



UNIVERSITÄT ZU LÜBECK
INSTITUTE OF MATHEMATICS AND
IMAGE COMPUTING

Der Quadraturfilter und dessen Anwendung im Morphon-Algorithmus

*The Quadrature Filter and its application
in the Morphon algorithm*

Bachelorarbeit

im Rahmen des Studiengangs
Mathematik in Medizin und Lebenswissenschaften
der Universität zu Lübeck

vorgelegt von

Patricia Galuschka

ausgegeben und betreut von

Prof. Dr. Jan Modersitzki

Institute of Mathematics and Image Computing

Lübeck, den 25. November 2015

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Arbeit unterstützt haben.

Ganz besonders gilt dieser Dank Jan Modersitzki für die Ausgabe und die Betreuung dieser Arbeit. Das stetig kritische Hinterfragen und die konstruktive Kritik haben einen großen Teil zu dieser Arbeit beigetragen. Vielen Dank für die Zeit und die Geduld.

Danken möchte ich Lennart Tautz für die guten Gespräche, die viel Licht in das Thema gebracht haben.

Ein großes Dankeschön gilt Nadine Traulsen. Danke, dass ich immer mit Fragen und Problemen zu Dir kommen durfte und Du mich immer wieder ermutigt hast.

Außerdem möchte ich mich bei meiner Familie und meinen Freunden bedanken, nicht nur für das Korrekturlesen, sondern auch für die Unterstützung in allen Lebenslagen.

Eidesstattliche Erklärung

Ich versichere an Eides statt, die vorliegende Arbeit selbstständig und nur unter Benutzung der angegebenen Hilfsmittel angefertigt zu haben.

Lübeck, den 25. November 2015

Kurzfassung

Diese Arbeit beschäftigt sich mit einem Algorithmus der Bildregistrierung. Die Bildregistrierung ist ein Aufgabengebiet der Bildverarbeitung, die beispielsweise in der Medizin weitreichende Anwendung findet. Der Algorithmus, der hier vorgestellt wird, ist der Morphon-Algorithmus, dessen Grundlage ein Quadraturfilter ist. Zum Verständnis werden die benötigten mathematischen Grundlagen, wie auch die Grundlagen der Bildregistrierung geliefert und im Anschluss daran der Quadraturfilter und der Morphon-Algorithmus genau beschrieben und erläutert. Außerdem werden die aufgetretenen Schwierigkeiten in der Implementierung aufgezeigt. Aufgrund dieser Schwierigkeiten besitzt der im Rahmen dieser Bachelorarbeit implementierte Quadraturfilter zwar die grundlegend richtigen Eigenschaften, ist allerdings durch die Schwierigkeit der Parameterwahl nicht optimal. Dennoch soll ein abschließendes Beispiel am Ende der Arbeit die Funktionsfähigkeit dieses Algorithmus und damit des Quadraturfilters belegen.

Abstract

This bachelor thesis discusses an algorithm in image registration. Image registration is a branch of image processing and plays an important role in medical imaging. The algorithm presented here is called the Morphon algorithm. This algorithm is based on a set of quadrature filters. The thesis begins with an overview of the required mathematical background, provides the concepts of image registration and leads into the presentation and explanation of the quadrature filter and the Morphon algorithm. Furthermore, the difficulties of implementation is pointed out. This is the reason why the quadrature filter implemented within this thesis has the right basic principles but not the optimal properties. Nevertheless a final example will document the efficiency of the Morphon algorithm and therefore the quadrature filter.

Inhaltsverzeichnis

1	Einleitung	1
2	Grundlegende Begriffe der Bildverarbeitung	3
2.1	Definition eines Bildes	3
2.2	Kanten und Linien	3
2.3	Definition eines Signals	5
2.4	Transformationsmodelle	5
2.5	Bildregistrierung und deren Anwendung in der Medizin	6
3	Filter im Ortsraum	9
3.1	Lineare und verschiebungsinvariante Operatoren	9
3.2	Faltung	10
3.3	Filter zum Extrahieren von Kanten	10
3.3.1	Sobel-Operator	11
3.3.2	Prewitt-Operator	12
3.3.3	Laplace-Operator	12
4	Filter im Frequenzraum	15
4.1	Ortsraum versus Frequenzraum	15
4.1.1	Einschub zu komplexen Zahlen	15
4.1.2	Einschub Impulsfunktionen	16
4.2	Fouriertransformation	17
4.3	Wavelet	19
4.4	Fensterfunktion	20
4.5	Gaborfilter	22
4.6	Quadraturfilter	23
4.6.1	Die Hilberttransformation	24
4.6.2	Das analytische Signal	25
4.6.3	Beispiel: Der Quadraturfilter zur Kantenerkennung	30
5	Der Morphon-Algorithmus	33
5.1	Verschiebungsschätzung	33
5.2	Bestimmung des Deformationsfeldes	35
5.3	Deformation	36
6	Beispiel: Anwendung des Morphon-Algorithmus	37
7	Fazit	39
	Literaturverzeichnis	41

1 Einleitung

Bilder – für die meisten Menschen sind Bilder der einflussreichste Zugang zur Welt. Die Lichtstrahlen, die in unserer Umgebung reflektiert werden, fallen auf die Netzhaut im Auge. Die Sehzellen in der Netzhaut wandeln das Licht in Nervenimpulse um, die durch den Sehnerv an das Gehirn weitergeleitet werden. Dort entsteht schließlich das Bild der Umgebung [7]. Mit Hilfe dieser Bilder, die mehrmals pro Sekunde im Gehirn neu berechnet werden, kann sich der Mensch in seiner Umgebung mühelos orientieren. Der Mensch muss sich im Grunde genommen keine weiteren Gedanken über diese Tatsache machen. Er sieht 'einfach' Bäume, Straßen, Berge, Meer etc. und kann sich dementsprechend verhalten.

Schwieriger wird es, wenn man dies mit einem Computer bestimmen möchte. Denn dabei stößt man direkt auf einige Fragen. Wie lässt sich ein Bild genau definieren? Was macht ein Bild aus? Wie lassen sich aus Bildern Informationen entnehmen und wie können diese weiterverarbeitet werden?

Diese Arbeit beschäftigt sich Schritt für Schritt mit diesen Fragen und daran anknüpfend mit einem Algorithmus, dem Morphon-Algorithmus, der erstmals in [13] und [23] vorgestellt wurde. Der Schlüssel dieses Algorithmus ist ein Quadraturfilter, der hier besonders untersucht werden soll.

Dieser Algorithmus beschäftigt sich mit einem allgemeinen Problem in der Bildregistrierung. Zwei Bilder der gleichen Szene sind gegeben, ein Ausgangsbild und ein Zielbild. Diese beiden Bilder unterscheiden sich voneinander, weil sie beispielsweise zu verschiedenen Zeitpunkten oder auf unterschiedliche Weise aufgenommen wurden. Ziel ist es nun, das Ausgangsbild durch eine Transformation so zu verändern, dass der Unterschied zwischen dem veränderten Ausgangsbild und dem Zielbild möglichst klein wird [16]. Damit lassen sich Aussagen darüber treffen, wie das eine Bild sich vom anderen genau unterscheidet. Diese Informationen werden z. B. in der Medizin benötigt. Es können zwei Bilder eines Tumors vorliegen, zwischen denen eine gewisse Zeitspanne liegt und man möchte nun wissen wie genau sich der Tumor verändert hat, um eine bestmögliche Therapie anwenden zu können [3]. Es kann sich auch um Bilder handeln, die mit unterschiedlichen Geräten aufgenommen wurden und deren verschiedene Bildinformationen kombiniert werden sollen [3].

Die Besonderheit des hier vorgestellten Algorithmus ist, dass dieser phasenbasiert ist. Die Phasenunterschiede zwischen dem Ausgangsbild und dem Zielbild werden mit einem Quadraturfilter bestimmt. Die Aufarbeitung und Erklärung des Quadraturfilters und dessen Verwendung in dem Morphon-Algorithmus soll Gegenstand dieser Arbeit sein. Zudem soll dessen Funktionalität an einem Beispiel gezeigt werden. Bei der Implementierung traten einige Schwierigkeiten auf, auf die in dieser Arbeit noch genau eingegangen wird. Aus diesem Grund basieren die generierten Beispiele auf der Implementierung von Daniel Forsberg, die in [4] gefunden werden kann.

1 Einleitung

Im nachfolgenden Kapitel 2 werden grundlegende Begriffe der Bildverarbeitung eingeführt, die für das allgemeine Verständnis wichtig sind. Auf die ersten der oben genannten Fragen werden Antworten gegeben, beispielsweise findet man dort eine Definition eines Bildes.

In Kapitel 3 gehen wir näher an ein Bild heran und schauen uns Operationen an, die direkt auf ein Bild angewandt werden können. Es beginnt mit Punktoperationen, gefolgt von Nachbarschaftsoperationen und an diese anschließend folgt die Faltung, eine sehr wichtige und viel genutzte Operation in der Bildverarbeitung [6]. Am Ende dieses Kapitels beschäftigen wir uns mit Operationen, die aus einem Bild Strukturen herausfiltern können.

In Kapitel 4 wird der Frequenzraum eingeführt und damit zusammenhängend die Fourierreihe und die Fouriertransformation. Wir schauen uns kurz unterschiedliche Arten von Filtern im Frequenzraum an, bevor wir auf Wavelets zu sprechen kommen, die im Zusammenhang mit dem Quadraturfilter stehen. Den Quadraturfilter möchten wir im Anschluss über die Hilberttransformation und das analytische Signal einführen. Am Ende dieses Kapitels steht ein Anwendungsbeispiel des Quadraturfilters, das diesen als Kantenfilter zu erkennen gibt.

Der Quadraturfilter wird in Kapitel 5 benötigt, da dieser nun in den Morphon-Algorithmus eingefügt wird. So können die verschiedenen Schritte des Algorithmus beschrieben und erläutert werden.

Im Anschluss daran wird in Kapitel 6 ein Beispiel des Morphon-Algorithmus gegeben, wodurch dessen Funktionalität und die des Quadraturfilters belegt wird.

Zum Abschluss dieser Bachelorarbeit folgt ein Fazit in Kapitel 7.

2 Grundlegende Begriffe der Bildverarbeitung

2.1 Definition eines Bildes

Ein Bild stellt eine flächenhafte Verteilung der Bestrahlungsstärke in einer Ebene dar [10]. Um dem Computer ein Bild 'verständlich' zu machen, muss dieses digitalisiert werden. Wie auch in [18] nachzulesen ist, ist dazu zuerst eine Rasterung und im Anschluss daran eine Quantisierung notwendig. Dazu wird ein Bild durch eine Überlagerung eines rechteckigen oder quadratischen Gitters in einzelne Rasterflächenstücke unterteilt. Dieses Raster wird als Bildmatrix bezeichnet. Ein Element der Bildmatrix heißt Bildpunkt bzw. Pixel (engl.: picture element). Damit lässt sich nun jeder Punkt des Bildes als Matrixeintrag der Bildmatrix ansprechen. Bei der anschließenden Quantisierung wird jedem Pixel ein Wert zugeordnet.

Ein Bild I ist eine Abbildung, die jedem Punkt x , der zu einem bestimmten Gebiet $\Omega \subset \mathbb{R}^d$ gehört, einen Grauwert $I(x)$ zuordnet. Dabei bezeichnet $d \in \mathbb{N}$ die Dimension des Gebietes. Da in dieser Arbeit 2D-Grauwertbilder verwendet werden, beschränken wir uns auf $d = 2$. Grauwertbilder können beschrieben werden als

$$I : \Omega \rightarrow G,$$

wobei G die Grauwertmenge bezeichnet (vgl. [17]). Bei einem Schwarz/Weiß-Bild besteht die Grauwertmenge nur aus den zwei Werten, 0 für schwarz und 1 für weiß. Bei Grauwertbildern wird die Menge $G = \{0, 1, 2, \dots, 255\}$ verwendet, wobei 0 als schwarz und 255 als weiß interpretiert wird. Der Grauwert stellt den Helligkeits- bzw. Intensitätswert eines einzelnen Bildpunktes dar [6]. Die Unterteilung in 256 Grauwerte ist sinnvoll, da diese mit einem Byte (1 Byte = 8 Bit und $2^8 = 256$) dargestellt werden können und somit der Speicherplatzbedarf optimal ausgenutzt ist.

In einem Bildraster hat jedes Pixel zwei Arten von Nachbarschaften [18]. Eine Nachbarschaft ist eine kleine Bildregion um ein Pixel. Zum einen gibt es die Vierer-Nachbarschaft. Zu dieser gehören die Pixel um ein Pixel P , die jeweils eine gemeinsame Kante mit Pixel P haben. Angenommen das Pixel P hat die Koordinaten (x, y) , so sind die Koordinaten der vier Nachbarn gegeben durch $(x, y - 1)$, $(x, y + 1)$, $(x - 1, y)$, $(x + 1, y)$. Zum anderen gibt es die Achter-Nachbarschaft. Hier kommen zu den vier Nachbarn noch die vier diagonalen Nachbarn hinzu, diese haben mit P nur eine Ecke gemeinsam. Die vier diagonalen Nachbarn sind gegeben durch die Koordinaten $(x - 1, y - 1)$, $(x - 1, y + 1)$, $(x + 1, y - 1)$, $(x + 1, y + 1)$.

2.2 Kanten und Linien

Linien und Kanten definieren Strukturen in einem Bild. Diese geben meist mehr Informationen als beispielsweise homogen graue oder einfarbige Bereiche. Linien und Kanten sind daher sehr charakteristisch für ein Bild. Das wird besonders klar, wenn

2 Grundlegende Begriffe der Bildverarbeitung

man sich vor Augen führt, dass dem menschlichen Auge selbst Strichzeichnungen genügen, um Personen oder Objekte zu erkennen. In der Bildverarbeitung möchte man diese Strukturen extrahieren. Damit lassen sich Aussagen über ein Bild machen. Es können Informationen über ein Bild gewonnen werden und das Bild kann weiterverarbeitet werden.

Doch was genau versteht man unter einer Kante bzw. einer Linie?

