association for Pattern Recognition*, pages 125–132, 2004.

[18] V. Gouet and N. Boujemaa. About optimal use of color points of interest for content-based image retrieval. Rapport de recherche RR-4439, Institut National de Recherche en Informatique et en Automatique, INRIA '02, 2002.

[19] K. Grauman and B. Leibe. *Visual Object Recognition.* Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2011.

[20] C. Harris and M. Stephens. A combined corner and edge detection. *In Proceedings of The Fourth Alvey Vision Conference*, 1988. Harris Operator.

[21] D. Hearn and M. P. Baker. *Computer Graphics with OpenGL.* Prentice Hall Professional Technical Reference, 3 edition, 2003.

[22] T. Hermes. *Digitale Bildverarbeitung - Eine praktische Einführung.* Hanser Verlag, 2004.

[23] R. Horaud, F. Veillon, and T. Skordas. Finding geometric and relational structures in an image. In *Proceedings of the First European Conference on Computer Vision*, ECCV '90, pages 374–384, London, UK, 1990. Springer-Verlag.

[24] A. Jacobs. *Ein deformationsinvarianter Point-of-Interest-Detektor.* PhD thesis, Universität Bremen, Fachbereich 3 (Mathematik und Informatik), Mai 2010.

[25] B. Jähne. *Digitale Bildverarbeitung.* Springer, Berlin, 7. edition, 2010.

[26] F. Jung. Objekterkennung mit sift-features, September 2006.

[27] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45:83–105, 2001. 10.1023/A:1012460413855.

[28] T. Kadir, A. Zisserman, and J. M. Brady. An affine invariant salient region detector. In *European Conference on Computer Vision.* Springer-Verlag, 2004.

[29] V. Kangas. Comparison of local feature detectors and descriptors for visual object categorization. Master's thesis, Lappeenranta University of Technology, 2011.

[30] D. T. Kien. A review of 3d reconstruction from video sequences. Technical report, Faculty of Science, University of Amsterdam, 2005.

[31] J. Koenderink and A. J. Van Doorn. The structure of locally orderless images, 1998.

[32] T. Lindeberg. Scale-space for discrete signals. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12:234–254, March 1990.

[33] T. Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *International Journal of Computer Vision*, 11:283–318, 1993. 10.1007/BF01469346.

[34] T. Lindeberg. On scale selection for differential operators. *8TH SCIA*, 1993.

[35] T. Lindeberg. *Scale-Space Theory in Computer Vision*. 1994.

[36] T. Lindeberg. Direct estimation of affine image deformations using visual front-end operations with automatic scale selection. In *In Proc. 5th International Conference on Computer Vision*, pages 134–141. IEEE Computer Society Press, 1995.

[37] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30:79–116, 1998.

[38] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. Seventh IEEE Int Computer Vision Conf. The*, volume 2, pages 1150–1157, 1999.

[39] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60:91–110, November 2004.

[40] E. Maggio and A. Cavallaro. *Video tracking: theory and practice*. Wiley, 2011.

[41] O. Marques and F. Borivoje. *Content-Based Image and Video Retrieval*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[42] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *In British Machine Vision Conference*, pages 384–393, 2002.

[43] K. Mikolajczyk. *Detection of local features invariant to affine transformations*. PhD thesis, INPG, Grenoble, juillet 2002.

[44] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *In Proceedings of the 8th International Conference on Computer Vision*, pages 525–531, 2001.

[45] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *In Proceedings of the 7th European Conference on Computer Vision*, pages 0–7, 2002.

[46] K. Mikolajczyk and C. Schmid. Comparison of affine-invariant local detectors and descriptors. In *12th European Signal Processing Conference, Autstria*, 2004.

[47] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[48] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1615–1630, 2005.

[49] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65:2005, 2005.

[50] P. Montesinos, V. Gouet, F-Nimes Cedex, and R. Deriche. Differential invariants for color images, 1998.

[51] H. P. Moravec. Towards automatic visual obstacle avoidance. In *IJCAI*, page 584, 1977.

