

Gerichtete Haarwavelet-Systeme in der Bildverarbeitung

Von der Fakultät für Mathematik
der Universität Duisburg-Essen (Campus Duisburg)

zur Erlangung des akademischen Grades eines

Dr. rer. nat.

genehmigte Dissertation

von

Jens Krommweh

aus Duisburg



UNIVERSITÄT
D U I S B U R G
E S S E N

Referentin: Prof. Dr. Gerlind Plonka-Hoch
Korreferentin: Prof. Dr. Gabriele Steidl

Tag der mündlichen Prüfung: 27. April 2010

Inhaltsverzeichnis

Einleitung	1
1 Grundlagen der Waveletsysteme und ihrer Anwendungen	6
1.1 Funktionensysteme	6
1.2 Multiresolutionsanalyse und Wavelets	8
1.3 Filterbänke	19
1.4 Bildverarbeitung	25
2 Orientierte Haarwavelet-Systeme auf Dreiecken	30
2.1 Waveletbasen mit vier Richtungen	30
2.1.1 Konstruktion von Skalierungs- und Waveletfunktionen	31
2.1.2 Anwendung in der Bildverarbeitung	34
2.2 Waveletframes mit acht Richtungen	38
2.2.1 Konstruktion der Approximationsräume	38
2.2.2 Konstruktion der Waveleträume	43
2.2.3 Gerichtete Haarwavelet-Filterbank	47
2.2.4 Dünne Darstellung von Bildern in Waveleträumen	54
2.2.5 Anwendung in der Bildverarbeitung	61
2.3 Charakterisierung von Tight-Frames mittels der Polyphasen-Matrix	64
2.3.1 Die Extension Principles	65
2.3.2 Die Polyphasen-Matrix und die Tight-Frame-Eigenschaft	67
2.3.3 Zusammenhang zwischen Polyphasen- und Modulations-Darstellung	70
2.3.4 Polyphasen-Untersuchung des gerichteten Haarwavelet-Frames	71
2.4 Waveletframes mit beliebig vielen Richtungen	74
2.4.1 Konstruktion von Skalierungs- und Waveletfunktionen	74
2.4.2 Die Polyphasen-Matrix und die Tight-Frame-Eigenschaft	79
2.5 Ausblick: Stetige Wavelets auf Dreiecken	80
2.5.1 Konstruktion eines linearen Splinepaares	82
2.5.2 Konstruktion der linearen Spline-Wavelets	86

3	Adaptive Tetrolet-Transformation	93
3.1	Notation	94
3.2	Tetromino-Überdeckungen	95
3.3	Die Idee der Tetrolets	96
3.4	Eine Orthonormalbasis der Tetrolets	97
3.5	Detaillierte Beschreibung des Tetrolet-Filterbank-Algorithmus	98
3.6	Adaptivitätskosten: Modifikationen der Tetrolet-Transformation	103
3.7	Arithmetische Komplexität	105
3.8	Numerische Resultate	106
3.9	Multiresolutionsanalyse und Approximation	114
3.9.1	Notation	115
3.9.2	Adaptive Multiresolutionsanalyse der Tetrolets	117
3.9.3	Approximationseigenschaften	119
4	Postprocessing-Methode für gerichtete Haarwavelets	126
4.1	Methode des projizierten Subgradienten-Abstiegs	128
4.2	ATV-Minimierung	130
4.3	Anwendungen in der Bildapproximation	134
	Anhang	141
A	Konstruktion eines dualen Frames	141
B	Herleitungen zu Kapitel 2.5	144
B.1	Skalarprodukte der Splinefunktionen ϕ_1 und ϕ_2	144
B.2	Konstruktion der dualen Splinefunktionen $\tilde{\phi}_1$ und $\tilde{\phi}_2$ im Abschnitt 2.5.1	145
B.3	Konstruktion der Wavelet-Splinefunktionen im Abschnitt 2.5.2	148
	Literaturverzeichnis	149
	Abbildungsverzeichnis	156
	Tabellenverzeichnis	158

Einleitung

Motivation Die Wavelettransformation bietet ein vorzügliches Instrument, um eindimensionale Signale effektiv darzustellen. Das liegt darin begründet, dass Wavelets in der Lage sind, punktweise Singularitäten gut zu approximieren. Die Waveletapproximation ist damit wesentlich besser als die klassische Fourierapproximation.

Im Falle von mehrdimensionalen Datenmengen erweisen sich traditionelle Wavelets als weniger geeignet. Betrachten wir das idealisierte Bild f , das Hölder-stetig von der Ordnung $\alpha > 0$ ist. Dann können wir das Bild mithilfe eines Waveletsystems $\{\psi_\mu : \mu \in I\}$ durch die Partialsummen

$$f_M = \sum_{\mu \in I_M} \langle f, \psi_\mu \rangle \psi_\mu$$

approximieren, wobei I_M die Indexmenge der M größten Koeffizienten $|\langle f, \psi_\mu \rangle|$ ist. Verwenden wir hinreichend glatte Tensorprodukt-Wavelets, so besitzt der Approximationsfehler ϵ_M der besten M -Term-Approximation f_M das optimale, asymptotische Abklingverhalten der Ordnung $\mathcal{O}(M^{-\alpha})$, d. h. für ϵ_M gilt

$$\epsilon_M := \|f - f_M\|_2^2 = \sum_{\mu \notin I_M} |\langle f, \psi_\mu \rangle|^2 \sim \mathcal{O}(M^{-\alpha}).$$

In der Regel besitzen Bilder jedoch Bildkanten, die verschiedene Bildgebiete voneinander abgrenzen. Die Existenz solcher Unstetigkeiten verschlechtert die Approximationsqualität drastisch, unabhängig davon, wie glatt die Bildfunktion innerhalb einzelner Bildregionen ist. Für Unstetigkeiten endlicher Länge kann im Fall $\alpha \geq 1$ nur noch ein Approximationsverhalten der Ordnung $\mathcal{O}(M^{-1})$ garantiert werden. Damit ist die Waveletapproximation auch im zweidimensionalen Fall besser als die Fouriertransformation, dessen Approximationsfehler lediglich mit der Ordnung $\mathcal{O}(M^{-1/2})$ sinkt [Mal99, Kapitel 9.2], aber weit von der optimalen Approximationseigenschaft $\mathcal{O}(M^{-\alpha})$ entfernt. Dies liegt an der unzureichenden Behandlung der gerichteten Bildkanten. In den letzten Jahren wurden verschiedene Waveletsysteme vorgeschlagen, die aufgrund ihrer Richtungseigenschaften in der Lage sind, Bildkanten besser zu adaptieren.

Zusammenhang zu anderen Arbeiten Im Wesentlichen gibt es zwei verschiedene Ansätze in der gerichteten Bildverarbeitung, die adaptiven und die nicht-adaptiven Verfahren. Letz-

tere basieren auf einem a priori gewählten Waveletsystem, das eine Basis oder einen Frame des Raumes $L^2(\mathbb{R}^2)$ generiert. Die Basis- bzw. Framefunktionen werden – wie gewöhnlich – skaliert und translatiert, aber darüber hinaus werden sie einem weiteren, geometrischen Operator (wie z. B. der Rotation oder der Scherung) unterzogen, der den Funktionen eine Orientierung verleiht. Dadurch erhöht sich in der Regel die Redundanz der Waveletsysteme, was eine „oversampled“ Filterbank nach sich zieht. Bei der Klasse der adaptiven Verfahren erhalten die Wavelets ihre Richtungssensitivität, indem sie entsprechend der jeweiligen Bildgeometrie definiert werden. Das führt dazu, dass viel weniger Wavelets verwendet werden müssen, allerdings entstehen zusätzliche Adaptivitätskosten, die für die Rekonstruktion des Bildes vonnöten sind.

An dieser Stelle geben wir einen kurzen Überblick über verschiedene richtungsorientierte Waveletsysteme, die sich in den letzten Jahren als sehr erfolgreich erwiesen haben und für unsere Arbeit relevant sind. In den folgenden Kapiteln unserer Arbeit gehen wir dann an den entsprechenden Stellen detaillierter auf die Zusammenhänge zwischen unseren Methoden und den schon bekannten Verfahren ein.

Die vielleicht populärsten Vertreter der nicht-adaptiven Verfahren sind die *Curvelets*, die erstmals 1999 von Candès und Donoho in [CD00] vorgeschlagen wurden und später etwas modifiziert wurden [CD04]. Curvelets werden als glatte Funktionen mit kompaktem Träger auf Kreisscheibensegmenten im Frequenzbereich definiert, die ihre Richtungseigenschaften mittels einer Rotation erhalten. Während sie im Frequenzbereich einen kompakten Träger haben, besitzen sie im Ortsbereich aufgrund ihrer anisotropen Skalierung eine nadelähnliche Form. Für stückweise C^2 -Bilder liefern die Curvelets gemäß [CD04] die Approximationseigenschaft

$$\epsilon_M = \|f - f_M\|_2^2 \leq CM^{-2}(\log_2 M)^3$$

die bis auf den Faktor $(\log_2 M)^3$ optimal ist, wenn man eine zusätzliche Regularitätsbedingung an die Kanten stellt, nämlich, dass diese auch zweimal stetig in Tangentialrichtung differenzierbar sind.

Ein analoges Approximationsresultat wurde in [GL07] für die *Shearlets* bewiesen. Shearlets sind ein Spezialfall der sogenannten *Wavelets mit zusammengesetzter Dilatation* [GLL⁺06, Lim06], die im zweidimensionalen Raum die Form

$$\{\psi_{j,k,i} = |\det A|^{j/2} \psi(B^i A^j \cdot -k) : j, i \in \mathbb{Z}, k \in \mathbb{Z}^2\}$$

haben, wobei $\psi \in L^2(\mathbb{R}^2)$, und A, B invertierbare 2×2 -Matrizen sind mit $|\det B| = 1$. Im Fall der Shearlets wählt man eine anisotrope Skalierungsmatrix A und die Schermatrix $B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$. Das Shearletsystem bildet ebenso wie das Curveletsystem einen Parseval-Frame des $L^2(\mathbb{R}^2)$, im Gegensatz zu den Curvelets liegt diesem Ansatz aber eine Multiresolutionsanalyse zugrunde. Wir betonen schon an dieser Stelle die Wichtigkeit dieser Waveletsysteme für die vorliegende Arbeit, weil unsere konstruierten Wavelets auf Dreiecken auch zu der Klasse der

Wavelets mit zusammengesetzter Dilatation gehören werden. Dabei konstruieren wir erstmals Shearlets mit kompaktem Träger im Ortsbereich.

Die *Contourlets* von Do und Vetterli [DV02, DV05] sind zwar ursprünglich nicht als Diskretisierung der Curvelets eingeführt worden, aber ihre diskrete Formulierung im Ortsbereich führt im Wesentlichen zu derselben Frequenzraum-Überdeckung wie bei den Curvelets [FDF06]. Die Contourlets erfreuen sich wegen ihrer guten Implementierung großer Beliebtheit, die Redundanz des Contourletsystems ist im Vergleich zu den anderen erwähnten Systemen sehr gering. Insbesondere die diskrete Curvelettransformation weist in der Praxis bei der Bildapproximation aufgrund der hohen Redundanz große Schwächen auf.

Die *Directionlets* bilden hingegen ein nicht-redundantes Waveletsystem, das zudem auch noch im Diskreten definiert wird, so dass der häufig sehr schwierige Schritt der Diskretisierung hinfällig ist [VBLVD06].

Aus der großen Menge der „X-lets“ erwähnen wir noch ohne weitere Erläuterung die Ridgelets [CD99], einen Vorläufer der Curvelets, und die Brushlets [MC97] sowie die komplexen Wavelets [Kin01, NS05].

Unter den adaptiven Methoden beschränken wir uns auf diejenigen, die in engem Zusammenhang zu unserer Arbeit stehen.

Während Curvelets oder Shearlets eine fast-optimale Approximation nur für stückweise Hölder-stetige Funktionen zweiter Ordnung mit Unstetigkeiten entlang von C^2 -Kurven liefern, sind adaptive Methoden in der Lage, für allgemeinere Klassen von Bildern eine optimale Approximation bereitzustellen.