Wie in [18] beschrieben ist, sind Kanten und Linien Stellen im Bild mit abrupter Helligkeitsänderung. Betrachten wir ein Grauwertbild $I = (i(x, y))$ mit $i(x, y) \in G$ [18], das beispielsweise ein Bildausschnitt einer vergrößerten Grauwertkante sein könnte und den Übergang zwischen einem hellen und einem dunklen Bildbereich darstellt (s. Abbildung 2.1).

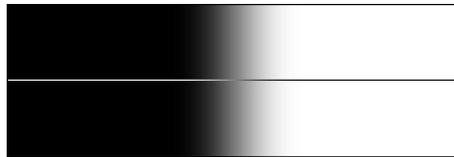


Abbildung 2.1: Bildausschnitt einer vergrößerten Grauwertkante [18]

Von diesem Grauwertbild kann ein Grauwertprofil bestimmt werden. Ein Grauwertprofil ist eine grafische Darstellung der Grauwerte entlang einer beliebig im Bild verlaufenden Linie. Das Grauwertprofil des Grauwertbildes aus Abbildung 2.1, entlang der eingezeichneten Linie von links nach rechts, ist in Abbildung 2.2 dargestellt.

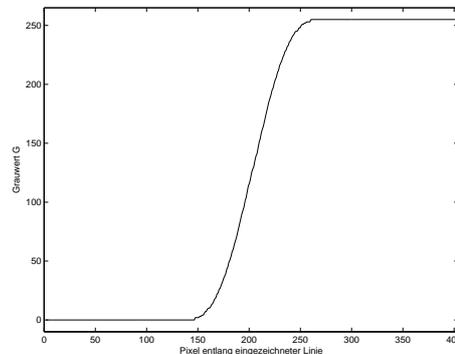


Abbildung 2.2: Grauwertprofil des Grauwertbildes aus Abbildung 2.1 entlang der eingezeichneten Linie von links nach rechts [18].

Es lässt sich eine starke Intensitätsänderung feststellen. In dem Grauwertprofil erkennt man einen starken lokalen Anstieg. Demnach liegt nahe, eine Kante über die Steigung des Grauwertprofils, somit über dessen Gradienten, auf den ich später noch zu sprechen kommen möchte, zu charakterisieren. Aus Abbildung 2.2 ist ersichtlich, dass in einem homogenen oder nahezu homogenen Bildbereich der Betrag des Gradienten des Grauwertprofils gleich null oder sehr klein ist, wohingegen der Gradient

des Grauwertprofils im Bereich einer Kante einen großen Betrag aufweist. Diese Charakterisierung einer Kante ist nicht vollkommen ausreichend, da hierunter auch z. B. Strukturen mit vielen Punkten fallen würden. Das würde bedeuten, dass die Richtungen mehr oder weniger zufällig verteilt wären. Aus diesem Grund verlangt man von einer Kante zusätzlich, dass sie sich durch eine Vorzugsrichtung des Gradienten auszeichnet und damit kontinuierlich verläuft. Ab welcher Ausdehnung man von einer Kante spricht, muss im jeweiligen Anwendungsfall festgelegt werden [18].

Im Vergleich zu einer Kante ist eine Linie eine sehr ähnliche Struktur. Eine Linie kann nach [18] definiert werden als ein Objekt, das von beiden Seiten durch eine Kante begrenzt ist. Diese beiden Kanten dürfen zum einen nicht zu weit voneinander entfernt sein und müssen zum anderen weitgehend parallel verlaufen. Bei welchem Abstand und welcher Ausdehnung der begrenzenden Kanten eine Struktur als Linie angesehen wird oder nicht, ist wiederum vom jeweiligen Anwendungsfall abhängig.

Nach [7] kann die Unterscheidung zwischen einer Kante und einer Linie auch über die Form des Grauwertprofils erfolgen. In Abbildung 2.3 ist das Grauwertprofil einer Kante und einer Linie nebeneinander dargestellt. Vergleicht man diese beiden Grauwertprofile, kann in dem der Kante die Form einer ungeraden Funktion und in dem der Linie die Form einer geraden Funktion wiedererkannt werden. Eine Funktion $f : D \rightarrow \mathbb{R}$ mit $D \subseteq \mathbb{R}$ wird als gerade Funktion bezeichnet, falls für alle $x \in D$ gilt: $f(-x) = f(x)$. Eine Funktion wird als ungerade bezeichnet, falls $f(x) = -f(-x)$ für alle $x \in D$ gilt [2]. Die Eigenschaft, dass sich Kanten demnach als ungerade Funktionen und Linien als gerade Funktionen auffassen lassen, wird beim Quadraturfilter noch eine wichtige Rolle spielen.

2.3 Definition eines Signals

Signale sind Träger von Informationen und spielen im Alltag eine wichtige Rolle. Beispiele sind Sprache, Musik, biomedizinische Messdaten aber auch Bildsignale [22]. Ein Signal im physikalischen Sinn, in welchem der Begriff 'Signal' hier auch verwendet werden soll, lässt sich im Allgemeinen mathematisch als eine Funktion beschreiben. Es kann zwischen kontinuierlichen und diskreten Signalen unterschieden werden.

Kontinuierliche Signale sind Signale, die zu jedem Zeitpunkt definiert sind und jede Stelle auf der Wertachse annehmen können. Ein solches Signal könnte man mathematisch wie folgt als Funktion beschreiben: $f : \mathbb{R} \rightarrow \mathbb{R}$.

Ein diskretes Signal, ist ein Signal, das nur an bestimmten Stellen definiert ist. Meistens sind die Abstände dieser Stellen äquidistant gewählt, d. h. sie haben zueinander den gleichen Abstand. (Vgl. [15])

2.4 Transformationsmodelle

Transformationsmodelle legen fest, wie ein registriertes Bild verändert werden darf. Es werden lokale und globale Modelle unterschieden. Bei einem globalen Modell wird

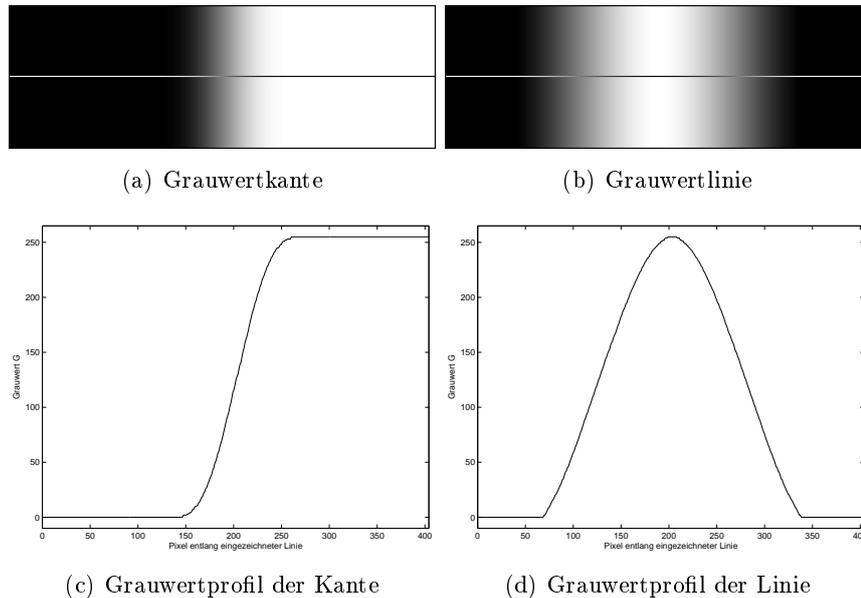


Abbildung 2.3: Darstellung einer Grauwertkante und einer Grauwertlinie und deren jeweiliges Grauwertprofil (vgl. [18]).

die gleiche Transformation auf das gesamte Bild angewandt, während bei einem lokalen Modell auf verschiedene Teile des Bildes unterschiedliche Transformationen angewendet werden können [23]. Außerdem lassen sich Transformationen mit verschiedener Freiheitsgradanzahl unterscheiden [23]. Nach [17] ist die einfachste Transformation die Translation. Durch diese erfolgt eine Verschiebung des Bildes. Die affin lineare Transformation erlaubt zusätzlich noch Rotation, Scherung und Skalierung. Eine spezielle affin lineare Transformation ist die rigide oder starre Transformation, die ausschließlich Translation und Rotation erlaubt. Transformationen, die noch mehr Freiheitsgrade erlauben, werden als elastische Transformationen bezeichnet.

2.5 Bildregistrierung und deren Anwendung in der Medizin

Wie in [3] nachzulesen ist, ist die Bildregistrierung eine wichtige Anwendung in der digitalen Bildverarbeitung, um zwei oder mehr Bilder dergleichen Szene bestmöglich in Übereinstimmung zu bringen. Ein Bild wird als Zielbild ('reference image') R festgesetzt. Das andere wird Ausgangsbild ('template image') T genannt, welches optimal an das Referenzbild angepasst werden soll. Ziel ist es, hierfür eine Transformation φ zu finden, sodass der Unterschied zwischen dem veränderten Ausgangsbild $T(\varphi(x))$ und R möglichst klein ist. $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ wird hierbei als Deformation bezeichnet (vgl. [16]).

Für dieses Problem gibt es verschiedene Lösungsansätze. Im Allgemeinen lassen sich zwei Arten der Bildregistrierung unterscheiden, die parametrische und die nicht-

parametrische Bildregistrierung [16]. Bei der parametrischen Registrierung werden nur affin lineare Transformationen zugelassen. In einigen Fällen ist die parametrische Registrierung allerdings nicht mehr ausreichend, um eine geeignete Deformation zu finden. Dafür verwendet man die nicht-parametrische Registrierung, die jedes Pixel einzeln betrachtet. Da der Morphon-Algorithmus nach [3] zu der nicht-parametrischen Registrierung gezählt werden kann, möchte ich kurz auf das Grundgerüst einer nicht-parametrischen Registrierung zu sprechen kommen. Die nicht-parametrische Registrierung ist aufgeteilt in die Berechnung eines Deformationsfeldes, einer Ansammlung der Deformationen in einem Feld und einer Regularisierung des Deformationsfeldes. Im ersten Schritt wird zwischen dem Ausgangsbild und dem Zielbild eine erste Deformation bestimmt, die in einem Deformationsfeld gespeichert wird. Der Morphon-Algorithmus ist ein iterativer Algorithmus, der nun erneut eine Deformation berechnet und diese zu dem schon vorhandenen Deformationsfeld hinzufügt. Dieser Vorgang wird als Ansammlung der Deformationen in einem Feld bezeichnet. Wichtig ist, dass in jedem Schritt das Deformationsfeld regularisiert wird, um sicherzustellen, dass die Transformation bzw. Deformation das gewünschte Ergebnis liefert [3].

Zunächst möchte ich die Bildregistrierung anhand einer kleinen Auswahl von Anwendungsbeispielen in der Medizin motivieren. Diese können in [3] nachgelesen werden. In der Medizin spielt die Bildregistrierung in vielen verschiedenen Situationen eine große Rolle.

Eine der häufigsten Anwendungen ist die Angleichung von Bildern desselben Patienten, die jedoch zu verschiedenen Zeitpunkten und/oder mit unterschiedlichen Geräten aufgenommen wurden, um die verschiedenen Bildinformationen zu kombinieren. Des Weiteren können diese Informationen genutzt werden, um die Bewegung zwischen den Bildern zu quantifizieren.

Eine andere Anwendung ist der Ausgleich von Artefakten im Bild, die durch Bewegung während der Aufnahme des Bildes oder zwischen Bildsequenzen entstehen können, z. B. bei der Magnetresonanztomographie (MRT).

Außerdem kann durch die Registrierung der Bilder einer bestimmten Patientengruppe eine Repräsentation dieser erstellt werden. Diese kann dann z. B. zur atlasbasierten Segmentierung verwendet werden.

Um die Besonderheit des Quadraturfilters hervorzuheben, möchte ich im nächsten Kapitel die einfachsten Filter der Bildregistrierung zur Kantenerkennung darstellen, mit denen ich auch zum Schluss den Quadraturfilter vergleichen möchte.

3 Filter im Ortsraum

Nach [10] sind Filter Operationen, die auf ein Bild angewandt werden, um dieses zu verändern. Dadurch lässt sich beispielsweise Rauschen unterdrücken oder es lassen sich lokale Strukturen wie Kanten, Linien, Oberflächen etc. feststellen und unterscheiden. Damit wird auch der Begriff 'Filter' klar. Ein Filter extrahiert, je nach verwendetem Filter, bestimmte interessierende Eigenschaften aus einem Bild.

Werden Operationen auf einzelne Pixel eines Bildes angewandt, so spricht man von Punktoperationen. Diese werden vor allem im ersten Schritt der Bildverarbeitung verwendet, um z. B. den Bildkontrast zu verstärken.

Daneben gibt es Nachbarschaftsoperationen, die auf Pixel in einer kleinen Umgebung operieren, diese kombinieren und ein neues Bild mit neu gewonnenen Informationen erzeugen.

3.1 Lineare und verschiebungsinvariante Operatoren

In diesem Unterkapitel soll nach [21] auf lineare und verschiebungsinvariante Operatoren eingegangen werden, deren Definition im Weiteren direkt auf die Faltung führt.

Definition 3.1 (Linearer Operator [21])

Es seien F und G Vektorräume in \mathbb{R} oder \mathbb{C} . Eine Abbildung $O : F \rightarrow G$ heißt linearer Operator, wenn für alle $f_1, f_2 \in F$ und $a_1, a_2 \in \mathbb{R}$ (bzw. $\in \mathbb{C}$) gilt:

$$O(a_1 f_1 + a_2 f_2) = a_1 \cdot O(f_1) + a_2 \cdot O(f_2).$$

Das bedeutet, dass ein Signal durch einen linearen Filter nicht verzerrt, die Form des Signals also nicht grundlegend verändert wird. Wenn das Eingangssignal um einen bestimmten Faktor vergrößert wird, so wird auch das Ausgangssignal entsprechend dieses Faktors vergrößert.

Definition 3.2 (Verschiebungsinvarianz [21])

Es seien F und G Vektorräume in \mathbb{R} bzw. \mathbb{C} . Eine Abbildung $O : F \rightarrow G$ heißt verschiebungsinvarianter Operator, wenn gilt:

$$O[f(\bullet + z)](x) = O[f](x + z) \quad \forall x, z \in \mathbb{R},$$

wobei \bullet ein Platzhalter für den Definitionsbereich ist.