[52] H. P. Moravec. Obstacle avoidance and navigation in the real world by a seeing robot rover. In *tech. report CMU-RI-TR-80-03, Robotics Institute, Carnegie Mellon University & doctoral dissertation, Stanford University*, number CMU-RI-TR-80-03. September 1980.

[53] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d objects. *Int. J. Comput. Vision*, 73:263–284, July 2007.

[54] U. Neisser. *Visual search*. Scientific American offprints. W.H. Freeman, 1964.

[55] J. A. Noble. Finding corners. *Image Vision Comput.*, 6:121–128, May 1988.

[56] A. Reiterer and T. Eiter. A distance-based method for the evaluation of interest point detection algorithms. In *Proc. IEEE Int Image Processing Conf*, pages 2745–2748, Februar 2007.

[57] K. Rohr. Recognizing corners by fitting parametric models. *Int. J. Comput. Vision*, 9:213–230, December 1992.

[58] B. M. Romeny. *Front-End Vision and Multi-Scale Image Analysis: Multi-scale Computer Vision Theory and Applications, written in Mathematica*. Springer Publishing Company, Incorporated, 1st edition, 2009.

[59] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511, October 2005.

[60] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006.

[61] A. Schilham, B. Van Ginneken, and M. Loog. Multi-scale nodule detection in chest radiographs. In Randy Ellis and Terry Peters, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003*, volume 2878 of *Lecture Notes in Computer Science*, pages 602–609. Springer Berlin / Heidelberg, 2003.

[62] M. Schlattmann. Bestimmung skalenbehafteter merkmalspunkte auf zweimannigfaltigen, triangulierten oberflächen. Diplomarbeit, Institut für Informatik II, Universität Bonn, 2005.

[63] C. Schmid, R. Mohr, and C. Bauckhage. Comparing and evaluating interest points. In *Proc. Sixth Int Computer Vision Conf*, pages 230–235, 1998.

[64] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Int. J. Comput. Vision*, 37:151–172, June 2000.

[65] N. Sebe, Q. Tian, E. Loupias, M. Lew, and T. Huang. Evaluation of salient point techniques, 2002.

[66] L. G. Shapiro and G. C. Stockman. *Computer Vision*. Prentice Hall, January 2001.

[67] J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pages 593 –600, jun 1994.

[68] S. M. Smith and J. M. Brady. Susan a new approach to low level image processing. *International Journal of Computer Vision*, 23:45–78, 1997. 10.1023/A:1007963824710 SUSAN.

[69] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Brooks/Cole, 2 edition, 1999.

[70] J. Stöttinger. Local colour features for image retrieval. Master's thesis, Technische Universität Wien, Institut für Rechnergestütze Automation, Arbeitsgruppe für Mustererkennung und Bildverarbeitung, April 2007.

[71] J. Stöttinger. Detection and evaluation methods for local image and video features. Technical Report CVL-TR-4, Computer Vision Lab, Institute of Computer Aided Automation, Vienna University of Technology, March 2011.

[72] R. Szeliski. Computer vision: Algorithms and applications. *Computer*, september 2010.

[73] A. Teynor. *Visual object class recognition using local descriptions*. PhD thesis, Albert-Ludwigs-Universität Freiburg im Breisgau; Fakultät: Technische Fakultät (bisher: Fak. f. Angew. Wiss.); Institut: Institut für Informatik, August 2008.

[74] T. Tuytelaars and L. Gool. Wide baseline stereo matching based on local, affinely invariant regions. In *In Proc. BMVC*, pages 412–425, 2000.

[75] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *FnT Comp. Graphics and Vision*, pages 177–280, 2008.

[76] J. Van de Weijer, T. Gevers, and A. D. Bagdanov. Boosting color saliency in image feature detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28:150–156, January 2006.

[77] A. P. Witkin. Scale-space filtering. In *Proceedings of the Eighth international joint conference on Artificial intelligence - Volume 2*, pages 1019–1022, San Francisco, CA, USA, 1983. Morgan Kaufmann Publishers Inc.