Die *ENO-EA-Methoden* [ACD⁺08] erlauben eine nichtlineare kantenadaptierende Multiskalen-Zerlegung eines Bildes und weisen für $\alpha = 2$ ein optimales, asymptotisches Abklingverhalten des Approximationsfehlers ϵ_M auf.

Die *Bandelets* von Le Pennec und Mallat [LPM05a, LPM05b] basieren auf Tensorprodukt-Wavelets, die ihre anisotrope Eigenschaft dadurch erhalten, dass sie mittels eines Krümmungsoperators an die Bildkanten angepasst werden. Die Bandelet-Frames erreichen bei hinreichend glatten Tensorprodukt-Wavelets das optimale Approximationsverhalten für $\alpha \geq 1$.

Eine ganz neue geometrische Methode ist von Plonka vorgeschlagen worden [Plo09]. Die *Easy Path Wavelet Transformation (EPWT)* wandelt ein Bild in einen Vektor um, auf den dann eine eindimensionale Wavelettransformation, wie z. B. D4, angewandt wird. Dabei durchläuft man das Bild auf einem günstigen Pfad entsprechend der Bildgeometrie und schreibt die Bildwerte in den Vektor. In [PTI09] konnte gezeigt werden, dass die EPWT in Verbindung mit Haarwavelets unter gewissen Voraussetzungen optimale Approximationseigenschaften für $\alpha \in (0, 1]$ aufweist.

Die *Wedgelets* [Don99] sind adaptive Haar-Funktionen und erhalten daher in dieser Arbeit besondere Aufmerksamkeit. Sie sind als charakteristische Funktionen auf keilförmigen

Bildgebieten definiert, die man durch Splitten von dyadischen Quadraten erhält. Bemerkenswert ist, dass die Wedgelet-Approximation aus dem Wedgelet-Dictionary nicht mit Hilfe eines Pursuit-Algorithmus gewonnen wird, sondern durch einen Variationsansatz. Die Wedgelets garantieren eine optimale Approximation nur für $1 \leq \alpha \leq 2$, vgl. [FDF06].

In Anbetracht der Tatsache, dass wir uns mit Wavelets auf Dreiecksgebieten beschäftigen wollen, erwähnen wir die Approximationsmethode von Demaret, Dyn und Iske [DDI06], die auf einer adaptiven Triangulierung der Kantengeometrie beruht.

Die von Mallat vorgestellten *Geometrischen Grouplets* [Mal09] sind in der Lage, Bildkanten zu adaptieren, indem sie mittels Assoziationsfelder gewisse Pixelwerte gemäß ihrer lokalen Bildgeometrie gruppieren. Der besondere Reiz an dieser Methode ist die Definition im diskreten Fall, die eine häufig nicht triviale Diskretisierung überflüssig macht. Außerdem kann man über die Definition des Assoziationsfeldes die Redundanz der Grouplet-Systeme beeinflussen, so dass man Basis- oder Framesysteme für den diskreten Bildraum erhält. Theoretische Approximationsresultate sind uns bisher nicht bekannt.

Überblick und Struktur der Arbeit Die vorliegende Arbeit lässt sich grob in zwei Teile gliedern, die in den zentralen Kapiteln 2 und 3 vorgestellt werden. Die in diesen Kapiteln konstruierten Richtungswavelets sind Vertreter der beiden oben beschriebenen Klassen von Waveletsystemen. In Kapitel 2 stellen wir eine nicht-adaptive Wavelettransformation vor, während wir in Kapitel 3 eine adaptive Methode vorschlagen.

Im Einzelnen ist die Arbeit wie folgt strukturiert.

In Kapitel 1 erinnern wir an einige grundlegende Begriffe und Aussagen über Waveletsysteme. Neben der Waveletanalyse wenden wir uns auch der Filterbank-Theorie zu, die für die Implementierung der konstruierten Wavelettransformationen eine wichtige Rolle spielt.

Im umfangreichen Kapitel 2 konstruieren wir gerichtete Haarwavelets auf Dreiecken. Dazu zerlegen wir den Ortsraum in Dreiecke gleichen Flächeninhalts. Zunächst betrachten wir Dreieckszerlegungen mit vier Richtungen und das daraus resultierende Waveletsystem, das eine Orthonormalbasis des $L^2(\mathbb{R}^2)$ generiert. Im nächsten Schritt erweitern wir den Ansatz, indem wir vier weitere Richtungen zulassen. Das nun entstehende Waveletsystem verliert seine Basiseigenschaft, wir erhalten einen Parseval-Frame mit vierfacher Redundanz. Diese Eigenschaft kann im Frequenzbereich mithilfe der Unitarität der Polyphasen-Matrix des Systems gezeigt werden, wie wir im Unterkapitel 2.3 beweisen werden. Schließlich konstruieren wir ein noch allgemeineres Haarwavelet-System: Durch die Verwendung einer anisotropen Dilatationsmatrix können wir beliebig viele Richtungen erzeugen, indem wir entsprechend viele Verfeinerungslevel zulassen. Die Wavelets auf den immer feiner werdenden Dreiecken sind Shearlets mit kompakten Träger im Ortsbereich. Nachdem wir für alle Methoden den Filterbank-Algorithmus ausführlich beschrieben haben, wenden wir ihn zur Entstörung und

zur Approximation von Bildern an. Abschließend zeigen wir, dass es erstaunlich schwierig ist, auf den betrachteten Triangulierungen stückweise lineare Wavelets mit kompakten Trägern zu konstruieren.

In Kapitel 3 führen wir eine adaptive Wavelettransformation ein, die sogenannte Tetrolet-Transformation. Die verwendeten Haarwavelets basieren auf einer adaptiven Tetromino-Zerlegung des diskreten Bildraumes und bilden eine Orthonormalbasis. Auch hier beschreiben wir den Filterbank-Algorithmus und illustrieren an vielen Beispielen die enorme Effizienz der Methode bei der dünnen Darstellung (engl. *sparse representation*) von Bildern. Wir untersuchen die numerische Komplexität und die zusätzlichen Adaptivitätskosten, was zu kostenreduzierenden Modifikationen der Transformation führt. Im abschließenden Unterkapitel 3.9 zeigen wir die adaptive Multiskalen-Struktur sowie die wichtige Eigenschaft, dass die Tetrolets ein optimales Approximationsverhalten für $\alpha \in (0, 1]$ liefern.

Das letzte Kapitel 4 ist einer Postprocessing-Methode gewidmet, die die Schwächen der beiden konstruierten Wavelettransformationen beheben soll. Die in Kapitel 2 und 3 vorgestellten Techniken beruhen auf Haarwavelets, was zu einer stückweise konstanten Approximation führt. Durch Anwenden eines angepassten Postprocessing-Schemas können wir die Regularität des approximierten Bildes nachträglich erhöhen, ohne dabei die Bildkanten zu verwischen. Unsere Methode verwendet dazu eine anisotrope TV-Minimierung. Die guten Ergebnisse in der Anwendung bestätigen die theoretischen Resultate, dass die Approximationsqualität bei geeigneter Wahl der Parameter des Postprocessing-Verfahrens verbessert werden kann.

Danksagung Diese Dissertation ist das Ergebnis einer mehr als dreijährigen Forschungszeit an der Fakultät für Mathematik an der Universität Duisburg-Essen, Campus Duisburg. Ich bedanke mich in erster Linie bei meiner Betreuerin Frau Prof. Dr. Plonka-Hoch für ihre intensive Unterstützung in dieser Zeit. Ihre anregenden Vorschläge und Ideen haben mir den Weg gewiesen. Außerdem möchte ich Frau Prof. Dr. Steidl für die Erstellung des Zweitgutachtens und die damit verbundenen hilfreichen Anmerkungen danken. Meinen Kollegen bin ich sehr dankbar für die angenehme Arbeitsatmosphäre, die ich in den letzten Jahren genossen habe. Insbesondere danke ich Michael Wozniczka und Stefanie Tenorth für viele fruchtbare Diskussionen. Weiterer Dank gebührt der Deutschen Forschungsgemeinschaft, die durch Gewährung einer Sachbeihilfe für das dreijährige Forschungsprojekt die finanzielle Grundlage für die Entstehung dieser Arbeit geschaffen hat.

Der außeruniversitäre Dank geht an meine Eltern, die mir das Studium ermöglichten, und an meine Freunde, die mich zu jeder Zeit unterstützt haben. Besonders bedanke ich mich bei meiner Verlobten Eva, der ich diese Arbeit widme.

Jens Krommweh, April 2010

Kapitel 1

Grundlagen der Waveletsysteme und ihrer Anwendungen

In diesem ersten Kapitel stellen wir einige grundlegende Werkzeuge der Waveletanalyse und der Signalverarbeitung bereit, auf die wir im Verlauf der Arbeit immer wieder zurückgreifen werden.

Das erste Unterkapitel stellt die verschiedenen Basiskonzepte in einem Hilbertraum vor. Danach erläutern wir Begriffe aus der Waveletanalyse in einer einheitlichen Notation. Der dritte Abschnitt beleuchtet die Wavelettheorie aus der Perspektive der Signalverarbeitung, indem verschiedene Filterbank-Algorithmen vorgestellt werden. Schließlich beschreiben wir noch mit ein paar allgemeinen Bemerkungen die Grundideen der Multiskalenmethoden in der digitalen Bildverarbeitung.

1.1 Funktionensysteme

In einem separablen Hilbertraum H und für eine abzählbare Indexmenge K gelten folgende Begrifflichkeiten:

Definition 1.1. Das System $\{f_k\}_{k \in K}$ heißt ...

1. ... vollständig in H , wenn sein linearer Aufspann dicht in H ist, d. h. $\overline{\text{span}\{f_k\}_{k \in K}} = H$.
Ein vollständiges System nennen wir Dictionary.

2. ... orthogonal, wenn

$$\langle f_k, f_{k'} \rangle = \delta_{k,k'} \quad \text{für alle } k, k' \in K. \quad (1.1)$$

3. ... orthonormal, wenn es orthogonal ist und $\|f_k\|_H = 1$ für alle $k \in K$ gilt.

4. ... Basis von H , wenn für alle $f \in H$ eindeutige Skalare c_k existieren, so dass gilt:

$$f = \sum_{k \in K} c_k f_k. \quad (1.2)$$

5. ... Orthonormalbasis (ONB) von H , wenn das System eine Basis und zusätzlich orthonormal ist.

Die Orthogonalitätsbedingung in (1.1) ist eine starke Forderung an das Funktionensystem, die wir etwas lockern können, indem wir von einer ONB zu einer Rieszbasis übergehen.

Definition 1.2. Eine Basis $\{f_k\}_{k \in K}$ heißt ...

1. ... beschränkt, wenn gilt: $0 < \inf_{k \in K} \|f_k\|_H \leq \sup_{k \in K} \|f_k\|_H < \infty$.
2. ... unbedingt (engl. unconditional), wenn jede Reihendarstellung $\sum_{k \in K} c_k f_k$ in H unbedingt konvergiert, d. h. wenn auch jede Umordnung der Reihe konvergiert.
3. ... Rieszbasis von H , wenn sie beschränkt und unbedingt ist.

Für Orthonormalbasen impliziert die Orthogonalitätsbedingung (1.1) die Parseval'sche Gleichung

$$\|f\|_H^2 = \sum_{k \in K} |c_k|^2 = \|\{c_k\}_{k \in K}\|_{l^2}^2 \quad \text{für alle } f \in H,$$

dabei haben die eindeutigen Koeffizienten die Gestalt $c_k = \langle f, f_k \rangle$ für alle $k \in K$. Für Rieszbasen kann eine ähnliche Relation zwischen der darzustellenden Funktion f aus H und den Koeffizienten c_k aus l^2 formuliert werden. Diese Ungleichung charakterisiert die Rieszbasis eindeutig:

Satz 1.3. Das System $\{f_k\}_{k \in K}$ ist eine Rieszbasis von H genau dann, wenn es vollständig in H ist, und wenn positive, endliche Konstanten A und B existieren, so dass für alle $\{c_k\}_{k \in K} \in l^2(K)$ gilt:

$$A \|\{c_k\}_{k \in K}\|_{l^2}^2 \leq \left\| \sum_{k \in K} c_k f_k \right\|_H^2 \leq B \|\{c_k\}_{k \in K}\|_{l^2}^2. \quad (1.3)$$

Die optimalen Konstanten A und B heißen untere und obere Riesz-Schranken.