Mit dieser Definition stellt man fest, dass das Operatorergebnis eines verschiebungsinvarianten Operators nicht vom Ort der Operation abhängig ist. Anwendungen dieses Operators liegen z. B. bei der Beschreibung einer gleichmäßigen Unschärfe in einem Bild.

3.2 Faltung

Ein häufig benutzter linearer und verschiebungsinvarianter Operator ist die Faltung. Die Faltung ist ein Operator, der zwei Funktionen benötigt. Die eine Funktion ist beispielsweise ein Signal f , die andere Funktion wird als Faltungsmaske bezeichnet. Wie in [6] beschrieben, geschieht bei der Faltung der zwei Funktionen folgendes: Zuerst wird das Signal gespiegelt, im Anschluss wird punktweise jedes Element der Faltungsmaske mit jedem Element des Signals multipliziert. Somit lässt sich die Funktion, die dabei entsteht, als gewichteten Mittelwert von f mit der Gewichtung g verstehen. In der Bildverarbeitung sind es meistens diskrete Funktionen, die miteinander gefaltet werden sollen.

Definition 3.3 (diskrete Faltung)

Gegeben sei eine Bildmatrix $f(x, y)$ der Dimension $M \times N$ mit $M, N \in \mathbb{R}$ und eine Faltungsmaske $g(x, y)$. Die diskrete Faltung ist gegeben durch (vgl.[10]):

$$(g * f)(x, y) = \sum_{s=0}^{M-1} \sum_{t=0}^{N-1} g(s, t) f(x - s, y - t).$$

Im Laufe der Arbeit werden einige Berechnungen durchgeführt, in denen unter anderem folgender Satz verwendet wird. Diese Eigenschaften lassen sich in [10] nachlesen.

Satz 3.1 (Eigenschaften der Faltung)

Für die Funktionen f, g und h mit $f, g, h : \mathbb{R} \rightarrow \mathbb{R}$ besitzt die Faltung folgende Eigenschaften [10]:

- (i) Kommutativität: $f * g = g * f$
- (ii) Assoziativität: $(f * g) * h = f * (g * h)$
- (iii) Distributivität: $f * (g + h) = (f * g) + (f * h)$
- (iv) Multiplikation mit einem Skalar ($a \in \mathbb{R}$ oder \mathbb{C}): $a(f * g) = (af) * g = f * (ag)$

3.3 Filter zum Extrahieren von Kanten

Wie in Kapitel 2.2 erwähnt, ist eine Kante eine Struktur in einem Bild, die zwei Objekte voneinander trennt. Um den Ort und die Richtung einer Kante zu lokalisieren ist es sinnvoll, die Stärke einer Kante über den Betrag des Gradienten, also über die Steilheit der Grauwertfunktion, zu bestimmen. Der Gradient ist ein Vektor, dessen Länge die Kantenstärke darstellt und dessen Richtung orthogonal zur Kantenrichtung verläuft [21]. Gehen wir zunächst davon aus, dass das betrachtete Signal stetig und differenzierbar ist und die Form $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ besitzt. Von dieser kann der Gradient bestimmt werden: $\nabla f = (\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y})$. Da Bilder nicht stetig

sind, kann dies nicht direkt auf Bildsignale übertragen werden. Dennoch kann nach [10] eine Annäherung des Gradienten über diskrete Differenzen erfolgen:

$$\begin{aligned}\frac{\partial f(x, y)}{\partial x} &= \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x}, \\ \frac{\partial f(x, y)}{\partial y} &= \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - f(x, y - \Delta y)}{2\Delta y}.\end{aligned}$$

Für $\Delta x = 1$ bzw. $\Delta y = 1$ kann diese Operation durch die Faltung mit folgenden zwei verschiedenen Faltungsmasken durchgeführt werden [21]:

$$\begin{aligned}\frac{\partial f(x, y)}{\partial x} &\approx h_x * f(x, y) && \text{mit } h_x = \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}, \\ \frac{\partial f(x, y)}{\partial y} &\approx h_y * f(x, y) && \text{mit } h_y = \frac{1}{2} \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.\end{aligned}$$

Die Abschätzung des Gradienten kann nun verbessert werden, indem die Faltungsmasken ein wenig modifiziert werden.

3.3.1 Sobel-Operator

Ein Operator, der auf diesen Faltungsmasken basiert, ist der Sobel-Operator, dessen Filtermasken wie folgt aussehen [21]:

$$S_x = \frac{1}{8} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \quad \text{und} \quad S_y = \frac{1}{8} \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}.$$

Operatoren mit derartigen Filtermasken sprechen am stärksten auf horizontal oder vertikal verlaufende Grauwertkanten an.

Weiterhin lassen sich auch Kompassfilter bilden, die aus acht Filtermasken für acht unterschiedliche Richtungen bestehen.

3.3.2 Prewitt-Operator

Die Filtermasken des Prewitt-Operators erhält man durch inkrementelle Rotation um 45° [21]:

$$\begin{aligned}
 h_1 &= \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}, & h_2 &= \begin{pmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix}, \\
 h_3 &= \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}, & h_4 &= \begin{pmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \\
 h_5 &= \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}, & h_6 &= \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix}, \\
 h_7 &= \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix}, & h_8 &= \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{pmatrix}.
 \end{aligned}$$

Die hier vorgestellten Operatoren basierten bis jetzt alle auf der ersten Ableitung. Nach [21] lassen sich auch über die zweite Ableitung der Bildfunktion wichtige Informationen über Kanten entnehmen. Ebenso wie die erste Ableitung besteht die zweite Ableitung einer mehrdimensionalen Funktion aus einer Menge von partiellen Ableitungen. Für die Ableitung einer zweidimensionalen Funktion $f(x, y)$ gibt es vier partielle zweite Ableitungen:

$$\frac{\partial^2 f}{\partial x^2}(x, y), \quad \frac{\partial^2 f}{\partial y^2}(x, y), \quad \frac{\partial^2 f}{\partial x \partial y}(x, y), \quad \frac{\partial^2 f}{\partial y \partial x}(x, y).$$

$\frac{\partial^2 f}{\partial x^2}(x, y)$ bedeutet, dass die Funktion f zweimal partiell nach x abgeleitet wird und $\frac{\partial^2 f}{\partial x \partial y}(x, y)$ bedeutet, dass die Funktion erst nach x und dann nach y abgeleitet wird. Die zweiten partiellen Ableitungen können in der Hesse-Matrix notiert werden:

$$H(f) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2}(x, y) & \frac{\partial^2 f}{\partial y \partial x}(x, y) \\ \frac{\partial^2 f}{\partial x \partial y}(x, y) & \frac{\partial^2 f}{\partial y^2}(x, y) \end{pmatrix}.$$

3.3.3 Laplace-Operator

Nach [21] ist beispielsweise der Laplace-Operator ein Operator, der über die zweite Ableitung in einem Bild Kanten erkennt. Man sollte sich hier nochmal vor Augen führen, dass an der Stelle, an der die zweite Ableitung ein Vorzeichenwechsel, auch Nulldurchgang genannt, hat, bei der ersten Ableitung ein Extremum vorliegt. Eine Kante verläuft entlang diesem Vorzeichenwechsel in der zweiten Ableitung. Diese Eigenschaft macht man sich bei der Kantendetektion mit dem Laplace-Operator zunutze. Gehen wir zunächst davon aus, dass ein stetiges und zweimal differenzierbares

Signal vorliegt, $f : \mathbb{R}^2 \rightarrow \mathbb{R}$. Dessen partielle Ableitungen werden durch diskrete Differenzen approximiert, um auf diskrete Bildsignale angewandt werden zu können. Der Laplace-Operator kann auch als Spur der Hesse-Matrix betrachtet werden und ist somit die Summe der Hauptdiagonalen der Hesse-Matrix:

$$\Delta f(x, y) = \frac{\partial^2 f}{\partial x^2}(x, y) + \frac{\partial^2 f}{\partial y^2}(x, y).$$

Die partiellen Ableitungen werden durch folgende Differenzen approximiert:

$$\begin{aligned} \frac{\partial^2 f}{\partial x^2}(x, y) &= \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - 2f(x, y) + f(x - \Delta x, y)}{\Delta x^2}, \\ \frac{\partial^2 f}{\partial y^2}(x, y) &= \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - 2f(x, y) + f(x, y - \Delta y)}{\Delta y^2}. \end{aligned}$$

Für $\Delta x = 1$ bzw. $\Delta y = 1$ wird auf ein diskretes Signal der Laplace-Operator über eine Faltung (vergleichbar mit der des Sobel-Operators) mit folgenden Filtermasken angewandt:

$$L_1 = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} \text{ oder } L_2 = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}.$$

4 Filter im Frequenzraum

4.1 Ortsraum versus Frequenzraum

Die Originalrepräsentation einer Funktion ist die Repräsentation im Ortsraum [21]. Um zu verstehen was der Frequenzraum ist, kann man sich folgendes, aus [21] entnommenes, Szenario vorstellen: Man erhält ein Bild, das jedoch während einer Analogübertragung über ein Kabel wegen eines benachbarten, schlecht abgeschirmten Stromkabels mit der sinusförmigen Schwingung des Wechselstroms überlagert wird. Nun möchte man natürlich diese Störung beheben und müsste dafür die Offset (die Phase), die Höhe (die Amplitude), und die Häufigkeit (Frequenz) der Schwingung berechnen, und im Anschluss diese Schwingung entfernen. Einfacher wäre es, wenn man das Bild in all seine Schwingungen zerlegen könnte, um die störende Schwingung zu ermitteln und daraufhin zu entfernen. Genau dies ist im Frequenzraum gegeben. Denn hier wird das Signal als Zusammensetzung aus einzelnen Grundfrequenzen beschrieben.

Wie auch in [6] nachzulesen ist, war der Franzose Jean Baptiste Joseph Fourier ein bedeutender Mathematiker des 19. Jahrhunderts. Im Jahre 1822 publizierte er das Buch 'La Théorie Analytique de la Chaleur', das die Grundlagen der Fourier-Analyse beinhaltet. Eine der Hauptaussagen dieses Buches ist das Konzept der Fourierreihe. Dies besagt, dass jede periodische Funktion als Summe von Sinus- und/oder Kosinusfunktionen unterschiedlicher Frequenz, multipliziert mit verschiedenen Koeffizienten, ausgedrückt werden kann.

Sogar nicht periodische Funktionen, deren Fläche unter der Kurve jedoch endlich ist, können mithilfe eines Integrals über Sinus- und Kosinusfunktionen, multipliziert mit einer Gewichtungsfunktion, ausgedrückt werden. In diesem Fall spricht man von der Fouriertransformation, dessen Nutzen und Anwendungsbereich noch weitreichender als die der Fourierreihe ist. Eine schöne Eigenschaft, die der Fourierreihe und der Fouriertransformation zu eigen ist, ist deren Invertierbarkeit ohne Informationsverlust.

4.1.1 Einschub zu komplexen Zahlen

Hier sollen lediglich die im Folgenden genutzten Notationen dargestellt werden. Eine komplexe Zahl C ist definiert als:

$$C = Re + iIm,$$

wobei Re und Im reelle Zahlen sind und i die imaginäre Einheit ist. Damit sind reelle Zahlen eine Untergruppe der komplexen Zahlen, bei denen $Im = 0$ ist.

Die komplexe Zahl i hat folgende Eigenschaften:

$$i^2 = -1 \quad \text{und somit} \quad i = \sqrt{-1}$$

4 Filter im Frequenzraum

Die komplexe Konjugation, dessen Zeichen ein hochgestelltes * ist, ist definiert als:

$$C^* = Re - iIm.$$

Komplexe Zahlen werden geometrisch in einer Ebene als Vektoren dargestellt, dessen Abszisse die reelle Achse und dessen Ordinate die komplexe Achse ist. Somit stellt die komplexe Zahl $Re + iIm$ den Punkt (Re, Im) im rechtwinkligen Koordinatensystem dar.

Nützlich sind folgende Darstellungen

$$C = |C|(\cos \theta + i \sin \theta),$$
$$C = |C|e^{i\theta}.$$

$|C| = \sqrt{Re^2 + Im^2}$ ist dabei die Länge des Vektors vom Koordinatenursprung zum Punkt (Re, Im) . Den Winkel zwischen dem Vektor und der reellen Achse bezeichnet man als θ .

4.1.2 Einschub Impulsfunktionen

Die folgende Einführung in die Impulsfunktionen beruht auf [9]. Diese werden im weiteren Verlauf dieser Arbeit noch eine wichtige Rolle spielen.

Definition 4.1 (Heaviside-Funktion (Einheitssprungfunktion))

Für $x \in \mathbb{R}$ ist die Heaviside-Funktion $\Theta : \mathbb{R} \rightarrow \mathbb{R}$ definiert als:

$$\Theta(x) = \begin{cases} 1 & \text{für } x > 0 \\ \frac{1}{2} & \text{für } x = 0 \\ 0 & \text{für } x < 0 \end{cases}.$$

Nun können, durch die Überlagerung von geeignet verschobener Sprungfunktionen, verschiedene Treppenfunktionen zusammengestellt werden.

Beispiel 4.1 (Signumfunktion (Vorzeichenfunktion))

Mit $x \in \mathbb{R}$ kann die Signumfunktion $\text{sgn} : \mathbb{R} \rightarrow \mathbb{R}$ gebildet werden:

$$\text{sgn}(x) = \begin{cases} +1 & \text{für } x > 0 \\ 0 & \text{für } x = 0 \\ -1 & \text{für } x < 0 \end{cases}$$
$$= 2\Theta(x) - 1.$$

Beispiel 4.2 (Rechteckimpuls)

Mit $x \in \mathbb{R}$ und $T \in \mathbb{R}$ beliebig aber fest, gilt für den Rechteckimpuls $r : \mathbb{R} \rightarrow \mathbb{R}$:

$$r(x) = \begin{cases} 1 & \text{für } |x| < T \\ 0 & \text{sonst} \end{cases}$$
$$= \Theta(x + T) - \Theta(x - T).$$

Die diskrete Impulsfunktion ist wie folgt definiert:

Definition 4.2 (Diskreter Einheitsimpuls (Dirac-Impuls))

Mit x als eine diskrete Variable gilt für den diskreten Einheitsimpuls $\delta : \mathbb{R} \rightarrow \mathbb{R}$:

$$\delta(x) = \begin{cases} 1 & \text{für } x = 0 \\ 0 & \text{für } x \neq 0 \end{cases} \quad \text{mit} \quad \sum_{x=-\infty}^{\infty} \delta(x) = 1.$$

4.2 Fouriertransformation

Ein Analogon zu den Fourierreihen ist für aperiodische Funktionen die Fouriertransformation. Hierbei wird die Funktion wiederum in ihre Frequenzanteile zerlegt. Dieses Frequenzspektrum bezeichnet man als die zu der Funktion gehörige Fouriertransformierte.