Im Fall von Rieszbasen lassen sich die eindeutigen Koeffizienten durch $c_k = \langle f, \tilde{f}_k \rangle$, $k \in K$, berechnen. Hierbei ist $\{\tilde{f}_k\}_{k \in K}$ die eindeutige, zu $\{f_k\}_{k \in K}$ duale Rieszbasis von H , d. h. es gilt $\langle f_k, \tilde{f}_{k'} \rangle = \delta_{k,k'}$ für alle $k, k' \in K$.

Jetzt gehen wir noch einen Schritt weiter, indem wir das Basiskonzept aufweichen. Wir suchen ein reproduzierendes Funktionensystem, das nicht mehr die starken Forderungen einer Basis erfüllen soll, aber trotzdem eine Darstellung aller Funktionen f aus H in der Form von (1.2) bietet.

Definition 1.4. Das System $\{f_k\}_{k \in K}$ heißt Frame von H , wenn positive, endliche Konstanten A und B existieren, so dass für alle $f \in H$ gilt:

$$A \|f\|_H^2 \leq \|\{\langle f, f_k \rangle\}_{k \in K}\|_{l^2}^2 \leq B \|f\|_H^2.$$

Die optimalen Konstanten A und B heißen untere und obere Frame-Schranken. Wenn $A = B$ gewählt werden kann, spricht man von einem Tight-Frame, im Falle von $A = B = 1$ von einem Parseval-Frame.

Wenn eine positive, endliche Konstante B existiert, so dass die rechte Ungleichung gilt, ist $\{f_k\}_{k \in K}$ ein Besselsystem von H .

Jeder Frame ist also ein Besselsystem. Ein Frame unterscheidet sich von einer Basis durch die fehlende lineare Unabhängigkeit der Frame-Elemente f_k . Das führt dazu, dass jedes f aus H sich zwar in der Tat als Linearkombination der Framefunktionen $\{f_k\}_{k \in K}$ darstellen lässt, aber die Koeffizienten $c_k = \langle f, \tilde{f}_k \rangle$ nicht mehr eindeutig sind, weil es verschiedene duale Frames $\{\tilde{f}_k\}_{k \in K}$ gibt, die eine Darstellung in der Form (1.2) gewährleisten. Die lineare Abhängigkeit der Frame-Elemente f_k führt zu dem wichtigen Begriff der Redundanz:

Definition 1.5. 1. Ein Frame $\{f_k\}_{k \in K}$ von H heißt *exakt* (oder *minimal*), wenn er eine *Rieszbasis* von H ist.

2. Ein Frame $\{f_k\}_{k \in K}$ von H heißt *redundant* (oder *nicht minimal* oder *overcomplete*), wenn er *nicht exakt* ist.

Für eine vertiefende Frame-Analyse verweisen wir auf das ausgezeichnete Buch von Christensen [Chr03].

1.2 Multiresolutionsanalyse und Wavelets

Die Reihenfolge der eingeführten Begrifflichkeiten in diesem Abschnitt entspricht dem üblichen Vorgehen, Wavelets über eine Multiresolutionsanalyse und eine Skalierungsfunktion zu konstruieren. Das Ziel der Multiresolutionsanalyse ist die Zerlegung des Raumes $L^2(\mathbb{R}^d)$ in geeignete Unterräume $V_j, j \in \mathbb{Z}$, die wir *Approximationsräume* nennen. Die Grundidee ist dabei, die Approximationsräume ineinandergeschachtelt zu wählen, so dass $V_j \subset V_{j+1}$ gilt. Wenn nun P_{V_j} der L^2 -Orthogonalprojektor in den Raum V_j ist, so stellt $P_{V_{j+1}}f$ eine bessere Approximation von $f \in L^2(\mathbb{R}^d)$ dar als $P_{V_j}f$, weil V_{j+1} „feinere“ Funktionen enthalten soll als V_j und deshalb eine „feinere“ Auflösung (engl. resolution) der Funktion f gestattet. Diese Verfeinerung von einem Auflösungslevel zum nächsten geschieht mittels einer *Dilatationsmatrix* A der Größe $d \times d$, deren Einträge ganzzahlig und deren Eigenwerte betragsmäßig größer als Eins sein sollen.

Die klassische Multiresolutionsanalyse Das Konzept der Multiresolutionsanalyse wurde von Mallat und Meyer [Mal89, Mey90] für $d = 1$ mit der dyadischen Skalierung $A = 2$ eingeführt. In diesem Unterkapitel werden wir die Theorie für die allgemeine Dimension $d \in \mathbb{N}$ vorstellen, später werden wir im Fall von Bildern $d = 2$ wählen.

Definition 1.6. Eine Folge $\{V_j\}_{j \in \mathbb{Z}}$ von abgeschlossenen Unterräumen des $L^2(\mathbb{R}^d)$ heißt Multiresolutionsanalyse (MRA) (oder Multiskalenanalyse) von $L^2(\mathbb{R}^d)$, wenn die folgenden Eigenschaften erfüllt sind

(M1) $V_j \subset V_{j+1}$ für alle $j \in \mathbb{Z}$.

(M2) $f \in V_j \Leftrightarrow f(A \cdot) \in V_{j+1}$ für alle $j \in \mathbb{Z}$.

(M3) $\text{clos}_{L^2(\mathbb{R}^d)} \bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}^d)$.

(M4) $\bigcap_{j \in \mathbb{Z}} V_j = \{0\}$.

(M5) Es gibt eine Funktion ϕ , so dass $\{\phi(\cdot - k) : k \in \mathbb{Z}^d\}$ eine ONB von V_0 ist.

Die Funktion ϕ in (M5) heißt Generator der MRA.

Aus (M2) und (M5) folgt sofort, dass die Familie $\{\phi_{j,k} : k \in \mathbb{Z}^d\}$ mit

$$\phi_{j,k} := |\det A|^{j/2} \phi(A^j \cdot - k)$$

eine ONB für jeden Approximationsraum V_j ist. Die generierende Funktion ϕ der Approximationsräume heißt *Skalierungsfunktion*. Wegen (M1) existiert dann eine Koeffizienten-Folge $\{h_k^0\}_{k \in \mathbb{Z}^d} \in l^2(\mathbb{Z}^d)$, so dass die *Verfeinerungsgleichung* (oder *Skalierungsgleichung*)

$$\phi(x) = |\det A|^{1/2} \sum_{k \in \mathbb{Z}^d} h_k^0 \phi(Ax - k) \quad (1.4)$$

für die Skalierungsfunktion ϕ gilt. Man nennt ϕ *verfeinerbar*, die Folge $\{h_k^0\}_{k \in \mathbb{Z}^d}$ heißt *Maske* von ϕ . In der Anwendung sind Masken vorteilhaft, bei denen nur endlich viele Koeffizienten von Null verschieden sind. Skalierungsfunktionen mit kompaktem Träger besitzen stets endliche Masken (siehe z. B. [Mal99, Proposition 7.2]), und es lassen sich dazugehörige Wavelets konstruieren, die ebenfalls kompakten Träger besitzen. Solchen Skalierungs- und Waveletfunktionen gilt unser besonderes Interesse.

Wir werden sehen, dass die Verfeinerungsgleichung (1.4) das Herzstück der Wavelettheorie ist, weil sie eine schnelle Wavelettransformation mittels eines Filterbank-Algorithmus erlaubt. Die Gleichung (1.4) lässt sich auch im Frequenzbereich $\hat{\mathbb{R}}^d$ formulieren. Die *Fouriertransformierte* \hat{f} von $f \in L^2(\mathbb{R}^d) \cap L^1(\mathbb{R}^d)$ ist definiert durch

$$\hat{f}(\omega) = \int_{\mathbb{R}^d} f(x) e^{-i\omega x} dx,$$

wobei die Punkte $\omega \in \hat{\mathbb{R}}^d$ im Frequenzbereich als Zeilenvektoren gegeben sind (im Gegensatz zu den Spaltenvektoren $x \in \mathbb{R}^d$ im Zeitbereich). Die Anwendung der Fouriertransformation auf beide Seiten von (1.4) liefert die Verfeinerungsgleichung im Frequenzbereich

$$\hat{\phi}(\omega) = H^0(\omega A^{-1}) \hat{\phi}(\omega A^{-1}). \quad (1.5)$$

Dabei ist die Fourierreihe H^0 das sogenannte *Symbol* der Maske $\{h_k^0\}_{k \in \mathbb{Z}^d}$,

$$H^0(\omega) = \frac{1}{|\det A|^{1/2}} \sum_{k \in \mathbb{Z}^d} h_k^0 e^{-i\omega k}. \quad (1.6)$$

Nun sind wir in der Lage, die gewünschten Wavelets zu konstruieren. Um eine anschauliche Vorstellung von Wavelets zu haben, kann man Wavelets als Differenzen interpretieren.

Wir wissen, dass $V_j \subset V_{j+1}$ für $j \in \mathbb{Z}$ gilt. Sei nun W_j das orthogonale Komplement von V_j in V_{j+1} :

$$V_{j+1} = V_j \oplus W_j. \quad (1.7)$$

Die orthogonale Projektion von $f \in L^2(\mathbb{R}^d)$ in V_{j+1} kann dann zerlegt werden in die direkte Summe der beiden Orthogonalprojektionen in V_j und W_j :

$$P_{V_{j+1}} f = P_{V_j} f + P_{W_j} f.$$

Dabei enthält $P_{W_j} f$ die Detailinformationen von f , die in der feinen Auflösung V_{j+1} auftreten, aber in der nächst gröberen Auflösung V_j schon nicht mehr dargestellt werden können. Wegen (M4) folgt aus der Beziehung (1.7) die Zerlegung bis zum Level j

$$V_j = \bigoplus_{j'=-\infty}^{j-1} W_{j'}.$$

Benutzen wir jetzt (M3), so erhalten wir die vollständige Zerlegung des $L^2(\mathbb{R}^d)$ in die Differenzräume:

$$L^2(\mathbb{R}^d) = \bigoplus_{j \in \mathbb{Z}} W_j.$$

Analog wie bei den Approximationsräumen ist es wünschenswert, dass die Differenzräume W_j durch Basen aufgespannt werden, die durch Skalierung und Translation einer Generatorfunktion erzeugt werden.

Satz 1.7. Sei $\{V_j\}_{j \in \mathbb{Z}}$ eine MRA des $L^2(\mathbb{R}^d)$ mit der Dilatationsmatrix A . Sei $m := |\det A|$. Dann existieren $m - 1$ Wavelets $\psi^1, \dots, \psi^{m-1} \in V_1$, die eine ONB des orthogonalen Komplements von V_0 in V_1 erzeugen, d. h. die Menge

$$\{\psi_{j,k}^l : l = 1, \dots, m - 1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$$

mit

$$\psi_{j,k}^l := |\det A|^{j/2} \psi^l(A^j \cdot -k)$$

ist eine ONB von $L^2(\mathbb{R}^d)$.

Für festes $j \in \mathbb{Z}$ sind die Funktionensysteme $\{\psi_{j,k}^l : l = 1, \dots, m - 1, k \in \mathbb{Z}^d\}$ Orthonormalbasen von W_j , die Basisfunktionen heißen *Wavelets*, die $m - 1$ Generatorfunktionen

ψ^l heißen *Mutterwavelets*, und die Differenzräume W_j werden dementsprechend *Wavelet-räume* genannt.

Wegen (1.7) gilt $W_0 \subset V_1$, und wir können die Mutterwavelets $\psi^l \in W_0 \subset V_1$ bezüglich der Basis $\{\phi(A \cdot -k) : k \in \mathbb{Z}^d\}$ darstellen:

$$\psi^l(x) = |\det A|^{1/2} \sum_{k \in \mathbb{Z}^d} h_k^l \phi(Ax - k), \quad l = 1, \dots, m-1 \quad (1.8)$$

mit einer Maske $\{h_k^l\}_{k \in \mathbb{Z}^d}$. Die Konstruktion von orthonormalen Wavelets reduziert sich also auf die Aufgabe, geeignete Koeffizienten h_k^l zu bestimmen. Wenn ϕ kompakten Träger hat, lassen sich Wavelets mit der gleichen Eigenschaft konstruieren. Wir suchen also Wavelets mit nur endlich vielen, von Null verschiedenen Skalierungskoeffizienten h_k^l in (1.8). Die Bestimmung dieser Koeffizienten kann auf zwei verschiedene Weisen erfolgen: im Orts- oder im Frequenzbereich.

Ausgehend von der Skalierungsfunktion ϕ , die durch ihre Verfeinerungskoeffizienten h_k^0 bestimmt ist, können mit Hilfe des folgenden Satzes die Koeffizienten h_k^l im Ortsbereich so gewählt werden, dass die Wavelets eine Orthogonalbasis erzeugen.

Satz 1.8. *Die Skalierungsfunktion $\phi \in L^2(\mathbb{R}^d)$ generiere eine MRA für $L^2(\mathbb{R}^d)$, und ϕ erfülle die Verfeinerungsgleichung (1.4) mit einer endlichen Maske $\{h_k^0\}_{k \in \mathbb{Z}^d}$. Dann gilt:*

Das durch die Verfeinerungsgleichung (1.8) definierte Waveletssystem

$$\{\psi_{j,k}^l : l = 1, \dots, m-1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$$

ist eine ONB des $L^2(\mathbb{R}^d)$ genau dann, wenn gilt:

$$\sum_{k \in \mathbb{Z}^d} h_k^l \overline{h_{k-Ak'}^{l'}} = \delta_{0,k'} \delta_{l,l'}, \quad l, l' = 0, \dots, m-1.$$

Beweis. Siehe [Bow97]. □

Damit wir alternativ die Wavelets im Frequenzbereich konstruieren können, formulieren wir die Verfeinerungsgleichung (1.8) analog zu (1.5) im Frequenzbereich:

$$\hat{\psi}^l(\omega) = H^l(\omega A^{-1}) \hat{\phi}(\omega A^{-1}), \quad l = 1, \dots, m-1. \quad (1.9)$$

Hier bezeichnet die Fourierreihe H^l das *Symbol* der Wavelet-Maske $\{h_k^l\}_{k \in \mathbb{Z}^d}$,

$$H^l(\omega) = \frac{1}{|\det A|^{1/2}} \sum_{k \in \mathbb{Z}^d} h_k^l e^{-i\omega k}, \quad l = 1, \dots, m-1. \quad (1.10)$$

Ausgehend von dem Symbol H^0 der gegebenen Skalierungsfunktion ϕ , können wir mit Hilfe des folgenden Satzes geeignete Waveletsymbole $H^l, l = 1, \dots, m-1$, finden, so dass die dazugehörigen Wavelets eine ONB bilden.

Satz 1.9. Die Skalierungsfunktion $\phi \in L^2(\mathbb{R}^d)$ generiere eine MRA für $L^2(\mathbb{R}^d)$, und ϕ erfülle die Verfeinerungsgleichung (1.5) mit einem 2π -periodischen Symbol H^0 . Dann gilt für 2π -periodische $H^l, l = 1, \dots, m-1$:

Das durch die Verfeinerungsgleichung (1.9) definierte Waveletssystem

$$\{\psi_{j,k}^l : l = 1, \dots, m-1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$$

ist eine ONB des $L^2(\mathbb{R}^d)$ genau dann, wenn die Matrix $\mathcal{M}(\omega) = [H^l(\omega + 2\pi\gamma_j^T A^{-1})]_{l,j=0}^{m-1}$ unitär ist.

Beweis. Siehe [Woj97, Proposition 5.9]. □

Die $m \times m$ -Matrix $\mathcal{M}(\omega)$ heißt *Modulationsmatrix*, ihre Einträge sind 2π -periodische Funktionen. Die m Spaltenvektoren $\{\gamma_0, \gamma_1, \dots, \gamma_{m-1}\}$ sind Repräsentanten des Gitters \mathbb{Z}^2 . Beide Begrifflichkeiten werden wir im Verlauf des Kapitels noch näher erläutern.

Es ist aus der Signalverarbeitung bekannt, dass z. B. die Wahl von *Conjugate Quadrature Filtern* (CQF) die Unitaritätsbedingung des Satzes erfüllt, im Fall $d = 1$ wählt man $H^1(\omega) = e^{-i\omega} \overline{H^0(\omega + \pi)}$. Wenn man die Waveletsymbole H^l für $l = 1, \dots, m-1$, gefunden hat, kennt man auch die Masken $\{h_k^l\}_{k \in \mathbb{Z}^d}$ und kann damit die gesuchten Wavelets ψ^l konstruieren.

Im folgenden Beispiel führen wir die Haarwavelets für $d = 1$ ein, deren bivariate Versionen in unserer Arbeit eine zentrale Rolle spielen werden.

Beispiel 1.10. Das erste in der Literatur bekannt gewordene Wavelet tauchte schon 80 Jahre vor der Entwicklung der Wavelettheorie auf. Im Jahr 1909 bewies der ungarische Mathematiker Alfred Haar (* am 11. Oktober 1885 in Budapest, † am 16. März 1933 in Szeged) in seiner Dissertation, dass Differenzen von charakteristischen Funktionen auf Intervallen eine ONB des $L^2(\mathbb{R})$ bilden. Diese Ergebnisse wurden 1910 in seiner berühmten Arbeit [Haa10] veröffentlicht.

Sei die Skalierungsfunktion durch die charakteristische Funktion $\phi = \chi_{[0,1)}$ auf dem Intervall $[0, 1)$ gegeben. Die Räume $V_j := \text{clos}_{L^2(\mathbb{R})} \text{span}\{2^{j/2} \phi(2^j \cdot -k) : k \in \mathbb{Z}\}$ bilden für $j \in \mathbb{Z}$ eine MRA des $L^2(\mathbb{R})$, insbesondere gilt die Verfeinerungsgleichung

$$\phi(x) = \phi(2x) + \phi(2x - 1) = \chi_{[0, \frac{1}{2})}(x) + \chi_{[\frac{1}{2}, 1)}(x).$$

Das bedeutet, dass in der Maske $\{h_k^0\}_{k \in \mathbb{Z}}$ alle Koeffizienten bis auf $h_0^0 = h_1^0 = \frac{1}{\sqrt{2}}$ verschwinden und das Symbol die Gestalt $H^0(\omega) = \frac{1}{2}(1 + e^{-i\omega})$ annimmt.

Gemäß Satz 1.9 finden wir eine ONB von Wavelets für den $L^2(\mathbb{R})$ mittels der Waveletsymbole $H^1(\omega)$, wenn die Modulationsmatrix

$$\mathcal{M}(\omega) = \begin{pmatrix} H^0(\omega) & H^0(\omega + \pi) \\ H^1(\omega) & H^1(\omega + \pi) \end{pmatrix}$$

unitär ist. Man rechnet leicht nach, dass dies der Fall ist, wenn z. B. $H^1(\omega) = \frac{1}{2}(1 - e^{-i\omega})$ gilt. Also verschwinden alle Koeffizienten des Waveletsymbols H^1 bis auf $h_0^1 = \frac{1}{\sqrt{2}}$ und $h_1^1 = -\frac{1}{\sqrt{2}}$, d. h. das Mutterwavelet ist

$$\psi(x) = \phi(2x) - \phi(2x - 1) = \chi_{[0, \frac{1}{2})}(x) - \chi_{[\frac{1}{2}, 1)}(x),$$

und die Familie $\{\psi_{j,k} : j, k \in \mathbb{Z}\}$ mit $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$ bildet eine ONB für den Raum $L^2(\mathbb{R})$.

Verallgemeinerungen des MRA-Konzeptes: Riesz-MRA und Frame-MRA Die in Definition 1.6 formulierte MRA beschreibt das ursprüngliche MRA-Konzept von Mallat und Meyer [Mal89, Mey90]. Dabei wird in (M5) insbesondere gefordert, dass die ganzzahligen Translationen der Skalierungsfunktion eine ONB von V_0 bilden. Wenn man bestimmte Eigenschaften (wie z. B. Symmetrie) für die konstruierten Wavelets erzielen möchte, erweist sich das ONB-Korsett als zu eng. Deshalb ist es sinnvoll, eine Verallgemeinerung auf die schwächeren Funktionensysteme vorzunehmen, die im Unterkapitel 1.1 vorgestellt wurden. Wenn dabei aufgrund des Zusammenhangs eine deutliche Abgrenzung zwischen den verschiedenen Varianten der MRAs nötig ist, werden wir die klassische MRA aus Definition 1.6 auch manchmal als *ONB-MRA* bezeichnen.

Zunächst ersetzen wir (M5) durch

(M5') Es gibt eine Funktion ϕ , so dass $\{\phi(\cdot - k) : k \in \mathbb{Z}^d\}$ eine Rieszbasis von V_0 ist.

Eine solche MRA nennen wir *Riesz-MRA* [BNB00]. Statt einer ONB bilden die Familien $\{\phi_{j,k} : k \in \mathbb{Z}^d, j \in \mathbb{Z}\}$, nun jeweils eine Rieszbasis mit den gleichen Riesz-Konstanten für jeden Approximationsraum V_j . Zu $\{V_j\}_{j \in \mathbb{Z}}$ existiert eine *biorthogonale MRA* $\{\tilde{V}_j\}_{j \in \mathbb{Z}}$, d. h. eine zweite Riesz-MRA des $L^2(\mathbb{R}^d)$, die von einer Skalierungsfunktion $\tilde{\phi}$ generiert wird, die die Relation

$$\langle \phi(\cdot - k), \tilde{\phi}(\cdot - k') \rangle = \delta_{k,k'}$$

für alle $k, k' \in \mathbb{Z}^d$ erfüllt. Dann lassen sich Waveleträume W_j und \tilde{W}_j finden, die komplementär zu V_j in V_{j+1} bzw. zu \tilde{V}_j in \tilde{V}_{j+1} sind, und die miteinander verbunden sind durch

$$W_j \perp \tilde{V}_j, \quad \tilde{W}_j \perp V_j.$$

Für festes $j \in \mathbb{Z}$ existieren nun Waveletsysteme

$$\{\psi_{j,k}^l : l = 1, \dots, m-1, k \in \mathbb{Z}^d\} \quad \text{und} \quad \{\tilde{\psi}_{j,k}^l : l = 1, \dots, m-1, k \in \mathbb{Z}^d\},$$

die Rieszbasen von W_j bzw. \tilde{W}_j sind und biorthogonal zueinander sind:

$$\langle \psi_{j,k}^l, \tilde{\psi}_{j',k'}^{l'} \rangle = \delta_{j,j'} \delta_{k,k'} \delta_{l,l'}.$$

Für alle $j \in \mathbb{Z}$ bilden diese beiden Waveletsysteme jeweils eine Rieszbasis für den Raum $L^2(\mathbb{R}^d)$.

Schließlich erhalten wir eine *Frame-MRA* durch eine weitere Verallgemeinerung

(M5'') Es gibt eine Funktion ϕ , so dass $\{\phi(\cdot - k) : k \in \mathbb{Z}^d\}$ ein Frame von V_0 ist.

Die Familien $\{\phi_{j,k} : k \in \mathbb{Z}^d\}, j \in \mathbb{Z}$, bilden jetzt jeweils einen Frame mit den gleichen Frame-Schranken für die Approximationsräume V_j . Der Übergang von einer Basis zu einem Frame bringt viele technische Schwierigkeiten mit sich, die in [Chr03, Kapitel 13] für den Fall $d = 1$ untersucht worden sind. Unter gewissen weiteren Bedingungen kann die Existenz von Waveletframes für die Waveleträume W_j und damit auch für den $L^2(\mathbb{R}^d)$ garantiert werden, vgl. auch [BT01]. Solche Wavelets werden in der Literatur manchmal *Framelets* genannt.