Definition 4.3 (Eindimensionale kontinuierliche Fouriertransformation [6])

Sei f eine integrierbare kontinuierliche Funktion und x eine kontinuierliche Variable. Die Fouriertransformation ist mit $\mathcal{F} : \mathcal{L}^1(\mathbb{R}) \rightarrow \mathcal{L}^1(\mathbb{R})$ definiert durch:

$$\mathcal{F}(f)(x) = F(\mu) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi\mu x} dx.$$

Die inverse Fouriertransformation transformiert die Funktion vom Frequenzraum wieder in den Ortstraum zurück und wird folgendermaßen gebildet:

$$\mathcal{F}^{-1}(F)(\mu) = f(x) = \int_{-\infty}^{\infty} F(\mu) e^{i2\pi\mu x} d\mu.$$

Die diskrete Fouriertransformation bildet eine diskrete endliche Funktion, die periodisch fortgesetzt wird, auf ein diskretes periodisches Frequenzspektrum ab.

Definition 4.4 (Eindimensionale diskrete Fouriertransformation [6])

Sei f eine diskrete Funktion auf dem Definitionsbereich $t = [0, 1, \dots, N - 1]$ und $N \in \mathbb{N}$ die Anzahl der Abtastungen, dann ist die Fouriertransformierte definiert durch:

$$\mathcal{F}(f)(x) = \sum_{x=0}^{M-1} f(x) e^{-i2\pi \frac{\mu x}{M}}.$$

Den Raum, in dem sich die Fouriertransformierte befindet, wird Frequenzraum genannt. Die inverse Fouriertransformation transformiert die Funktion vom Frequenzraum wieder in den Ortsraum zurück und wird folgendermaßen gebildet:

$$\mathcal{F}^{-1}(F)(\mu) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{i2\pi \frac{\mu x}{M}}.$$

Die beiden Funktionen $\mathcal{F}(f)(x)$ und $\mathcal{F}^{-1}(F)(\mu)$, die das Signal vom Ortsraum in den Frequenzraum transformiert bzw. vom Frequenzraum in den Ortsraum zurück

4 Filter im Frequenzraum

transformiert, werden als Fouriertransformationspaar bezeichnet.

Nach [21] und [10] besitzt die Fouriertransformation verschiedene Eigenschaften, von denen im Folgenden die wichtigsten in einem Satz aufgezeigt werden sollen.

Satz 4.1 (Eigenschaften der Fouriertransformation [21], [10])

Seien f, g diskrete Funktionen auf dem Definitionsbereich $t = [0, 1, \dots, N - 1]$ und $N \in \mathbb{N}$ die Anzahl der Abtastungen. Die Fouriertransformation besitzt folgende Eigenschaften:

(i) Phase und Amplitude:

Die Fouriertransformation zerlegt eine Funktion in Wellen unterschiedlicher Frequenzen. Jede dieser Wellen ist charakterisiert durch ihre Amplitude und ihre Phase. Die Amplitude gibt beispielsweise bei einem Grauwertbild an, wie hoch der Grauwert des durch diese Welle repräsentierten Anteils im Bild ist. D. h. wenn man z. B. vor der Rücktransformation die Beträge der Funktionswerte halbieren würde, wäre das resultierende Bild auch nur halb so hell. Die Amplitude A an der Position (μ, ν) ist der Betrag des Funktionswertes. Da die Fouriertransformierte im Allgemeinen komplex ist, wird die Amplitude folgendermaßen berechnet:

$$A = |F(\mu, \nu)| = \sqrt{\operatorname{Re}^2(\mu, \nu) + \operatorname{Im}^2(\mu, \nu)}.$$

Die Phasenverschiebung jeder Welle ist durch den Winkel θ in der komplexen Ebene gegeben:

$$\theta = \tan^{-1} \left(\frac{\operatorname{Im}(\mu, \nu)}{\operatorname{Re}(\mu, \nu)} \right).$$

(ii) Linearität:

Seien f und g zwei eindimensionale Funktionen und $a, b \in \mathbb{R}$ oder \mathbb{C} . Dann gilt die Linearität $\mathcal{F}(af(x) + bg(x)) = aF(\mu) + bG(\mu)$.

(iii) Shift-Theorem-Verschiebungseigenschaft:

Wird ein Signal, auf das die Fouriertransformation angewandt werden soll, um eine bestimmte Einheit τ verzögert, so kann die Fouriertransformation bestimmt werden als Multiplikation von $e^{i2\pi\mu\tau}$ mit der Fouriertransformierten von f : $\mathcal{F}(f)(x - \tau) = e^{-i2\pi\mu\tau} F(\mu)$.

(iv) Faltungstheorem:

Eine Operation, die sich im Frequenzraum stark vereinfacht, ist die Faltung. $\mathcal{F}(f * g)(x) = F(\mu)G(\mu)$.

(v) Hermitesche Symmetrie:

Im Frequenzraum gilt für eine Funktion F und deren komplex Konjugierte F^* : $F^*(-\mu) = F(\mu)$.

(vi) Differentiationstheorem:

Sei f eine für alle $x \in \mathbb{R}$ differenzierbare Funktion mit $\lim_{x \rightarrow \pm\infty} f(x) = 0$, $\frac{\partial f}{\partial x}$ sei die erste Ableitung von f und die Fouriertransformierte F und $\frac{\partial F}{\partial \mu}$ existieren mit $\lim_{\mu \rightarrow \pm\infty} F(\mu) = 0$. Dann gilt: $\mathcal{F}\left(\frac{\partial f}{\partial x}\right)(x) = i2\pi\mu F(\mu)$.

Beweis. Der Beweis zu (iv) kann in [6], die restlichen können in [11] nachgelesen werden. \square

Mit der Einführung der Fouriertransformation und somit des Frequenzbereichs, in den ein Signal transformiert werden kann, ergeben sich neue Möglichkeiten zur Filterung eines Signals. Denn jetzt lassen sich bestimmte Frequenzgänge unterdrücken, wohingegen andere ungeschwächt hindurchgelassen werden können. Die drei grundlegenden Filtertypen sind der Tiefpassfilter, der Hochpassfilter und der Bandpassfilter. Hier soll nur auf den Bandpassfilter eingegangen werden, da der Quadraturfilter als ein solcher bezeichnet werden kann. Weiteres lässt sich beispielsweise in [15] oder [21] nachlesen.

Bandpassfilter sind Filter, die Frequenzen innerhalb eines bestimmten Bereiches passieren lassen und alle anderen Frequenzen abschwächen. Es werden die sehr hohen und die sehr tiefen Frequenzen abgeschwächt und die mittleren Frequenzen durchgelassen.

4.3 Wavelet

In [21] wird beschrieben, dass unter anderem bei der Fouriertransformation angenommen wird, dass die Wellen unterschiedlicher Frequenz, in die ein Bild zerlegt wird, im gesamten Bild vorkommen. Dies ist in der Realität meistens jedoch nicht der Fall [21]. In vielen Bildern gibt es Bereiche, in denen niedrige Frequenzen vorherrschend sind und andere Bereiche, in denen kleine, detailreiche Strukturen durch hohe Frequenzen repräsentiert werden. Ein Extremfall ist beispielsweise ein weißes Quadrat auf schwarzem Hintergrund. Es werden nur an den Kanten des Quadrates Wellen mit hoher Frequenz benötigt, um den Wechsel von schwarz zu weiß darzustellen. In allen anderen Bereichen ist der Grauwert konstant. Da aber bei der Fouriertransformation die Wellen, die die Kanten modellieren, im ganzen Bild vorhanden sind, müssen diese an allen anderen Stellen des Bildes durch andere Wellen ausgeglichen werden.

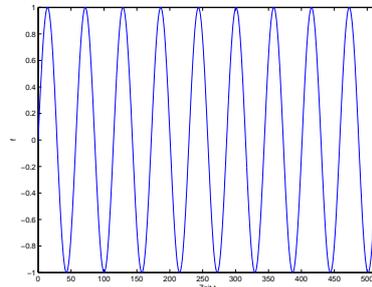
Eine bessere Repräsentation erhält man durch Wavelets als Basisfunktionen. Das Wort 'Wavelete' kommt ursprünglich aus dem französischen Wort 'ondelette', was 'kleine Welle' bedeutet. Dieses Wort wurde teils wörtlich, teils phonetisch ins Englische übertragen und so wurde aus 'onde' 'wave' und aus 'lette' 'let'. Die Wavelet Funktionen sind zwar Wellen mit unterschiedlicher Frequenz, ihre Amplitude hingegen ist nur in einem kleinen Bereich von Null verschieden. Auf diese Eigenschaft ist auch der Name 'Wavelet' zurückzuführen. Wie in [7] nachzulesen ist, kennzeichnet

4 Filter im Frequenzraum

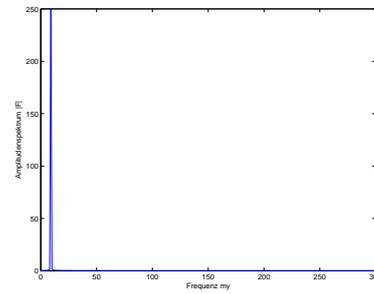
sich das Konzept der Wavelets dadurch aus, dass durch Verschiebung und Skalierung eines 'Mother-Wavelets', Waveletbasen gebildet werden können. Folglich unterscheiden sich zwei Wavelets von ihrer Mother-Wavelet durch zwei Parameter, einem Skalierungsparameter und einem Verschiebungsparameter. Dabei legt der Skalierungsparameter die Frequenz des Wavelets fest, wodurch die relative Bandbreite bestimmt ist. Die Transformation, die die Wavelets als Basisfunktion verwenden, wird Wavelet-Transformation genannt. Im Gegensatz zu den Basen einer Fouriertransformation haben die Basisfunktionen einer Wavelet-Transformation zwei Parameter, den Skalierungsparameter und den Verschiebeparameter, wobei diese beiden Parameter jeweils ganze Zahlen sind.

4.4 Fensterfunktion

Bis jetzt hat das Bild bzw. die Funktion oder das Signal, das betrachtet wird noch eine unendliche Größe. In der Praxis kann mit so einer Menge von Information natürlich nicht umgegangen werden und es ist meistens auch nicht nötig. Außerdem wird bei der Fouriertransformation angenommen, dass die Länge des betrachteten Signals ein ganzzahliges Vielfaches der Periode des Signals ist und somit ohne Unstetigkeiten aneinandergereiht werden könnte. Dies ist allerdings nicht immer der Fall. Das hat zur Folge, dass es zu Störungen im Frequenzraum kommt, welche sich durch ein 'verschmiertes' Frequenzspektrum darstellen. Diese Eigenschaft ist in Abbildung 4.1 und Abbildung 4.2 dargestellt. In Abbildung 4.1 (a) ist eine Sinusschwingung mit genau neun Perioden dargestellt und in (b) dessen Amplitudenspektrum. In Abbildung 4.2 (a) ist eine Sinusschwingung dargestellt, die nicht aus exakt neun Perioden besteht. Das bedeutet, dass eine Aneinanderreihung dieses Signals zu Unstetigkeiten führen würde. Dies zeigt sich in (b), wo das Amplitudenspektrum dargestellt ist. Vergleicht man die beiden Amplitudenspektren sieht man, dass das Amplitudenspektrum in Abbildung 4.1 genau bei der Frequenz neun einen Peak aufweist, wohingegen das Amplitudenspektrum in Abbildung 4.2 'verschmiert' ist (vgl. [14]).

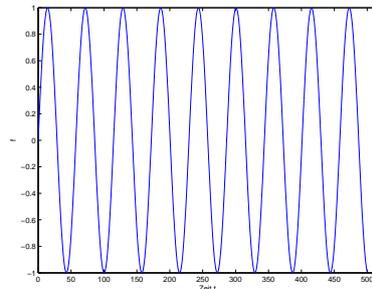


(a) Sinusfunktion

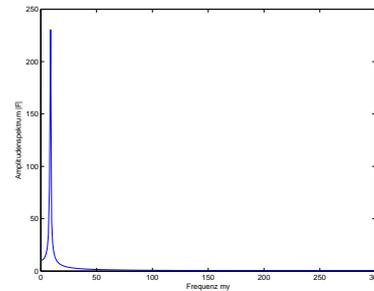


(b) Amplitudenspektrum der Sinusfunktion

Abbildung 4.1: Sinusfunktion und dessen Amplitudenspektrum, wobei die Sinusfunktion aus einer ganzen Anzahl von Perioden besteht (vgl. [15]).



(a) Sinusfunktion



(b) Amplitudenspektrum der Sinusfunktion

Abbildung 4.2: Sinusfunktion und dessen Amplitudenspektrum, wobei die Sinusfunktion nicht aus einer ganzen Anzahl von Perioden besteht (vgl. [15]).

Eine Möglichkeit mit diesem Problem umzugehen ist, eine Fensterfunktion auf das Signal anzuwenden [10]. Eine Fensterfunktion ist eine Funktion, die zu den Rändern hin auf null abfällt. Eine ideale Fensterfunktion sollte die Eigenschaft haben, dass ihre Fouriertransformierte so steil wie möglich zu den Rändern hin abnimmt und gleichzeitig eine hohe spektrale Auflösung besitzt. Damit würden sich zwei Signale, die ähnliche Frequenz und Amplitude besitzen, erkennen lassen. [22]. Da es nicht möglich ist diesen beiden Forderungen gleichzeitig gerecht zu werden, muss immer ein Kompromiss eingegangen werden. Eine Möglichkeit ist das Hann-Fenster als Fensterfunktion zu verwenden. Dieses ist definiert als $w(n) = 0.5 \cdot [1 - \cos(\frac{2\pi n}{N-1})]$ mit $n = 0, 1, \dots, N - 1$ und N die Länge der Fensterfunktion [15]. Eine Fensterfunktion wird im Ortsbereich mit dem zu fensternden Signal multipliziert. In Abbildung 4.3 ist in (a) ein nicht stetig fortsetzbares Signal, in (b) das Hann-Fenster und in (c) das gefensterte Signal abgebildet. In (d) und (e) sind die jeweiligen Amplitudenspektren von (a) und (c) dargestellt, wobei auffällt, dass in (e) die 'Versmierung' um einiges

4 Filter im Frequenzraum

geringer geworden ist.