Einen leicht modifizierten Ansatz präsentierten Ron und Shen in [RS97], indem sie statt der ONB-Bedingung nur die Verfeinerbarkeit der Generatorfunktion $\phi \in L^2(\mathbb{R}^d)$ forderten. Ausgehend von den beiden Voraussetzungen

- (i) Es gibt ein 2π -periodisches Polynom H^0 , so dass $\hat{\phi}(\omega A) = H^0(\omega) \hat{\phi}(\omega)$ für $\omega \in \hat{\mathbb{R}}^d$ gilt;
- (ii) Es gilt $\lim_{\omega \rightarrow 0} \hat{\phi}(\omega) = 1$;

sucht man Funktionen ψ^l in $V_1 = \text{clos}_{L^2(\mathbb{R}^d)} \text{span}\{\phi(A \cdot -k) : k \in \mathbb{Z}^d\}$, so dass das Waveletsystem $\{\psi^l(A^j \cdot -k) : l = 1, \dots, L-1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$ mit $L \geq m$ ein Tight-Frame für den $L^2(\mathbb{R}^d)$ ist. Dies geschieht im Frequenzbereich mittels der so genannten *Extension Principles*, die wir unten anführen und im weiteren Verlauf dieser Arbeit häufiger anwenden werden.

Der große Vorteil bei diesem Ansatz ist, dass nicht alle MRA-Bedingungen für die Funktion ϕ überprüft werden müssen, sondern dass lediglich die Verfeinerbarkeit sichergestellt werden muss. Man kann aber zeigen [Chr03, Satz 13.3.3], dass die beiden Bedingungen (i) und (ii) implizieren, dass ϕ eine MRA $\{V_j\}_{j \in \mathbb{Z}}$ mit $V_j := \text{clos}_{L^2(\mathbb{R}^d)} \text{span}\{\phi(A^j \cdot -k) : k \in \mathbb{Z}^d\}$ generiert, wobei die klassische Stabilitätsbedingung ersetzt wird durch

(M5''') Es gibt eine verfeinerbare Funktion ϕ , so dass $\text{clos}_{L^2(\mathbb{R}^d)} \text{span}\{\phi(\cdot - k) : k \in \mathbb{Z}^d\} = V_0$ gilt.

Die Translationen $\{\phi(\cdot - k) : k \in \mathbb{Z}^d\}$ einer verfeinerbaren Funktion bilden nach [Chr03, Kapitel 14] ein Besselsystem und nicht notwendigerweise einen Frame von V_0 .

Die Extension Principles sind Erweiterungen von Satz 1.9, der uns erlaubte, Waveletbasen anhand der Symbole zu charakterisieren bzw. zu konstruieren. Diese Principles beantworten die Frage, ob eine ähnliche Charakterisierung des Waveletsystems auch möglich ist, wenn die generierende Skalierungsfunktion nicht die ONB-Eigenschaft erfüllt, sondern nur ein Erzeugendensystem bildet, wenn also statt (M5) nur (M5''') gilt. In ihrer fundamentalen Arbeit über Waveletframes haben Daubechies et al. [DHRS03] vier verschiedene Extension

Principles vorgestellt, von denen das *Unitary Extension Principle (UEP)* die einfachste Version ist. Es ist ebenso wie das *Oblique Extension Principle (OEP)* bereits in [RS97] zu finden – wengleich auch ohne die Bezeichnungen UEP und OEP. Es ist das Analogon zu Satz 1.9.

Satz 1.11 (UEP). *Die Funktion $\phi \in L^2(\mathbb{R}^d)$ erfülle die Verfeinerungsgleichung $\hat{\phi}(\omega A) = H^0(\omega) \hat{\phi}(\omega)$, $\omega \in \hat{\mathbb{R}}^d$, mit 2π -periodischem H^0 . Außerdem gelte $\lim_{\omega \rightarrow 0} \hat{\phi}(\omega) = 1$. Dann gilt:*

Wenn $\mathcal{M}(\omega) = [H^l(\omega + 2\pi\gamma_j^T A^{-1})]_{l,j=0}^{L-1,m-1}$ unitär ist, wobei H^l , $l = 1, \dots, L-1$, mit $L \geq m$, 2π -periodische Polynome sind, so ist das durch die Verfeinerungsgleichungen $\hat{\psi}^l(\omega A) = H^l(\omega) \hat{\phi}(\omega)$ definierte Waveletsystem

$$\{\psi_{j,k}^l : l = 1, \dots, L-1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$$

ein Parseval-Frame des $L^2(\mathbb{R}^d)$.

Beweis. Das UEP ist ein Spezialfall des folgenden OEPs. Dessen Beweis ist in [RS97] zu finden. \square

Eine erste Verallgemeinerung dieser Aussage ist das OEP, das mehr Freiheiten bei der Konstruktion der Waveletsymbole zulässt.

Satz 1.12 (OEP). *Die Funktion $\phi \in L^2(\mathbb{R}^d)$ erfülle die Verfeinerungsgleichung $\hat{\phi}(\omega A) = H^0(\omega) \hat{\phi}(\omega)$, $\omega \in \hat{\mathbb{R}}^d$, mit 2π -periodischem H^0 . Außerdem sei $S(\omega)$ ein 2π -periodisches trigonometrisches Polynom, so dass gilt: $\lim_{\omega \rightarrow 0} S(\omega) \hat{\phi}(\omega) = 1$ und $\lim_{\omega \rightarrow \infty} S(\omega) \hat{\phi}(\omega) = 0$. Wenn nun für $j = 0, \dots, m-1$*

$$S(\omega A) \overline{H^0(\omega + 2\pi\gamma_j^T A^{-1})} H^0(\omega) + \sum_{l=1}^{L-1} \overline{H^l(\omega + 2\pi\gamma_j^T A^{-1})} H^l(\omega) = \delta_{j,0} S(\omega) \quad (1.11)$$

gilt, wobei H^l , $l = 1, \dots, L-1$, 2π -periodisch sind, so ist das durch die Verfeinerungsgleichungen $\hat{\psi}^l(\omega A) = H^l(\omega) \hat{\phi}(\omega)$ definierte Waveletsystem

$$\{\psi_{j,k}^l : l = 1, \dots, L-1, j \in \mathbb{Z}, k \in \mathbb{Z}^d\}$$

ein Parseval-Frame des $L^2(\mathbb{R}^d)$.

Beweis. Siehe [RS97, Theorem 6.5]. \square

Das zusätzliche trigonometrische Polynom $S(\omega)$ kann bei geeigneter Wahl dazu benutzt werden, um Eigenschaften der Wavelets wie verschwindende Momente oder hohe Regularität zu erzwingen. Anwendungsbeispiele finden sich in [CHS02, DHRS03].

Bemerkung 1.13. 1. *Unter Verwendung der Modulationsmatrix $\mathcal{M}(\omega)$ können wir (1.11) in Matrix-Vektor-Schreibweise formulieren:*

$$\left(S(\omega A) H^0(\omega), H^1(\omega), \dots, H^{L-1}(\omega) \right) \overline{\mathcal{M}(\omega)} = (S(\omega), 0, \dots, 0).$$

2. Wählt man $S(\omega) = 1$, $\omega \in \hat{\mathbb{R}}^d$, so reduziert sich das OEP zum UEP.

Der Vollständigkeit halber erwähnen wir noch die beiden übrigen Extension Principles ohne sie explizit zu formulieren. Das *Mixed Unitary Extension Principle (MUEP)* bzw. das *Mixed Oblique Extension Principle (MOEP)* erweitern das UEP bzw. OEP, indem sie aus zwei verfeinerbaren Funktionen bei geeignetem Zusammenspiel der Symbole zwei duale Waveletframes konstruieren.

Unabhängig von Daubechies et al. [DHRS03] haben Chui, He und Stöckler [CHS02] ähnliche Resultate erhalten. Dabei beschränkten sie sich auf den Fall von skalarwertigen Funktionen im univariaten Fall $d = 1$. Im Kapitel 2.3 werden wir die Varianten der Extension Principles für vektorwertige Skalierungsfunktionen und Wavelets untersuchen.

Zum Schluss stellen wir noch eine spezielle Frame-MRA für eine besondere Klasse von Wavelets vor, die in Kapitel 2 eine wichtige Rolle spielen wird.

Sei \mathcal{B} eine abzählbare Untermenge von $\{B \in GL_d(\mathbb{R}) : |\det B| = 1\}$ und $\mathcal{A} = \{A^j : A \in GL_d(\mathbb{Z}), j \in \mathbb{Z}\}$ zu einer fest vorgegebenen Dilatationsmatrix A . Mit $GL_d(\mathbb{Z})$ bezeichnen wir dabei die Gruppe der invertierbaren Matrizen über den Körper \mathbb{Z} . Dann heißen die Wavelets der Gestalt

$$\psi_{j,i,k} := |\det A|^{j/2} \psi(B^i A^j \cdot -k) \quad j, i \in \mathbb{Z}, k \in \mathbb{Z}^d$$

mit $A \in \mathcal{A}, B \in \mathcal{B}$ *Wavelets mit zusammengesetzter Dilatation* oder auch *\mathcal{AB} -Wavelets* [LLKW05, Lim06, GLL⁺06]. Diese Wavelets sollen mit einer MRA verbunden werden, der eine spezielle Form zugrunde liegt:

Definition 1.14. Eine Folge $\{V_j\}_{j \in \mathbb{Z}}$ von abgeschlossenen Unterräumen des $L^2(\mathbb{R}^d)$ heißt *\mathcal{AB} -MRA*, wenn gilt:

(M1) $V_j \subset V_{j+1}$ für alle $j \in \mathbb{Z}$.

(M2) $f \in V_j \Leftrightarrow f(A \cdot) \in V_{j+1}$ für alle $j \in \mathbb{Z}$.

(M3) $\text{clos}_{L^2(\mathbb{R}^d)} \bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}^d)$.

(M4) $\bigcap_{j \in \mathbb{Z}} V_j = \{0\}$.

(M5) Für $f \in V_0$ gilt auch $f(B \cdot -k) \in V_0$ für alle $B \in \mathcal{B}, k \in \mathbb{Z}^d$.

(M6) Es gibt eine Funktion ϕ , so dass $\{\phi(B \cdot -k) : B \in \mathcal{B}, k \in \mathbb{Z}^d\}$ ein *Tight-Frame* von V_0 ist.

Der entscheidende Unterschied zur klassischen MRA ist die Tatsache, dass der Raum V_0 nicht nur invariant bezüglich der ganzzahligen Translationen sondern auch invariant bezüglich der Dilatationen mit $B \in \mathcal{B}$ ist. Wir nennen den Generator ϕ *\mathcal{AB} -Skalierungsfunktion* und die entsprechenden Wavelets – wie bereits erwähnt – *\mathcal{AB} -Wavelets*.

Wenn (M6) durch die schärfere ONB-Bedingung ersetzt wird, spricht man auch von einer *orthonormalen \mathcal{AB} -MRA* sowie von *orthonormalen \mathcal{AB} -Skalierungsfunktionen* bzw. *orthonormalen \mathcal{AB} -Wavelets*.

Typischerweise besteht die Matrixmenge \mathcal{B} aus Isometrien, die den \mathcal{AB} -Wavelets eine Orientierung verleihen. Wenn $\mathcal{B} = \{B_0, B_1, \dots, B_{r-1}\}$, $r \in \mathbb{N}$, eine endliche Menge ist, so kann man das Konzept der \mathcal{AB} -MRA auch aus der Perspektive einer durch einen verfeinerbaren Funktionenvektor erzeugten MRA betrachten. Dazu schreibt man die \mathcal{AB} -Skalierungsfunktionen in einen Vektor der Länge r

$$\Phi(\cdot) := (\phi(B_0\cdot), \phi(B_1\cdot), \dots, \phi(B_{r-1}\cdot))^T, \quad (1.12)$$

der nun wieder im herkömmlichen Sinne verfeinerbar ist, wie der folgende Paragraph näher erläutern wird.

Verallgemeinertes MRA-Konzept Im nächsten Kapitel werden wir das Konzept der vektorwertigen Wavelets (Multiwavelets) benötigen, das wir deshalb in diesem Abschnitt in aller Kürze vorstellen. Eine aus Funktionenvektoren erzeugte MRA wird auch *vektorwertige MRA* genannt. Weiterführende Untersuchungen über Multiwavelets findet man in [Kei04].

Sei $\Phi = (\phi_0, \dots, \phi_{r-1})^T$ ein Vektor von Funktionen $\phi_i \in L^2(\mathbb{R}^d)$, $i = 0, \dots, r-1$, mit der Vielfachheit r , der einer Matrix-Verfeinerungsgleichung genügt,

$$\Phi(x) = |\det A|^{1/2} \sum_{k \in \mathbb{Z}^d} M_k^0 \Phi(Ax - k), \quad x \in \mathbb{R}^d,$$

wobei $A \in \mathbb{Z}^{d \times d}$ wieder die Dilatationsmatrix ist, und $M_k^0 \in \mathbb{R}^{r \times r}$ Koeffizienten-Matrizen sind. Im Frequenzbereich hat die Verfeinerungsgleichung die Form

$$\hat{\Phi}(\omega) = H^0(\omega A^{-1}) \hat{\Phi}(\omega A^{-1}), \quad \omega \in \hat{\mathbb{R}}^d,$$

hierbei bezeichnet die $r \times r$ -Matrix H^0 das Symbol der Maske $\{M_k^0\}_{k \in \mathbb{Z}^d}$,

$$H^0(\omega) = \frac{1}{|\det A|^{1/2}} \sum_{k \in \mathbb{Z}^d} M_k^0 e^{-i\omega k}.$$

Der Fouriertransformierte Funktionenvektor $\hat{\Phi} = (\hat{\phi}_0, \dots, \hat{\phi}_{r-1})^T$ ist komponentenweise zu verstehen mit

$$\hat{\phi}_i(\omega) = \int_{\mathbb{R}^d} \phi_i(x) e^{-i\omega x} dx, \quad i = 0, \dots, r-1.$$

Jetzt sind wir in der Lage $L-1$ Waveletvektoren $\Psi^l = (\psi_0^l, \dots, \psi_{r-1}^l)^T$, $l = 1, \dots, L-1$, mit $L \geq m$ durch

$$\hat{\Psi}^l(\omega) = H^l(\omega A^{-1}) \hat{\Phi}(\omega A^{-1}), \quad \omega \in \hat{\mathbb{R}}^d,$$

zu definieren, wobei $H^l(\omega)$ geeignete 2π -periodische Matrixsymbole

$$H^l(\omega) = \frac{1}{|\det A|^{1/2}} \sum_{k \in \mathbb{Z}^d} M_k^l e^{-i\omega k} \quad (1.13)$$

der Waveletmaske $\{M_k^l\}_{k \in \mathbb{Z}}$ sind.

Sei nun $m = |\det A|$, und sei $\Gamma = \{\gamma_0, \dots, \gamma_{m-1}\}$ eine *Repräsentantenmenge* derart, dass das Gitter \mathbb{Z}^d in m disjunkte Co-Mengen $\mathbb{Z}_s^d = \{Ak + \gamma_s : k \in \mathbb{Z}^d\}$ zerlegt wird für $s = 0, \dots, m-1$, siehe [GM92] und [CHM04]. Die $rL \times rm$ -Matrix

$$\mathcal{M}(\omega) := [H^l(\omega + 2\pi\gamma^T A^{-1})]_{l=0, \gamma \in \Gamma}^{L-1}$$

heißt *Modulationsmatrix*. Die Symbole $H^l, l = 0, \dots, L-1$, in (1.13) können in m Polyphasen-Komponenten

$$H^l(\omega) = \frac{1}{|\det A|^{1/2}} \sum_{\gamma \in \Gamma} e^{-i\omega\gamma} H_\gamma^l(\omega A) \quad (1.14)$$

zerlegt werden, mit $H_\gamma^l(\omega) = \sum_{k \in \mathbb{Z}^d} M_{Ak+\gamma}^l e^{-i\omega k}$ für $\gamma \in \Gamma$. Die $rL \times rm$ -Matrix

$$\mathcal{P}(\omega) := [H_\gamma^l(\omega)]_{l=0, \gamma \in \Gamma}^{L-1} \quad (1.15)$$

heißt *Polyphasen-Matrix*. Sie ist *unitär*, wenn $\overline{\mathcal{P}(\omega)}^T \mathcal{P}(\omega) = I_{rm}$ gilt, wobei I_{rm} die Einheitsmatrix der Größe $rm \times rm$ bezeichnet. Wegen der besonderen Blockstruktur von $\mathcal{P}(\omega)$ ist die Unitarität von $\mathcal{P}(\omega)$ äquivalent zu

$$\sum_{l=0}^{L-1} \overline{H_\gamma^l(\omega)}^T H_{\gamma'}^l(\omega) = \delta_{\gamma, \gamma'} I_{r \times r}, \quad \forall \gamma, \gamma' \in \Gamma \quad (1.16)$$

$$\Leftrightarrow \sum_{l=1}^{L-1} \overline{H_\gamma^l(\omega)}^T H_{\gamma'}^l(\omega) = \delta_{\gamma, \gamma'} I_{r \times r} - \overline{H_\gamma^0(\omega)}^T H_{\gamma'}^0(\omega), \quad \forall \gamma, \gamma' \in \Gamma. \quad (1.17)$$

Eigenschaften von Skalierungsfunktionen und Wavelets In diesem Paragraphen verlassen wir wieder den vektorwertigen Fall und zählen kurz ein paar wünschenswerte Eigenschaften der Wavelets und der korrespondierenden Skalierungsfunktion auf.

Die Skalierungsfunktion ϕ besitzt die *Approximationsordnung* p , wenn für jedes f in dem Sobolevraum $W^p(L^2(\mathbb{R}^d))$

$$\text{dist}(f, V_j) := \min_{g \in V_j} \|f - g\|_2 = \mathcal{O}(v^{-pj})$$

gilt, vgl. [Ehl07]. Hier bezeichnet v den Spektralradius von A , d.h. den Betrag des betragsmäßig größten Eigenwertes von A . Im Falle einer ONB-MRA ist $P_{V_j} f$ die orthogonale L^2 -Projektion in den Approximationsraum $V_j, j \in \mathbb{Z}$, und es gilt $\text{dist}(f, V_j) = \|f - P_{V_j} f\|_2$.

Eine damit eng verbundene Eigenschaft ist die Polynomreproduktion durch die Translationen von ϕ . Die Skalierungsfunktion ϕ mit kompaktem Träger *reproduziert Polynome bis zum Grad* $p-1$, wenn $\Pi_{p-1} \subset V_0$ gilt.

Damit eng verwandt ist die Eigenschaft der verschwindenden Momente der zugehörigen Wavelets. Wenn nämlich ϕ eine Polynomreproduktion bis zum Grad $p-1$ garantiert, so folgt

aus der Orthogonalitätsbedingung $V_j \perp W_j$, dass die Wavelets p verschwindende Momente besitzen, d. h. für $l = 1, \dots, m - 1$ gilt:

$$\int_{\mathbb{R}^d} x^r \psi^l(x) dx = 0, \quad \text{für alle } 0 \leq r < p.$$

Insbesondere muss jedes Wavelet notwendigerweise ein erstes verschwindendes Moment haben: $\int_{\mathbb{R}^d} \psi^l(x) dx = 0$ oder $\hat{\psi}^l(0) = 0$. Also muss die Funktion oszillieren. Diese charakteristische Eigenschaft bezeichnet man auch als Auslöschungseigenschaft, sie hängt damit zusammen, dass man mit Wavelets Differenzen darstellt. In der Anwendung führt dieses Merkmal zu der wichtigen Dekorrelationseigenschaft, denn unter Verwendung von $\hat{\psi}^l(0) = 0$ ergibt sich aus (1.9) mit der Normierung $\hat{\phi}(0) = 1$ die Gleichung $H^l(0) = 0$ und damit auch

$$\sum_{k \in \mathbb{Z}^d} h_k^l = 0, \quad l = 1, \dots, m - 1.$$

Deswegen können die Koeffizienten h_k^l als Filterkoeffizienten eines Differenzfilters (Hochpass-Filters) interpretiert werden. Darauf gehen wir im nächsten Abschnitt näher ein.

Die Verfeinerbarkeit der Skalierungs- und Waveletfunktionen liefert in der Praxis eine Multiskalen-Zerlegung mittels eines Filterbank-Algorithmus.

Wie bereits oben erwähnt, setzen wir für ϕ und ψ^l , $l = 1, \dots, m - 1$, kompakte Träger voraus. Deshalb enthalten die Koeffizienten $\langle f, \phi_{j,k} \rangle$ und $\langle f, \psi_{j,k}^l \rangle$ lokale Informationen über die Funktion f . Je kleiner der Träger ist, desto größer ist die Lokalisierung im Ortsbereich. Auf den engen Zusammenhang zwischen der Trägergröße von ϕ bzw. ψ^l und den korrespondierenden Masken $\{h_k^0\}_{k \in \mathbb{Z}^d}$ bzw. $\{h_k^l\}_{k \in \mathbb{Z}^d}$ haben wir schon hingewiesen. Kleine Masken sind äußerst wünschenswert, weil sie in der Filterbank die Komplexität der Faltungsoperationen verringern, wie der nächste Unterabschnitt verdeutlichen wird. Wir werden sehen, dass unsere konstruierten Skalierungs- und Waveletfunktionen minimale Trägergröße aufweisen und deshalb eine effiziente Filterbank liefern.

Weil wir mit reellwertigen Bildern arbeiten, betrachten wir nur *reellwertige* Wavelets.

In der Bildverarbeitung sind in den letzten Jahren viele Waveletmethoden vorgeschlagen worden, die auf *nicht-separablen* Wavelets beruhen, die gegenüber den klassischen Tensorprodukt-Wavelets den Vorteil haben, dass sie Richtungseigenschaften besitzen. Diese wichtige Eigenschaft werden wir im Unterkapitel 1.4 näher untersuchen.

1.3 Filterbänke

Der große Erfolg der Wavelets rührt im Wesentlichen daher, dass die schnelle Wavelettransformation mittels Filterbänken berechnet werden kann. Nachdem Mallat die MRA-Struktur mit einer Pyramiden-Filterbank in Verbindung gebracht hatte, war man in der Lage, die

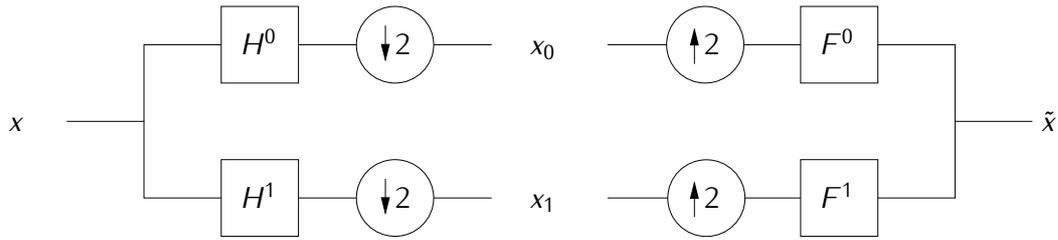


Abbildung 1.1: Die klassische 2-Kanal-Filterbank.

ganze Maschinerie der schon bekannten Filterbanktheorie mit ihren schnellen Zerlegungs- und Rekonstruktionsalgorithmen zu benutzen. In Folge dessen hat die Filterbank-Sichtweise wiederum fruchtbare Anregungen für die Wavelettheorie geliefert.

Wir versuchen in diesem Unterabschnitt, den engen Zusammenhang zwischen Wavelets und Filterbänken kurz vorzustellen. Ausführliche Untersuchungen lassen sich in den beiden Standardwerken [VK95] und [SN96] finden.