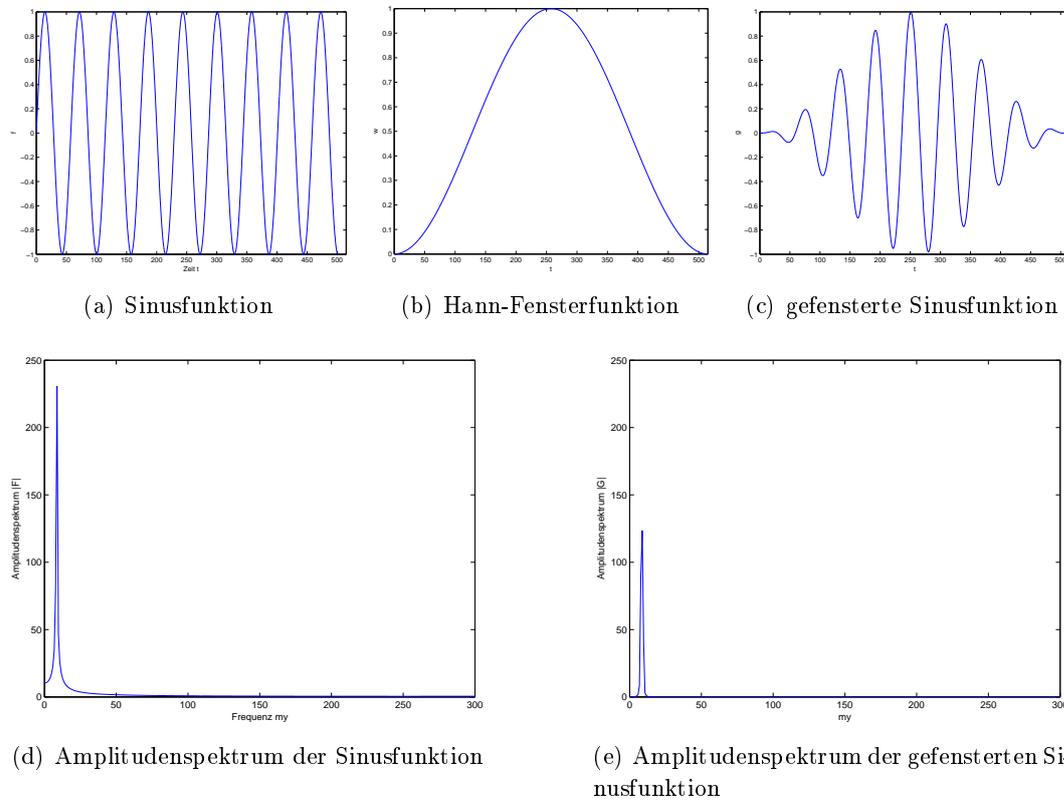


Abbildung 4.3: (a) Darstellung der Sinusfunktion, die aus einer ganzen Anzahl von Perioden besteht, (b) der Hannfensterfunktion und (c) der mit der Hann-Fensterfunktion gefensterten Sinusfunktion. (d) und (e) die Amplitudenspektren der Sinusfunktion und der gefensterten Sinusfunktion (vgl. [15]).

4.5 Gaborfilter

Unter Beachtung des Kapitels der Fouriertransformation 4.2 und des Kapitels der Waveletfunktionen 4.3 lässt sich wie in [19] ein weiteres Wavelet bilden. Ein solcher Filter ist der Gaborfilter oder das Gabor-Wavelet auch Kurzzeit-Fouriertransformation, gefensterte Fouriertransformation oder Gabortransformation genannt.

Definition 4.5 (Gabor-Wavelet [19])

Mit $x \in \mathbb{R}$ als freie Variable, x_0 eine bestimmte Position und $a \in \mathbb{R}$ ist das Gabor-Wavelet gegeben durch:

$$gw(x, \mu, x_0, a) = e^{-i2\pi\mu x} e^{-\left(\frac{x-x_0}{a}\right)^2}. \quad (4.1)$$

Lässt man den rechten Faktor in der Definition weg, erkennt man die Basisfunktion der Fouriertransformation wieder. Dieser rechte Faktor, der zu der Basisfunktion

multipliziert wird, ist eine Gaußsche Glockenkurve. Der Parameter a bestimmt dessen Breite.

Das Gabor-Wavelet kann in Real- und Imaginärteil aufgesplittet werden. Ein Beispiel eines Gabor-Wavelets ist in Abbildung 4.4 zu sehen. Daran erkennt man auch, dass der Realteil des Gabor-Wavelets eine gerade Funktion und der Imaginärteil eine ungerade Funktion ist.

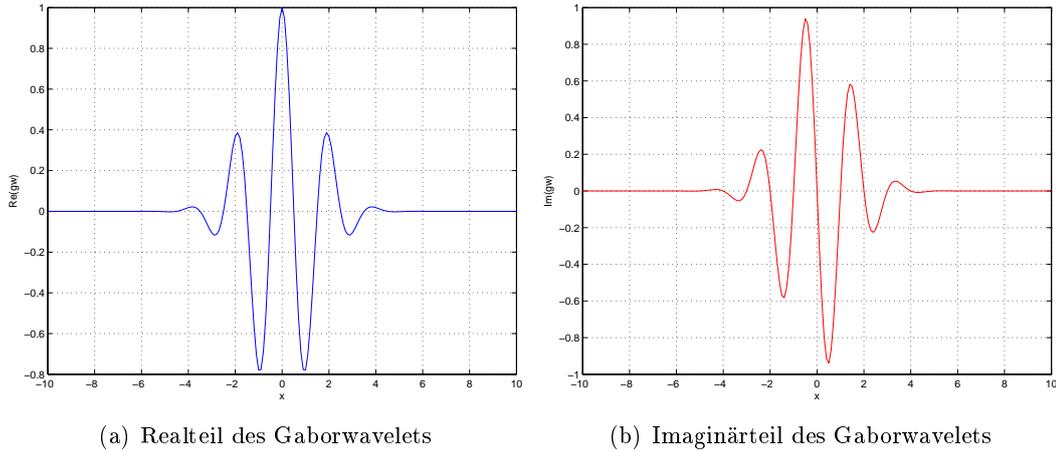


Abbildung 4.4: Beispiel eines Gabor-Wavelet-Paars nach Gleichung (4.1) mit $a = 0.5$, $x_0 = 0$ und $\mu = 0.5$. In (a) ist in blau der Realteil des Gabor-Waveletes dargestellt und in (b) in rot der Imaginärteil des Gabor-Waveletes (vgl. [19]).

4.6 Quadraturfilter

Wie in dem Kapitel 4.2 schon erwähnt, besitzt die Fouriertransformation die Eigenschaft der hermiteschen Symmetrie. Das bedeutet, dass für ein reellwertiges Bild $i(x, y)$ im Frequenzraum mit $\boldsymbol{\mu} = (\mu_x, \mu_y)$ folgendes gilt:

$$I(\boldsymbol{\mu}) = I^*(-\boldsymbol{\mu}).$$

Das wiederum heißt, dass die quadrierte Amplitude folgendermaßen geschrieben werden kann:

$$\begin{aligned} |I(\boldsymbol{\mu})|^2 &= |I^*(-\boldsymbol{\mu})|^2 = \sqrt{(Re[I(\boldsymbol{\mu})])^2 + (Im[I(\boldsymbol{\mu})])^2}^2 \\ &= (Re[I(\boldsymbol{\mu})])^2 + (Im[I(\boldsymbol{\mu})])^2. \end{aligned} \tag{4.2}$$

Diese Symmetrie wird nun im Folgenden ausgenutzt. Es genügt demnach ein Filter im Frequenzraum zu konstruieren, der das Signal im Frequenzbereich nur auf einer Seite filtert bzw. auf einer Seite unterdrückt.

Es soll nun ein Filter, der Quadraturfilter, vorgestellt werden, mit Hilfe dessen über

4 Filter im Frequenzraum

lokale Phasenunterschiede die Verschiebung zwischen zwei Bildern geschätzt werden kann. Dazu werden zunächst weitere Begrifflichkeiten benötigt.

Die Filter, die in den letzten Kapiteln vorgestellt wurden, sind größtenteils für die Kantenbestimmung geeignet. Das sind die einfachsten Strukturen in einer lokalen Umgebung, und zwar konstante Flächen bzw. Kanten. Es können sich allerdings auch Strukturen nur durch die Orientierung eines Musters vom Hintergrund unterscheiden. Eine lokale Umgebung mit lokaler Orientierung kennzeichnet sich dadurch aus, dass der Grauwert sich nur in einer Richtung ändert. Granlund und Knutsson haben in [7] den Begriff 'einfache Nachbarschaft' eingeführt, da die Grauwerte entlang von Linien konstant sind. Nun soll nach [10] eine Unterscheidung zwischen den Begriffen Orientierung und Richtung gemacht werden. Die Richtung ist definiert über den gesamten Winkelbereich des Intervalls $[0, 2\pi]$. Beispielsweise unterscheidet sich die Richtung zwischen zwei Vektoren, die in entgegengesetzte Richtungen zeigen. Anders ist das bei der Beschreibung einer lokalen Umgebung. Da ein um 180° gedrehtes Muster sich nicht vom nicht gedrehten unterscheidet, ist hier nur ein Winkelbereich von 180° nötig. In diesen Fällen spricht man von Orientierung. Eine andere Eigenschaft einfacher Nachbarschaften ist die Phase. Das Konzept der lokalen Phase im Eindimensionalen stammt von der Idee des analytischen Signals [7], welche durch die eindimensionale Hilberttransformation definiert ist.

4.6.1 Die Hilberttransformation

Die Hilberttransformation ist eine Abbildung, die eine Funktion des Ortsbereichs in eine andere Funktion ebenfalls im Ortsbereich überführt [7].

Definition 4.6 (Hilberttransformation)

Sei $f \in \mathbb{R}$ und x eine freie Variable. Die Hilberttransformation ist mit $\mathcal{H}i : \mathbb{R} \rightarrow \mathbb{R}$ definiert durch:

$$\mathcal{H}i(f) = f_{\mathcal{H}i}(x) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(\tau)}{\tau - x} d\tau. \quad (4.3)$$

Anmerkung 4.1

Wie man an der Gleichung (4.3) erkennen kann und in [7] auch erwähnt ist, ist die Hilberttransformation nicht für alle Funktionen im Ortsbereich wohldefiniert. Grund dafür ist, dass der Hilbertfilter unendlich ausgedehnt ist und es damit unmöglich ist, einen exakten Hilbertfilter für beliebige Signale mit einer kleinen Faltungsmaske zu konstruieren [10].

Betrachten wir die Definition der Hilberttransformation genauer, so erkennt man, dass die Hilberttransformation nichts anderes ist als die Faltung zwischen einem Signal f mit der Funktion $\frac{-1}{\pi x}$, mit \bullet als Platzhalter für den Definitionsbereich:

$$f_{\mathcal{H}i} = f * \frac{-1}{\pi \bullet}. \quad (4.4)$$

Erwähnenswert an dieser Stelle ist, dass das Signal und das analytische Signal eine Phasendifferenz von $\frac{\pi}{2}$ aufweist.

4.6.2 Das analytische Signal

Mit diesem Wissen kann nun das analytische Signal eingeführt werden. Dieses ist komplexwertig und wird durch den Zusammenschluss eines reellwertigen Signals und der dazugehörigen Hilberttransformierten gebildet [7].

Definition 4.7

Das analytische Signal $f_A : \mathbb{R} \rightarrow \mathbb{C}$, das zu dem reellen Signal f gehört, ist definiert als:

$$f_A = f - \mathbf{i}f_{\mathcal{H}i}. \quad (4.5)$$

Eine Eigenschaft des analytischen Signals ist die, dass es im Frequenzbereich aus rein positiven Frequenzen besteht. Dies wird im Weiteren auch für den Quadraturfilter wichtig sein und soll im Folgenden genauer dargestellt werden.

Das analytische Signal lässt sich folgendermaßen schreiben:

$$f_A = f * \left(\delta + \frac{\mathbf{i}}{\pi \bullet} \right).$$

Dies erhält man aus (4.5) durch folgende Umformungsschritte:

$$\begin{aligned} f_A = f - \mathbf{i}f_{\mathcal{H}i} &\stackrel{\text{nach Gleichung(4.4)}}{=} f - \mathbf{i} \left(f * \frac{-1}{\pi \bullet} \right) \stackrel{\text{nach Satz 3.1(iv)}}{=} f - \left(f * \frac{\mathbf{i}(-1)}{\pi \bullet} \right) \\ &= (f * \delta) - \left(f * \frac{\mathbf{i}(-1)}{\pi \bullet} \right) \stackrel{\text{nach Satz 3.1(iii)}}{=} f * \left(\delta - \frac{\mathbf{i}(-1)}{\pi \bullet} \right) \\ &= f * \left(\delta + \frac{\mathbf{i}}{\pi \bullet} \right). \end{aligned}$$

Mit Hilfe des Faltungstheorems und der Linearität der Fouriertransformation lässt sich nun aus dieser Form relativ einfach die Fouriertransformierte bilden.

Die Fouriertransformierte des Signals f sei F und die des Dirac-Signals ist 1 [14]. Die inverse Fouriertransformierte des Dirac-Signals ist ebenfalls 1 [14]. Die Funktion $\frac{\mathbf{i}}{\pi \bullet}$ hat als Fouriertransformierte die Signumfunktion sgn . Dies lässt sich am einfachsten nachweisen, indem man die inverse Fouriertransformierte der Signumfunktion bildet. Dies soll im Folgenden auf ähnlichem Weg geschehen wie in [1] die Fouriertransformierte der Heaviside-Funktion hergeleitet wird. Nach Kapitel 4.1.2 gilt:

$$\text{sgn}(\mu) = 2 \Theta(\mu) - 1.$$

Bilden der Ableitung auf beiden Seiten (die Ableitung der Signumfunktion sgn sei hier ohne Beweis gegeben durch den Dirac-Impuls δ):

$$\frac{\text{d}}{\text{d}\mu}(\text{sgn}(\mu)) = 2 \delta(\mu).$$

4 Filter im Frequenzraum

Bilden der inversen Fouriertransformation auf beiden Seiten:

$$\begin{aligned}\mathcal{F}^{-1}\left(\frac{d}{d\mu}(\text{sgn})\right)(\mu) &= 2\mathcal{F}^{-1}(\delta)(\mu) \\ \mathcal{F}^{-1}\left(\frac{d}{d\mu}(\text{sgn})\right)(\mu) &= 2 \cdot 1\end{aligned}$$

Durch Anwendung des Differentiationstheorems der Fouriertransformation (vgl. Satz 4.1 (vi)) erhält man:

$$\begin{aligned}-i2\pi \bullet \mathcal{F}^{-1}(\text{sgn})(\mu) &= 2 \\ \mathcal{F}^{-1}(\text{sgn})(\mu) &= \frac{2}{-i2\pi \bullet} \\ &= \frac{i}{\pi \bullet}.\end{aligned}$$

Damit ergibt sich für die Fouriertransformation von f_A mit Satz 4.1 (iv):

$$\begin{aligned}\mathcal{F}(f_A)(x) &= \mathcal{F}\left(f * \left(\delta + \frac{i}{\pi \bullet}\right)\right)(x) \\ &= \mathcal{F}(f)(x) \cdot \left(\mathcal{F}(\delta) + \mathcal{F}\left(\frac{i}{\pi \bullet}\right)\right)(x) \\ &= F(\mu) \cdot (1 + \text{sgn}(\mu)) \\ &= F(\mu) + F(\mu) \text{sgn}(\mu) \\ &= F(\mu) + F(\mu) (2\Theta(\mu) - 1) \\ &= 2F(\mu) \Theta(\mu) \\ &= \begin{cases} 0 & \text{für } \mu < 0 \\ F(\mu) & \text{für } \mu = 0 \\ 2F(\mu) & \text{für } \mu > 0 \end{cases}.\end{aligned}$$

Daran sieht man, dass die Fouriertransformierte des analytischen Signals im negativen Bereich Null ist. An dieser Stelle möchte ich noch einmal auf die lokale Phase zu sprechen kommen, die über das analytische Signal definiert ist. Wie auch in [3] beschrieben, ist die lokale Phase von der Fourierphase zu unterscheiden. Die Fourierphase wird wie in Kapitel 4.2 dargestellt berechnet. Die lokale Phase bezieht sich auf die Nachbarschaft eines Signals für eine gegebene Position und liefert Informationen über die lokale Struktur des Signals. Berechnet wird die lokale Phase über das Argument des analytischen Signals. Die lokale Phase werden wir auch im Morphon-Algorithmus zur Bestimmung der Verschiebungsschätzung verwenden. Ausgehend vom analytischen Signal kann der Quadraturfilter nun eingeführt werden. Nach [5] stehen zwei Signale in Quadratur, wenn das eine Signal die Hilberttransformierte des anderen Signals ist. Das heißt, ein analytisches Signal besteht aus zwei zueinander in Quadratur stehenden Signalen.

Nun möchte ich den Quadraturfilter einführen und danach auf diese Eigenschaft zurück kommen.

Mit dem Quadraturfilter erhält man, genau wie mit der Hilberttransformation, ein Signalpaar, das sich nur durch eine Phasenverschiebung von 90° unterscheidet [10]. Wie in [10] beschrieben, ist der Quadraturfilter eine andere Möglichkeit, wie man aus der Betrachtung von einfachen Nachbarschaften die lokale Orientierung und die lokale Wellenzahl bestimmen kann. Die Idee ist, Strukturen in einem bestimmten Bereich von Wellenzahlen und Richtungen zu extrahieren. Zur Bestimmung der lokalen Orientierung müssen mehrere gerichtete Filter angewendet werden. Dabei besitzt jeder Filter eine andere Orientierung, sodass jeder Filter für Strukturen unterschiedlicher Orientierung empfindlich ist. Die Filterantworten werden verglichen und man erhält eine maximale Filterantwort von dem Richtungsfilter, dessen Richtung am besten mit der lokalen Orientierung übereinstimmt. Erhält man von einem Filter eine starke Antwort im Vergleich zu den übrigen Filtern, dann enthält die Nachbarschaft ein lokal orientiertes Muster. Sind die Filterantworten alle ähnlich stark ausgeprägt, dann besteht die Nachbarschaft aus einer Verteilung unterschiedlich gerichteter Muster.

Diese Idee scheint im ersten Moment einfach und einleuchtend zu sein, es kommen jedoch im nächsten Moment einige Fragen auf. Wie sollte so ein Richtungsfilter aussehen und was für Eigenschaften sollte er für eine Bestimmung der lokalen Orientierung haben? Wie viele Filter sind notwendig, um ein möglichst genaues, aber dennoch zeitlich effizientes Ergebnis zu erhalten? Mit diesen Fragen beschäftigten sich Granlund und Knutsson in [7]. Sie stellten, wie in [8] bezeichnet, einen polar separierten Filter auf. Das ist ein Filter, der aus zwei verschiedenen Komponenten besteht, einer Winkelfunktion D und einer radialen Funktion R . Die Winkelfunktion bezieht sich auf die Orientierung des Filters und somit auf die lokale Orientierung der Nachbarschaft. Die radiale Funktion ist für die Bestimmung der Wellenzahl zuständig. Die polare Separierbarkeit des Filters vereinfacht die Veränderung von voneinander unabhängigen Aspekten.

Im Frequenzbereich sieht der Quadraturfilter folgendermaßen aus:

Definition 4.8 (Quadraturfilter [7])

Der Quadraturfilter F im Frequenzbereich mit der Richtungsfunktion (Winkelfunktion) D , der radialen Funktion R und $\boldsymbol{\mu}$ als freie Variable im Frequenzbereich, ist gegeben durch $Q_k : \mathbb{R}^d \rightarrow \mathbb{R}^d$ mit d als die Größe der Filtermaske und

$$Q_k = R(\|\boldsymbol{\mu}\|)D_k(\boldsymbol{\mu}), \tag{4.6}$$

mit

$$R(\|\boldsymbol{\mu}\|) = e^{-C_B \ln^2(\frac{\|\boldsymbol{\mu}\|}{\rho})} \quad \text{und} \quad C_B = \frac{4}{B^2 \ln(2)}. \tag{4.7}$$

4 Filter im Frequenzraum

Dabei bezeichnet $\|\boldsymbol{\mu}\|$ die Norm des Vektors $\boldsymbol{\mu}$, B die relative Bandbreite der Wellenzahl in Oktaven und ρ die Mittenfrequenz.

Außerdem ist die Richtungsfunktion durch

$$D_k(\boldsymbol{\mu}) = \begin{cases} (\boldsymbol{\mu}^T \hat{\boldsymbol{n}}_k)^2 & \text{für } \boldsymbol{\mu}^T \hat{\boldsymbol{n}}_k > 0 \\ 0 & \text{sonst} \end{cases} \quad (4.8)$$

mit $\hat{\boldsymbol{n}}_k$ als Richtungsvektor des Filters F_k gegeben. Dabei gibt der Index $k = [1, 2, 3, \dots, K]$ die jeweilige Filterrichtung an, wobei mit $K \in \mathbb{N}$ die Anzahl der Filterrichtungen beschrieben wird.

An dieser Stelle möchte ich genauer auf die Filterrichtungen eingehen. In [12] steht folgende Formel für die Anzahl der Richtungen, die benötigt werden:

$$K \geq \frac{1}{2}d(d+1),$$

dabei bezeichnet d die Dimension des eingehenden Signals. Diese Formel stimmt mit den Überlegungen in [7] für die Wahl der Richtungen überein und hat sich auch in der Implementierung von Daniel Forsberg als sinnvoll erwiesen. Knutsson nennt in [7] die Richtungen $22.5^\circ, 67.7^\circ, 112.5^\circ$ und 157.5° eine gute Wahl. Diese Richtungen werden wir später in einem Anwendungsbeispiel des Quadraturfilters wiederfinden.

Der Quadraturfilter wird als logarithmisch normalverteilt bezeichnet, da die radiale Funktion eine logarithmisch normalverteilte Form hat. D. h. auf einer logarithmischen Wellenzahlskala hat der Filter die Form einer Gaußfunktion. Mit folgenden genaueren Angaben zu der relativen Bandbreite B kann die radiale Funktion dargestellt werden: $B = \frac{1}{\ln 2} \ln \left(\frac{\rho_u}{\rho_l} \right)$, wobei mit ρ_l und ρ_u die Werte von ρ festgelegt sind, für die $R(\rho) = 0.5$ ist (siehe Abbildung 4.5(a)). Was mit unterschiedlichen Werten von ρ mit der radialen Funktion geschieht, kann man in Abbildung 4.5(b) sehen.

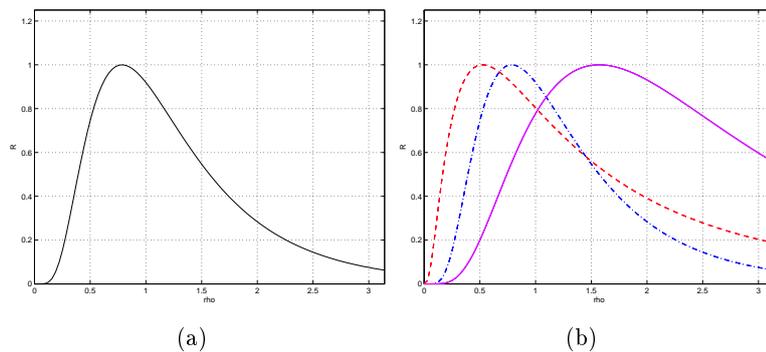


Abbildung 4.5: Darstellung der radialen Funktion des Quadraturfilters nach (4.7). In (a) mit $\rho = \frac{\pi}{8}$. In (b) drei weitere unterschiedliche radiale Funktionen mit $\rho = \frac{\pi}{6}$ (‘-’, rot), $\rho = \frac{\pi}{4}$ (‘-’, blau) und $\rho = \frac{\pi}{2}$ (‘-’, magenta) (vgl. [7]).

Betrachten wir nun noch einmal Gleichung (4.2). In dieser Gleichung sieht man, dass sich die Fouriertransformierte in Real- und Imaginärteil aufspalten lässt. Da der Quadraturfilter im Frequenzbereich definiert ist, ist diese Aufspaltung auch mit diesem möglich. Der Realteil entspricht einer geraden Funktion, der Imaginärteil einer ungeraden Funktion. Es lässt sich hier ein Vergleich zu dem Gabor-Wavelet ziehen (siehe Kapitel 4.5), welches als eine spezielle Art eines Quadraturfilters angesehen werden kann. Bezogen auf eine Struktur in einem Bild, entspricht eine gerade Funktion einer Linie und eine ungerade Funktion einer Kante (s. Kapitel 2.2). Nach [8] kann der Quadraturfilter folgendermaßen in geraden und ungeraden Anteil aufgeteilt werden. Der Index e bezeichnet den geraden Teil (engl.: even), der Index o den ungeraden Teil (engl.: odd):

$$\begin{aligned} Q_k(\boldsymbol{\mu}) &= Q_{ke}(\boldsymbol{\mu}) + Q_{ko}(\boldsymbol{\mu}) \\ Q_{ke}(\boldsymbol{\mu}) &= Q_{ke}(-\boldsymbol{\mu}) = \frac{1}{2}R(\|\boldsymbol{\mu}\|)(\boldsymbol{\mu}^T \hat{\mathbf{n}}_k)^2 \\ Q_{ko}(\boldsymbol{\mu}) &= -Q_{ko}(-\boldsymbol{\mu}) = \frac{1}{2}R(\|\boldsymbol{\mu}\|)(\boldsymbol{\mu}^T \hat{\mathbf{n}}_k)^2 \operatorname{sgn}(\boldsymbol{\mu}^T \hat{\mathbf{n}}_k) \end{aligned}$$

An dieser Stelle soll nun der Zusammenhang zwischen dem Quadraturfilter und dem analytischen Signal hergestellt werden. Wenn nun der Quadraturfilter bzw. dessen beide Komponenten $Q_{ke}(\boldsymbol{\mu})$ und $Q_{ko}(\boldsymbol{\mu})$ mit Hilfe der inversen Fouriertransformation in den Ortsbereich transformiert werden, erhält man $q_{ke}(x)$ und $q_{ko}(x)$. Diese beiden können auf folgende Weise zusammengefügt werden

$$q_k(x) = q_{ke}(x) - i q_{ko}(x). \quad (4.9)$$

Dabei erkennt man sofort die Beziehung zum analytischen Signal. Vergleicht man die Gleichung (4.9) mit Gleichung (4.5), kann $q_{ke}(x)$ als die Ausgangsfunktion und $q_{ko}(x)$ als dessen Hilberttransformierte angesehen werden.

Die Filterantworten $q_{ke}(x)$ und $q_{ko}(x)$ lassen sich nun auf ein Bild $b(x, y)$ im Ortsbereich anwenden:

$$\begin{aligned} q_{b,ke}(x, y) &= q_{ke}(x, y) * b(x, y), \\ q_{b,ko}(x, y) &= -i q_{ko}(x, y) * b(x, y). \end{aligned}$$

An dieser Stelle wird die Besonderheit des Quadraturfilters klar. Wir haben ihn zu Beginn im Frequenzraum definiert und ihm dort bestimmte Eigenschaften über die radiale Funktion R und die Richtungsfunktion D zugeschrieben. Für die Anwendung des Quadraturfilters auf ein Signal, beispielsweise ein Bild, wird der Quadraturfilter zuerst über die inverse Fouriertransformation in den Ortsraum transformiert, um ihn anschließend einfach über eine Faltung auf das gewünschte Signal anwenden zu können.

4.6.3 Beispiel: Der Quadraturfilter zur Kantenerkennung

In diesem Abschnitt möchte ich anhand eines einfachen Beispiels die Kantenerkennung des Quadraturfilters aufzeigen. Das Bild, in dem die Kanten erkannt werden sollen, ist in Abbildung 4.6 dargestellt und ist von mir in GIMP erstellt worden. Man sieht auf einem weißen Hintergrund zwei um 90° gedrehte schwarze Pfeile innerhalb eines kreisförmigen schwarzen Rahmens.