Klassische 2-Kanal-Filterbank

Definition 1.15. Ein Filter H ist eine Abbildung, die durch eine Folge von Filterkoeffizienten $\mathbf{h} = \{h[n]\}_{n \in \mathbb{Z}}$ gegeben ist:

$$H : l^\infty \rightarrow l^\infty, \quad \mathbf{x} \mapsto H(\mathbf{x}) = \mathbf{x} * \mathbf{h} =: \mathbf{y}.$$

Ein digitales Signal $\mathbf{x} = \{x[n]\}_{n \in \mathbb{Z}}$ wird mit einem Filter H gefiltert, wenn im Zeitbereich

$$y[k] = (\mathbf{x} * \mathbf{h})[k] = \sum_{j \in \mathbb{Z}} h[j] x[k - j]$$

gilt. Im Frequenzbereich hat die Filterung die Gestalt $\hat{Y}(\omega) = \hat{X}(\omega)H(\omega)$, wobei $H(\omega) = \sum_{j \in \mathbb{Z}} h[j] e^{-ij\omega}$ die Transferfunktion ist und $\hat{X}(\omega) = \sum_{j \in \mathbb{Z}} x[j] e^{-ij\omega}$ die Fouriertransformierte des diskreten Signals \mathbf{x} .

Das Downsampling eines digitalen Signals $\mathbf{x} = \{x[n]\}_{n \in \mathbb{Z}}$ mit dem Faktor 2 ist im Zeitbereich definiert durch

$$\mathbf{y} = (\downarrow 2)\mathbf{x}, \quad \text{bzw.} \quad y[k] = x[2k], \quad k \in \mathbb{Z},$$

bzw. durch $\hat{Y}(\omega) = \sum_{k \in \mathbb{Z}} x[2k] e^{-ik\omega} = \frac{1}{2} \left(\hat{X}\left(\frac{\omega}{2}\right) + \hat{X}\left(\frac{\omega}{2} + \pi\right) \right)$ im Frequenzbereich.

Das Upsampling eines digitalen Signals $\mathbf{x} = \{x[n]\}_{n \in \mathbb{Z}}$ mit dem Faktor 2 ist im Zeitbereich definiert durch

$$\mathbf{y} = (\uparrow 2)\mathbf{x}, \quad \text{bzw.} \quad y[2k] = x[k], y[2k + 1] = 0, \quad k \in \mathbb{Z},$$

bzw. durch $\hat{Y}(\omega) = \sum_{k \in \mathbb{Z}} x[k] e^{-i2k\omega} = \hat{X}(2\omega)$ im Frequenzbereich.

In einer 2-Kanal-Filterbank, wie sie in Abbildung 1.1 zu sehen ist, wird ein digitales Eingangssignal $x = \{x[n]\}_{n \in \mathbb{Z}}$ mit einem Tiefpass-Filter H^0 und mit einem Hochpass-Filter H^1 gefiltert. Die beiden resultierenden Signale werden durch ein anschließendes Downsampling mit dem Faktor 2 auf die Hälfte der Werte reduziert, so dass x in zwei Signale x_0 und x_1 halber Länge zerlegt wird. Diese Zerlegung nennt man auch *Signal-Analyse*. Zur Rekonstruktion des Signales (*Synthese*) werden die beiden Operationen „invers“ ausgeführt: Zuerst wird ein Upsampling durchgeführt, das die Signallänge durch Einfügen von Null-Werten verdoppelt, danach eine Filterung mit den Filtern F^0 bzw. F^1 . Das rekonstruierte Signal \tilde{x} soll möglichst dem Inputsignal x entsprechen, d. h. wir fordern *perfekte Rekonstruktion* mittels $\tilde{x} = x$. Diese Forderung bedeutet für das Zusammenspiel der vier Filter H^0, H^1, F^0 und F^1 , dass die beiden Bedingungen erfüllt sein müssen:

$$\begin{aligned} \text{Nichtverzerrung:} \quad & H^0(\omega)F^0(\omega) + H^1(\omega)F^1(\omega) = 2, \\ \text{Alias-Auslöschung:} \quad & H^0(\omega + \pi)F^0(\omega) + H^1(\omega + \pi)F^1(\omega) = 0. \end{aligned}$$

Wenn wir in diesen beiden Gleichungen für ω das verschobene Argument $\omega + \pi$ einsetzen, erhalten wir zwei weitere Gleichungen, die wir zusammenfassend in einer Matrixschreibweise ausdrücken können:

$$\mathcal{F}^T \mathcal{H} = 2I_2$$

$$\text{mit } \mathcal{F} = \begin{pmatrix} F^0(\omega) & F^0(\omega + \pi) \\ F^1(\omega) & F^1(\omega + \pi) \end{pmatrix} \text{ und } \mathcal{H} = \begin{pmatrix} H^0(\omega) & H^0(\omega + \pi) \\ H^1(\omega) & H^1(\omega + \pi) \end{pmatrix}.$$

Eine klassische Wahl der Rekonstruktionsfilter ist $\mathcal{F} = \overline{\mathcal{H}}$, dann reduziert sich die Forderung der perfekten Rekonstruktion zu $\overline{\mathcal{H}}^T \mathcal{H} = 2I_2$. Nun müssen nur noch H^0 und H^1 zueinander passend gewählt werden, ein üblicher Ansatz ist z. B. die Wahl der bereits erwähnten *Conjugate Quadrature Filter*

$$H^1(\omega) = e^{i\omega} \overline{H^0(\omega + \pi)}, \quad F^0(\omega) = \overline{H^0(\omega)}, \quad F^1(\omega) = \overline{H^1(\omega)} = e^{-i\omega} H^0(\omega + \pi).$$

Hier reduziert sich die Forderung der Verzerrungsfreiheit zu $|H^0(\omega)|^2 + |H^0(\omega + \pi)|^2 = 2$ und die gewünschte Eigenschaft der Alias-Auslöschung ist automatisch erfüllt!

m-Kanal-Waveletfilterbank In diesem Paragraphen werden wir sehen, dass die diskrete Wavelettransformation mittels einer Filterbank implementiert werden kann. Dabei ist der benötigte Tiefpass-Filter F^0 für die Rekonstruktion nichts anderes als das passend normierte Symbol der Skalierungsfunktion. Ebenso gewinnt man den Hochpass-Filter F^1 aus dem Waveletsymbol.

Da wir den d -dimensionalen Fall betrachten, haben wir mehrere Mutterwavelets, was zu mehreren Hochpass-Anteilen führt. Dementsprechend viele Hochpass-Kanäle besitzt die Filterbank. Statt des skalaren Down- und Upsampling-Faktors 2 verwenden wir die Dilationsmatrix A als Downsamplingmatrix. Wir erinnern daran, dass $m = |\det A|$ ist. Nun sei

wieder $\Gamma = \{\gamma_0, \dots, \gamma_{m-1}\}$ eine *Repräsentantenmenge* derart, dass das Gitter \mathbb{Z}^d in m disjunkte Co-Mengen $\mathbb{Z}_s^d = \{Ak + \gamma_s : k \in \mathbb{Z}^d\}$ zerlegt wird für $s = 0, \dots, m-1$, siehe [GM92] und [CHM04].

Sei nun ϕ eine Skalierungsfunktion mit kompaktem Träger, die eine MRA des $L^2(\mathbb{R}^d)$ erzeugt. Für ein festes $J \in \mathbb{Z}$ mit $J > 0$ erhalten wir die Zerlegung gemäß unseren Ausführungen im vorhergehenden Abschnitt

$$V_0 = V_{-J} \oplus \bigoplus_{j=-J}^{-1} W_j.$$

Das heißt, wir zerlegen eine Funktion $f \in V_0$ in einen groben Approximations-Anteil (Tiefpassanteil) in V_{-J} und die Detail-Anteile (Hochpassanteile), die in W_j enthalten sind:

$$f = \sum_{k \in \mathbb{Z}^d} \langle f, \phi_{-J,k} \rangle \phi_{-J,k} + \sum_{j=-J}^{-1} \sum_{l=1}^{m-1} \sum_{k \in \mathbb{Z}^d} \langle f, \psi_{j,k}^l \rangle \psi_{j,k}^l.$$

Wegen der Basisdarstellung genügt also die Berechnung der Tiefpass-Koeffizienten

$$\mathbf{a}_{-J} := \{a_{-J}[k]\}_{k \in \mathbb{Z}^d} \quad \text{mit} \quad a_{-J}[k] := \langle f, \phi_{-J,k} \rangle \quad (1.18)$$

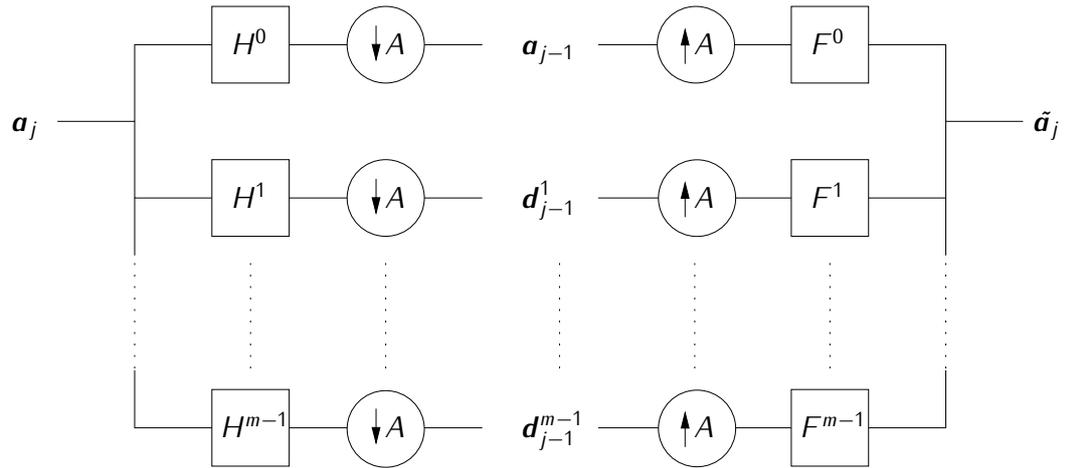
mit Hilfe der Skalierungsfunktion und die Berechnung der Hochpass-Koeffizienten in den Leveln $j = -1, \dots, -J$, für $l = 1, \dots, m-1$ mit Hilfe der Wavelets:

$$\mathbf{d}_j^l := \{d_j^l[k]\}_{k \in \mathbb{Z}^d} \quad \text{mit} \quad d_j^l[k] := \langle f, \psi_{j,k}^l \rangle. \quad (1.19)$$

Dies geschieht mittels eines Pyramiden-Algorithmus ausgehend von den gegebenen Koeffizienten $a_0[k] := \langle f, \phi_{0,k} \rangle$ im feinsten Level V_0 . In jedem Level $j = -1, \dots, -J$, wird der Tiefpassanteil \mathbf{a}_{j+1} des vorherigen Levels in die beiden Anteile \mathbf{a}_j und \mathbf{d}_j^l zerlegt. Die genaue Zerlegungsvorschrift erhalten wir, indem wir die Verfeinerungsgleichungen (1.4) und (1.8) in (1.18) bzw. (1.19) einsetzen:

$$\begin{aligned} a_{j-1}[k] &= \sum_{k' \in \mathbb{Z}^d} h^0[k' - Ak] a_j[k'] = (\mathbf{a}_j * \bar{\mathbf{h}}^0)[Ak] \\ d_{j-1}^l[k] &= \sum_{k' \in \mathbb{Z}^d} h^l[k' - Ak] a_j[k'] = (\mathbf{a}_j * \bar{\mathbf{h}}^l)[Ak], \quad l = 1, \dots, m-1, \end{aligned}$$

mit der Schreibweise $\bar{x}[k] = x[-k]$. Das entspricht nach dem vorherigen Paragraphen einer Filterung eines diskreten Signals \mathbf{a}_j mit einer Filterfolge $\mathbf{h}^l := \{h^l[k]\}_{k \in \mathbb{Z}^d}$, d.h. einer diskreten Faltung mit einem anschließenden Downsampling mit der Dilatationsmatrix A . Die Wavelettransformation kann also als eine Zerlegungsfiterbank interpretiert werden, wobei die Filterkoeffizienten genau die Koeffizienten $h^l[k] := h_k^l, k \in \mathbb{Z}^d$, aus den Verfeinerungsgleichungen sind, vgl. [Mal89]. Die Zerlegung (Analyse) des Signals \mathbf{a}_j in die $m-1$ Hochpassanteile \mathbf{d}_{j-1}^l für $l = 1, \dots, m-1$ und die Tiefpass-Koeffizienten \mathbf{a}_{j-1} ist für ein einzelnes Level schematisch in der linken Hälfte von Abbildung 1.2 dargestellt. Für eine Multilevel-Zerlegung wird dieses Schema iterativ auf den Tiefpassanteil angewandt, so dass

Abbildung 1.2: Analyse und Synthese einer m -Kanal-Filterbank in Modulations-Darstellung.

eine Pyramiden-Filterbank entsteht [BA83]. Auf diese Weise erhalten wir nach J Leveln einen Vektor von Koeffizienten der Skalierungs- und Waveletfunktionen:

$$[\mathbf{a}_{-J}, \mathbf{d}_{-J}, \mathbf{d}_{-J+1}, \dots, \mathbf{d}_{-1}], \quad (1.20)$$

hierbei enthält \mathbf{a}_{-J} die Approximationskoeffizienten im Level J gemäß (1.18) und $\mathbf{d}_j := \{\mathbf{d}_j^1, \dots, \mathbf{d}_j^{m-1}\}$ alle Waveletkoeffizienten im Level j gemäß (1.19).