Abbildung 4.6: Beispielbild zur Kantenerkennung

Auf dieses Bild kann nun der von Daniel Forsberg implementierte Quadraturfilter angewandt werden. Wählt man die Filtermaske relativ groß, so lassen sich sehr schön die vier Filterrichtungen, die im vorherigen Abschnitt angesprochen wurden, erkennen. Dies ist in Abbildung 4.7 dargestellt. Es ist eine Filtermaske der Größe 11×11 verwendet worden.



(a) mit Filter in Richtung 157.5° gefiltertes Ausgangsbild



(b) mit Filter in Richtung 112.5° gefiltertes Ausgangsbild



(c) mit Filter in Richtung 67.7° gefiltertes Ausgangsbild



(d) mit Filter in Richtung 22.5° gefiltertes Ausgangsbild

Abbildung 4.7: Darstellung der vier Richtungen des Quadraturfilters mit großer Filtermaske

Verkleinert man die Filtermaske des Quadraturfilters, so lässt sich dieser mit anderen Filtern zur Kantenerkennung vergleichen. Dies möchte ich im Folgenden mit denen in Kapitel 3.3 beschriebenen einfachen Kantensfiltern im Ortsraum tun. Der Vergleich soll sich hier auf eine optische Gegenüberstellung beschränken. Diese ist in Abbildung 4.8 zu finden. In den Bildern (a) bis (d) sieht man jeweils eine Richtung des Quadraturfilters. Hier nun mit einer kleineren Filtermaske, und zwar der Größe 3×3 . Schaut man sich diese im Gesamten an, sieht man, dass der Quadraturfilter sehr gute Ergebnisse als Kantensfilter liefert. In den Bildern (e) bis (g) ist der Sobel-, Prewitt- und Laplacefilter auf das Ausgangsbild (Abbildung 4.6) angewandt worden. Für die Filterung mit dem Sobeloperator wurde eine horizontale Filtermaske verwendet. Die Filterung mit dem Prewittoperator wurde mit einer vertikalen Filtermaske durchgeführt. Für den Laplaceoperator wurde die in Kapitel 3.3.3 mit L_1 bezeichnete Filtermaske angewandt. Vergleicht man nun die mit dem Quadraturfilter gefilterten Bilder mit den Bildern, die mit Standardfiltern gefiltert wurden, sieht man, dass der Quadraturfilter sogar bessere Ergebnisse liefert. In denen mit Standardmasken gefilterten Bildern ist um die eigentlichen Kanten Rauschen zu sehen, was beim Quadraturfilter nicht der Fall ist.



(a) mit Filter in Richtung 157.5° gefiltertes Ausgangsbild



(b) mit Filter in Richtung 112.5° gefiltertes Ausgangsbild



(c) mit Filter in Richtung 67.7° gefiltertes Ausgangsbild



(d) mit Filter in Richtung 22.5° gefiltertes Ausgangsbild



(e) mit Sobelfilter gefiltertes Ausgangsbild



(f) mit Prewittfilter gefiltertes Ausgangsbild



(g) mit Laplacefilter gefiltertes Ausgangsbild

Abbildung 4.8: Direkter Vergleich verschiedener Kantenfiltern

5 Der Morphon-Algorithmus

In diesem Kapitel soll nun der Morphon-Algorithmus vorgestellt werden, in dem der Quadraturfilter verwendet wird.

Der Morphon-Algorithmus wurde erstmals von Knutsson und Andersson in [23] und [13] als Registrierungsalgorithmus vorgestellt. Die grundlegende Idee dieses Algorithmus steckt direkt in dem Titel des zweitgenannten Papers: 'Paint on Priors'. Das bedeutet, dass die Vorgänger (engl.: 'priors') den weiteren Verlauf des Algorithmus bestimmen, was damit genau gemeint ist, wird im Folgenden noch klarer werden. Der Morphon-Algorithmus kann zur nicht-parametrischen Registrierung gezählt werden [3]. Es wird versucht, iterativ ein Ausgangsbild in ein Zielbild umzuwandeln, indem das Ausgangsbild verändert wird. Dabei ist jeder Schritt maßgeblich an der Berechnung im zweiten Schritt beteiligt. Dieser Prozess kann, angelehnt an das Grundgerüst einer nicht-parametrischen Registrierung, folgendermaßen unterteilt werden:

1. **Verschiebungsschätzung**

Ziel dieses Schrittes ist es, Anzeichen zu finden, wie das Ausgangsbild verändert werden muss, damit es dem Zielbild ähnlicher wird.

2. **Bestimmung des Deformationsfeldes**

Das Deformationsfeld ist ein Feld, in dem die Verformungen der vorherigen Iterationen akkumuliert werden. Zum einen wird hier das Deformationsfeld aktualisiert, zum anderen findet hier die Regularisierung des Deformationsfeldes statt.

3. **Deformation**

In diesem Schritt wird das ursprüngliche Ausgangsbild entsprechend des Deformationsfeldes transformiert.

Diese drei Schritte sollen nach [23] so lange wiederholt werden, bis im ersten Schritt angezeigt wird, dass weitere Umformungen keine Verbesserungen mehr bringen würden. Nach Gesprächen mit Lennart Tautz und der Implementierung von Daniel Forsberg wird ersichtlich, dass in der Praxis zwischen 3 und 5 Iterationen durchgeführt werden. Dieser Algorithmus beruht demnach darauf, dass immer die Vorgänger das Verhalten des nächsten Umformungsschrittes bestimmen.

5.1 Verschiebungsschätzung

Die Bestimmung der Ausrichtung oder der Verschiebung zweier Bilder benötigt im Grunde genommen den gleichen Lösungsansatz. Hier werden die Quadraturphasenunterschiede benutzt, um die Verschiebung zwischen zwei Bildern zu bestimmen. Quadraturphasen können als ein Maß von lokalen Strukturen beschrieben werden, wie z. B. Kanten zwischen dunklen und hellen Bereichen oder helle Linien auf dunklem

5 Der Morphon-Algorithmus

Hintergrund [23]. Somit ist der Unterschied zwischen einer lokalen Phase des Ausgangsbildes und der des Zielbildes ein gutes Maß dafür, wie stark das Ausgangsbild noch bewegt werden muss.

Zur Berechnung des Verschiebungsschätzers wird ein Quadraturfilter verwendet. Mit Hilfe der Filterantwort lässt sich der Verschiebungsschätzer d_k einer bestimmten Richtung bestimmen, darüber hinaus gibt er eine Amplitude zurück, mit der das Sicherheitsmaß bestimmt werden kann. Ist die Amplitude groß, so ist eine höhere Sicherheit gewährleistet, dass eine lokale Struktur (und nicht nur Rauschen etc.) im Bild gefunden wurde und diese nun bewegt werden soll (siehe auch Kapitel 5.2).

Im Folgenden möchte ich den Morphon-Algorithmus, wie er auch in [20] oder [3] beschrieben ist, darstellen.

Das Ausgangsbild soll hier als $I_R(\mathbf{x})$ bezeichnet werden. Das Zielbild unterscheidet sich vom Ausgangsbild durch ein unbekanntes Deformationsfeld $\mathbf{d}(\mathbf{x})$, das im Folgenden geschätzt werden soll. Demnach lässt sich das Zielbild als $I_T(\mathbf{x}) = I_R(\mathbf{x} + \mathbf{d}(\mathbf{x}))$ beschreiben. Zuerst wird der Quadraturfilter in jede Richtung jeweils auf das Referenzbild und das Zielbild angewandt. Der Index k gibt dabei die jeweilige Filterrichtung an:

$$\begin{aligned} q_{R,k}(\mathbf{x}) &= (I_R * q_k)(\mathbf{x}), \\ q_{T,k}(\mathbf{x}) &= (I_T * q_k)(\mathbf{x}). \end{aligned}$$

Um die lokale Phasendifferenz zwischen den Bildern berechnen zu können, wird zuerst das komplex konjugierte Produkt der Quadraturfilterantworten bestimmt:

$$Q_k(\mathbf{x}) = q_{R,k}(\mathbf{x}) \cdot q_{T,k}^*(\mathbf{x}).$$

Die Verschiebung $d_k(\mathbf{x})$ entlang der Orientierung $\hat{\mathbf{n}}_k$ und damit die lokale Phasendifferenz kann über das Argument des eben berechneten Produkts bestimmt werden:

$$d_k(\mathbf{x}) = \arg(Q_k(\mathbf{x})).$$

Die lokale Phase, die die Struktur in einem Bild beschreibt, ist ziemlich anfällig gegenüber Rauschen. Deswegen erfolgt eine Verknüpfung mit einem Sicherheitsmaß $c_k(\mathbf{x})$, das in jede Richtung bestimmt wird. Dieses wird mit der Magnitude des komplex konjugierten Produkts $Q_k(\mathbf{x})$ berechnet. Dies kann um einen weiteren Faktor ergänzt werden, um sicherzugehen, dass die geschätzte Phasendifferenz nur in den relevanten Regionen mit einer kleinen Phasendifferenz berücksichtigt wird:

$$c_k(\mathbf{x}) = \sqrt{|Q_k(\mathbf{x})|} \cos^2\left(\frac{d_k(\mathbf{x})}{2}\right).$$

Ein gesamtes Sicherheitsmaß über alle Richtungen lässt sich über die Summe der einzelnen Sicherheitsmaße bestimmen:

$$c_i(\mathbf{x}) = \sum_{k=1}^4 c_k(\mathbf{x}). \quad (5.1)$$

Die Überlegung, die hinter dieser Berechnung des Sicherheitsmaßes steht, ist die, dass die Quadraturfilterantwort bei Kanten und Linien groß ist [23]. Werden demnach im Ausgangs- und Zielbild ähnliche Strukturen erkannt, so ist die jeweilige Filterantwort groß und dementsprechend auch das Produkt aus den beiden. Wenn ein Filter eine andere Struktur sieht, wird das Produkt folglich kleiner und somit ist die Schätzung weniger sicher. Damit kann davon ausgegangen werden, dass, je größer die Magnitude ist, desto höher ist die Sicherheit, dass in den Bildern Strukturen gefunden wurden [20].

Die einfachste Möglichkeit das gesamte Deformationsfeld zu bestimmen, ist die Berechnung über den gewichteten Mittelwert. Dabei werden die Verschiebungsschätzer mit ihren zugehörigen Sicherheitsmaßen gewichtet. Der Parameter $\hat{\mathbf{n}}_k$ bezeichnet hier wiederum die Filterrichtung:

$$\mathbf{d}(\mathbf{x}) = \frac{\sum_{k=1}^4 c_k(\mathbf{x}) d_k(\mathbf{x}) \hat{\mathbf{n}}_k}{\sum_{k=1}^4 c_k(\mathbf{x})}. \quad (5.2)$$

Da der Morphon-Algorithmus ein iterativer Algorithmus ist, wird in jedem Schritt das Deformationsfeld neu berechnet. Aus diesem Grund soll das Deformationsfeld im Folgenden mit $\mathbf{d}_i(\mathbf{x})$ bezeichnet werden, wobei i für 'incremental' also 'schrittweise verändernd' steht.

5.2 Bestimmung des Deformationsfeldes

In jedem Schritt des Morphon-Algorithmus wird ein Deformationsfeld $\mathbf{d}_i(\mathbf{x})$ bestimmt, welches mit dem Deformationsfeld der vorherigen Iteration zusammengefügt wird. Das daraus resultierende Deformationsfeld wird als $\mathbf{d}_a(\mathbf{x})$ bezeichnet, wobei a für 'accumulated' also 'angesammelt' bzw. 'angehäuft' steht. Das Deformationsfeld $\mathbf{d}_a(\mathbf{x})$ wird mit Hilfe des in der vorherigen Iteration berechneten Deformationsfeldes $\mathbf{d}_a(\mathbf{x})$, des Deformationsfeldes der derzeitigen Iteration $\mathbf{d}_i(\mathbf{x})$ (siehe Gleichung (5.2)) und des Sicherheitsmaßes berechnet:

$$\mathbf{d}_a(\mathbf{x}) = \frac{c_a(\mathbf{x})\mathbf{d}_a(\mathbf{x}) + c_i(\mathbf{x})[\mathbf{d}_a(\mathbf{x}) + \mathbf{d}_i(\mathbf{x})]}{c_a(\mathbf{x}) + c_i(\mathbf{x})}. \quad (5.3)$$

In jeder Iteration wird dem Deformationsfeld durch das Sicherheitsmaß $c_i(\mathbf{x})$ eine entsprechende Gewichtung zugeteilt. Dieses Sicherheitsmaß wird, ähnlich wie das Deformationsfeld, mit dem Sicherheitsmaß der vorherigen Iteration akkumuliert. Dies erfolgt in [3] und [20] auf folgende Weise:

$$c_a(\mathbf{x}) = \frac{c_a(\mathbf{x})^2 + c_i(\mathbf{x})^2}{c_a(\mathbf{x}) + c_i(\mathbf{x})}. \quad (5.4)$$

Um Fehler gering zu halten, erfolgt in jeder Iteration eine Regularisierung des Deformationsfeldes. Dies geschieht, indem dieses z. B. mit der Gaußfunktion g geglättet

5 Der Morphon-Algorithmus

wird:

$$\mathbf{d}_i(\mathbf{x}) = \frac{c_i(\mathbf{x})\mathbf{d}_i(\mathbf{x}) * g(\mathbf{x})}{c_i(\mathbf{x}) * g(\mathbf{x})}. \quad (5.5)$$

5.3 Deformation

Mit Hilfe des Deformationsfeldes wird eine veränderte Version des Ausgangsbildes erzeugt. Um Fehler zu vermeiden, die entstehen, wenn das Ausgangsbild in jeder Iteration verändert wird, wird jeweils das originale Ausgangsbild entsprechend des akkumulierten Deformationsfeldes verändert und mit dem Zielbild verglichen.

6 Beispiel: Anwendung des Morphon-Algorithmus

In diesem Kapitel möchte ich, unter Zuhilfenahme der Implementierung von Daniel Forsberg [4], ein Beispiel zeigen, in dem der Morphon-Algorithmus erfolgreich angewandt wurde. Die Bilder, an denen der Algorithmus getestet worden ist, stammen aus dem in MATLAB befindlichen MRI-Datensatz. Aus diesem Datensatz wurde nun ein Bild geladen, das als Ausgangsbild fungieren soll (siehe Abbildung 6.1(a)). Auf dieses Bild wurde eine Transformation angewandt, um ein Zielbild zu erhalten, das dennoch mit dem Ausgangsbild vergleichbar ist (siehe Abbildung 6.1(c)). Nun soll auf diese beiden Bilder der Morphon-Algorithmus angewandt werden. Ziel ist es, das Ausgangsbild so zu verändern, dass es dem Zielbild ähnlich wird. In Abbildung 6.1(d) sieht man durch Übereinanderlegung den Unterschied zwischen dem Ausgangsbild und dem Zielbild, der berechnet werden soll. Wie in Kapitel 5 dargestellt, werden die Schritte Verschiebungsschätzung, Bestimmung eines Deformationsfeldes und Deformation durchgeführt. Es wurden hier drei Iterationen des Algorithmus vorgenommen. Nach diesen erhält man das in Abbildung 6.1(b) dargestellte Bild. Man sieht, dass der Algorithmus sogar für elastische Verformungen des Zielbildes ein sehr gutes Ergebnis liefert, da das deformierte Ausgangsbild ziemlich genau dem Zielbild ähnelt. Des Weiteren ist in Abbildung 6.1(e) das akkumulierte Deformationsfeld zu sehen. Die Pfeile geben hierbei die Richtung und Stärke der Verschiebung in den jeweiligen Bereichen an. In Abbildung 6.1(f) ist das Gitter zu sehen, nachdem die Deformationen durchgeführt wurden. Hier sieht man sehr gut, wie die Gitterpunkte verschoben wurden, um aus dem Ausgangsbild das Zielbild zu erhalten.

Wie bereits angesprochen, ist die Implementierung des Morphon-Algorithmus von Daniel Forsberg [4] entnommen und ich habe das eben beschriebene Beispiel generiert. Zudem habe ich einen eigenen Quadraturfilter implementiert, der stark auf dem in Kapitel 4.6 beschriebenen Quadraturfilter basiert. Dieser besitzt jedoch allem Anschein nach nicht die optimale Parameterwahl, weswegen dieser in Kapitel 4.6.3 zur Kantenerkennung nicht in Frage kam. Allerdings verfügt dieser über die grundlegenden richtigen Eigenschaften, da definitiv Tendenzen zur Erkennung der Kanten im Bild vorhanden waren. Aus diesem Grund habe ich meinen Quadraturfilter in den von Daniel Forsberg implementierten Morphon-Algorithmus eingefügt und anschließend auf diesen verschiedene Ausgangs- und Zielbilder angewandt. Dabei ist mir aufgefallen, dass sich gute Ergebnisse ergeben, wenn das Ausgangs- und das Zielbild sich nur durch eine Translation oder affin lineare Transformation unterscheiden. Unterscheiden sich Ausgangs- und Zielbild um eine Transformation mit mehr Freiheitsgraden, lässt sich der Morphon-Algorithmus jedoch nicht mehr durchführen. Dies liegt womöglich an der Wahl des Parameters μ , den ich für meinen Quadraturfilter äquidistant gewählt habe. In der Implementierung von Daniel Forsberg hingegen wurden einige weitere Berechnungen durchgeführt, um das Gitter für μ optimal zu wählen.

6 Beispiel: Anwendung des Morphon-Algorithmus

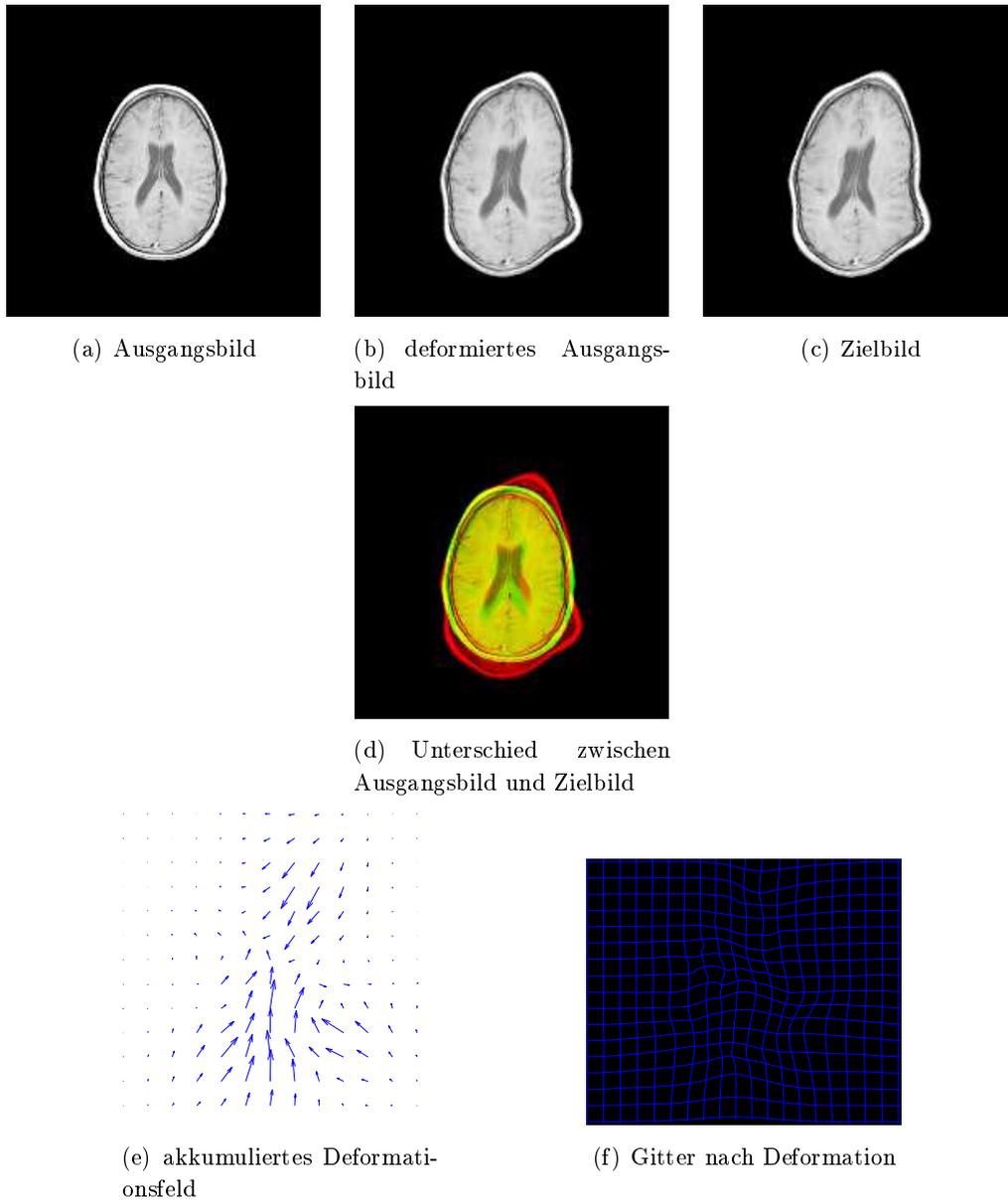


Abbildung 6.1: Ergebnis nach Anwendung des Morphon-Algorithmus

7 Fazit

In dieser Arbeit wurde ein spezieller Registrierungsalgorithmus, der Morphon-Algorithmus, vorgestellt, dessen Besonderheit in dem von ihm verwendeten Filter, dem Quadraturfilter, liegt. Dafür wurden Schritt für Schritt die Grundlagen eingeführt, um Zugang zu dem Quadraturfilter zu erhalten. Dieser wurde als Kantenerkennungsfiler dargestellt und in den Morphon-Algorithmus eingefügt.

Durch die Implementierung des Quadraturfilters ist mir aufgefallen, dass der, im ersten Moment relativ einfach aussehende Filter, doch einige Tücken aufweist, wie zum Beispiel bei der Wahl des Parameters μ . Dabei ist im Morphon-Algorithmus aufgefallen, dass dieser Parameter eine große Rolle spielt. Denn erst durch dessen richtige Wahl sind dort elastische Transformationen möglich geworden. Leider ist es mir nicht möglich gewesen, den Quadraturfilter optimal und den Morphon-Algorithmus komplett selbst zu implementieren. Dennoch halte ich es für eine gute Entscheidung die Implementierung von Daniel Forsberg zur Hand genommen zu haben. Denn erst dadurch konnte ich die Erfolge meines eigenen Quadraturfilters sehen und aufzeigen und habe noch weitere Schwierigkeiten der Implementierung erkannt. So bleibt neben der Wahl des Parameters μ noch offen, wie eine optimale radiale Funktion des Quadraturfilters aussieht, denn wie wir in Kapitel 4.6 gesehen haben, kann dieser durch unterschiedliche Parameterwahl verschieden gewählt werden.

Zusammengefasst lässt sich zum Quadraturfilter sagen, dass es schwierig war, Zugang zu ihm zu finden und zu verstehen, was sich genau hinter diesem Filter verbirgt. Zudem gab es keine Anhaltspunkte, wie bestimmte Parameter am besten gewählt werden sollten und somit mein selbst implementierter Quadraturfilter zwar die gewünschten Tendenzen aufzeigt, jedoch nicht optimal ist. Dennoch ergeben sich für diesen Filter viele Anwendungsmöglichkeiten. Gerade aufgrund seiner speziellen Eigenschaft der polaren Separierbarkeit ist dieser auf verschiedene Aufgaben individuell anpassbar. Der Morphon-Algorithmus ist ein sehr zugänglicher Algorithmus. Die iterative Bestimmung eines Deformationsfeldes und eines Sicherheitsmaßfeldes, wobei in jedem Schritt eine weitere Verbesserung gesucht wird, macht diesen zu einem relativ anschaulichen Algorithmus. Das Anwendungsbeispiel des Morphon-Algorithmus zeigt, dass dieser nicht nur auf selbst erzeugte Beispiele, sondern auch auf medizinische Bilder angewandt werden kann.

Literaturverzeichnis

- [1] BURROWS, B. L. und D. J. COLWELL: *The Fourier transform of the unit step function*. International Journal of Mathematical Education in Science and Technology, 21(4):629–635, 1990.
- [2] DORN, H.-J., H. FREUDIGMANN, M. HERBST, G. REINELT, M. SCHWIER, P. SIEBERT und S. STÖTZER: *Formelsammlung Mathematik: Gymnasium*. Ernst Klett, Stuttgart, 2005.
- [3] FORSBERG, D.: *Robust Image Registration for Improved Clinical Efficiency: Using Local Structure Analysis and Model-Based Processing*. Dissertation, Linköping University, 2013.
- [4] FORSBERG, D.: *fordanic/image registration*. <http://www.mathworks.com/matlabcentral/fileexchange/50702-fordanic-image-registration>, 2015. [Stand: 02.11.2015].
- [5] GABOR, D.: *Theory of communication. Part 1: The analysis of information*. Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering, 93(26):429–441, 1946.
- [6] GONZALES, R. C. und R. E. WOODS: *Digital Image Processing*. Pearson Education Inc., New Jersey, 3. Auflage, 2008.
- [7] GRANLUND, G. H. und H. KNUTSSON: *Signal Processing for Computer Vision*. Kluwer Academic Publishers, Dordrecht, 1995.
- [8] HAGLUND, L.: *Adaptive multidimensional filtering*. Dissertation, Linköping University, 1991.
- [9] HOHLOCH, E., H. KÜMMERER, G. GLATZ, H. GRIEB und R. MOHR: *Fourier-Analysis*. Cornelsen, Berlin, 1996.
- [10] JÄHNE, B.: *Digitale Bildverarbeitung und Bildgewinnung*. Springer Vieweg, Heidelberg, 7. Auflage, 2012.
- [11] KATZNELSON, Y.: *An introduction to harmonic analysis*. Cambridge University Press, 2004.
- [12] KNUTSSON, H. und M. ANDERSSON: *Robust N-dimensional orientation estimation using quadrature filters and tensor whitening*. In: *Acoustics, Speech, and Signal Processing, 1994. ICASSP-94., 1994 IEEE International Conference on*, Seiten V–529. IEEE, 1994.

LITERATURVERZEICHNIS

- [13] KNUTSSON, H. und M. ANDERSSON: *Morphons: Paint on priors and elastic canvas for segmentation and registration*. In: *Image Analysis*, Seiten 292–301. Springer, 2005.
- [14] MERTINS, A.: *Signaltheorie*. Springer Vieweg, Wiesbaden, 3 Auflage, 2013.
- [15] MEYER, M.: *Signalverarbeitung Analoge und digitale Signale, Systeme und Filter*. Springer Vieweg, Wiesbaden, 7. Auflage, 2014.
- [16] MODERSITZKI, J.: *Numerical methods for image registration*. Oxford University Press, 2004.
- [17] MODERSITZKI, J.: *FAIR: flexible algorithms for image registration*, Band 6. SIAM, 2009.
- [18] NISCHWITZ, A., M. FISCHER, P. HABERÄCKER und G. SOCHER: *Computergrafik und Bildverarbeitung, Band II: Bildverarbeitung*. Vieweg+Teubner, Wiesbaden, 3. Auflage, 2011.
- [19] NIXON, M. S. und A. S. AGUADO: *Feature Extraction and Images Processing for Computer Vision*. Elsevier, London, 3. Auflage, 2012.
- [20] TAUTZ, L., A. HENNEMUTH und H.-O. PEITGEN: *Motion analysis with quadrature filter based registration of tagged MRI sequences*. In: *Statistical Atlases and Computational Models of the Heart. Imaging and Modelling Challenges*, Seiten 78–87. Springer, 2012.
- [21] TÖNNIES, K. D.: *Grundlagen der Bildverarbeitung*. Pearson Studium, München, 2005.
- [22] VONGRÜNIGEN, D. C.: *Digitale Signalverarbeitung mit einer Einführung in die kontinuierlichen Signale und Systeme*. Hanser, München, 5. Auflage, 2014.
- [23] WRANGSJÖ, A., J. PETTERSSON und H. KNUTSSON: *Non-rigid registration using morphons*. In: *Image Analysis*, Seiten 501–510. Springer, 2005.