Die rechte Seite von Abbildung 1.2 zeigt die entsprechende Rekonstruktion (Synthese) bei inverser Ausführung der Filterung, sie ist gegeben durch

$$\begin{aligned} a_j[k] &= \sum_{k' \in \mathbb{Z}^d} h^0[k - Ak'] a_{j-1}[k'] + \sum_{l=1}^{m-1} \sum_{k' \in \mathbb{Z}^d} h^l[k - Ak'] d_{j-1}^l[k'] \\ &= ((\uparrow A)\mathbf{a}_{j-1} * \mathbf{h}^0)[k] + ((\uparrow A)\mathbf{d}_{j-1}^l * \mathbf{h}^l)[k]. \end{aligned}$$

Wir können die Gleichungen auch in der für die Signalverarbeitung üblicheren Frequenz-Schreibweise formulieren. Dann hat die Zerlegung die Gestalt

$$\hat{a}_{j-1}(\omega A) = m^{-1/2} \sum_{\gamma \in \Gamma} \hat{a}_j(\omega + 2\pi\gamma^T A^{-1}) \overline{H^0(\omega + 2\pi\gamma^T A^{-1})} \quad (1.21)$$

$$\hat{d}_{j-1}^l(\omega A) = m^{-1/2} \sum_{\gamma \in \Gamma} \hat{a}_j(\omega + 2\pi\gamma^T A^{-1}) \overline{H^l(\omega + 2\pi\gamma^T A^{-1})}, \quad l = 1, \dots, m-1, \quad (1.22)$$

und die Rekonstruktion hat die Form

$$\hat{a}_j(\omega) = \hat{a}_{j-1}(\omega A) m^{1/2} H^0(\omega) + m^{1/2} \sum_{l=1}^{m-1} \hat{d}_{j-1}^l(\omega A) H^l(\omega). \quad (1.23)$$

Die Zerlegungsformeln (1.21) und (1.22) lassen sich mit Hilfe einer Matrix-Vektor-

Multiplikation schreiben:

$$\begin{pmatrix} \hat{d}_{j-1}(\omega A) \\ \hat{d}_{j-1}^1(\omega A) \\ \vdots \\ \hat{d}_{j-1}^{m-1}(\omega A) \end{pmatrix} = m^{-1/2} \overline{\mathcal{M}(\omega)} \begin{pmatrix} \hat{a}_j(\omega + 2\pi\gamma_0^T A^{-1}) \\ \hat{a}_j(\omega + 2\pi\gamma_1^T A^{-1}) \\ \vdots \\ \hat{a}_j(\omega + 2\pi\gamma_{m-1}^T A^{-1}) \end{pmatrix}. \quad (1.24)$$

Dabei ist $\mathcal{M}(\omega)$ die bereits oben erwähnte *Modulationsmatrix* der Größe $m \times m$, die durch

$$\mathcal{M}(\omega) := \begin{pmatrix} H^0(\omega + 2\pi\gamma_0^T A^{-1}) & H^0(\omega + 2\pi\gamma_1^T A^{-1}) & \dots & H^0(\omega + 2\pi\gamma_{m-1}^T A^{-1}) \\ H^1(\omega + 2\pi\gamma_0^T A^{-1}) & H^1(\omega + 2\pi\gamma_1^T A^{-1}) & \dots & H^1(\omega + 2\pi\gamma_{m-1}^T A^{-1}) \\ \vdots & \vdots & \ddots & \vdots \\ H^{m-1}(\omega + 2\pi\gamma_0^T A^{-1}) & H^{m-1}(\omega + 2\pi\gamma_1^T A^{-1}) & \dots & H^{m-1}(\omega + 2\pi\gamma_{m-1}^T A^{-1}) \end{pmatrix} \quad (1.25)$$

definiert ist. Analog können wir die Rekonstruktionsformel (1.23) durch die Gleichung

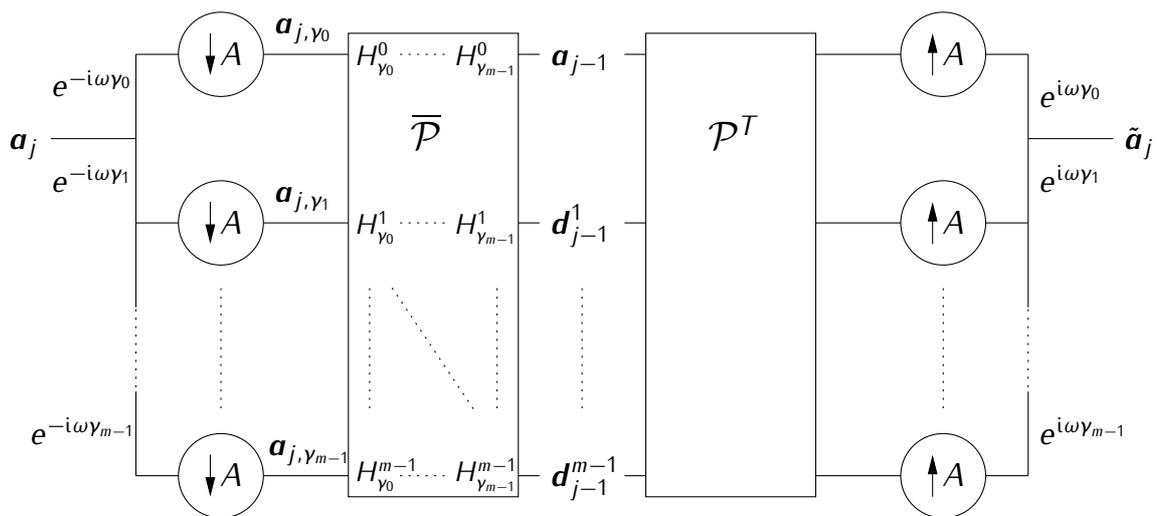
$$\begin{pmatrix} \hat{a}_j(\omega + 2\pi\gamma_0^T A^{-1}) \\ \hat{a}_j(\omega + 2\pi\gamma_1^T A^{-1}) \\ \vdots \\ \hat{a}_j(\omega + 2\pi\gamma_{m-1}^T A^{-1}) \end{pmatrix} = m^{1/2} \mathcal{M}(\omega)^T \begin{pmatrix} \hat{d}_{j-1}(\omega A) \\ \hat{d}_{j-1}^1(\omega A) \\ \vdots \\ \hat{d}_{j-1}^{m-1}(\omega A) \end{pmatrix} \quad (1.26)$$

ausdrücken, wenn wir in (1.23) anstelle von ω das Argument $\omega + 2\pi\gamma_i^T A^{-1}$ für $\gamma_i \in \Gamma$ einsetzen (Beachte die 2π -Periodizität der Fouriertransformierten \hat{a}_j und \hat{d}_j^l). Um perfekte Rekonstruktion zu gewährleisten, muss die Modulationsmatrix unitär sein, wie ein Einsetzen von (1.24) in (1.26) sofort verdeutlicht.

Polyphasen-Filterbank In der Filterbank-Theorie bevorzugt man statt der Modulations-Formulierung die Polyphasen-Darstellung, die eine wesentlich effizientere Filterbank liefert. Aus Abbildung 1.2 wird klar, dass man das Eingangssignal filtert und es *danach* mittels der Downsampling-Matrix A dezimiert. Man berechnet also eine große Anzahl von Werten, die man danach sofort wieder verwirft. Wir können diese (unnötige) Redundanz auch anhand der Modulationsmatrix in (1.25) feststellen: Die Matrix $\mathcal{M}(\omega)$ enthält die gesamte Information bereits in der ersten Spalte, die weiteren $m - 1$ Spalten bestehen lediglich aus den Translationen der Symbole. Die Polyphasen-Filterbank in Abbildung 1.3 splittet zunächst das Signal in m verschiedene Teilsignale (Polyphasen) und wendet die Filterung nur noch auf diese Polyphasen an. Dadurch verringert sich die arithmetische Komplexität der Filterbank.

Die Anzahl der Polyphasen hängt von der Matrix A ab. Und zwar lassen sich die Symbole H^l , $l = 0, \dots, m - 1$, in (1.6) bzw. (1.10) in $|\Gamma| = |\det A| = m$ Polyphasen-Komponenten

$$H_\gamma^l(\omega) = \sum_{k \in \mathbb{Z}^d} h_{Ak+\gamma}^l e^{-i\omega k}, \quad \gamma \in \Gamma$$

Abbildung 1.3: Analyse und Synthese einer m -Kanal-Filterbank in Polyphasen-Darstellung.

zerlegen:

$$H^l(\omega) = \frac{1}{m^{1/2}} \sum_{\gamma \in \Gamma} e^{-i\omega\gamma} H_{\gamma}^l(\omega A). \quad (1.27)$$

Die $m \times m$ -Matrix

$$\mathcal{P}(\omega) := [H_{\gamma}^l(\omega)]_{l=0, \gamma \in \Gamma}^{m-1}$$

heißt *Polyphasen-Matrix*. Sie ist *unitär*, wenn $\overline{\mathcal{P}(\omega)}^T \mathcal{P}(\omega) = I_m$ gilt. Die Filterbank garantiert eine perfekte Rekonstruktion des Signals, wenn die Polyphasen-Matrix unitär ist.

Bemerkung 1.16. 1. Beachte, dass man von *paraunitär* spricht, wenn man die z -Notation verwendet (wie es üblicherweise in der Sprache der Filterbänke getan wird). Eine Matrix $\mathcal{P}(z)$ heißt *paraunitär*, wenn sie für alle z auf dem Einheitskreis unitär ist, d. h. für $z = e^{i\omega}$.

2. Wenn ϕ bzw. ψ nicht normiert sind, ist die (fast) Unitarität von \mathcal{P} definiert durch $\overline{\mathcal{P}(\omega)}^T \mathcal{P}(\omega) = c I_m$ für $c \in \mathbb{Z}$.

1.4 Bildverarbeitung

Eine der erfolgreichsten Anwendungsgebiete für die diskrete Wavelettransformation ist die digitale Bildverarbeitung. Burt und Adelson schlugen 1983 einen Pyramiden-Algorithmus vor, der digitale Bilder in Approximations- und Differenzenbild zerlegte [BA83]. In den bereits zitierten Arbeiten [Mal89, Mey90] übertrugen Mallat und Meyer diese Grundidee auf die Funktionalanalysis und führten damit die fundamentale MRA-Struktur in der Wavelettheorie ein. Die Wavelettheorie ist also vom Ursprung her aufs Engste mit der Bildverarbeitung verknüpft.

Ein quadratisches, digitales Graustufenbild ist durch eine $N \times N$ -Matrix $\mathbf{a} = (a_{m_1, m_2})_{(m_1, m_2) \in I}$ mit der Indexmenge $I = \{0, \dots, N-1\}^2$ gegeben. Der ganzzahlige Wert a_{m_1, m_2} gibt den Grauwert des Bildes im Pixel mit den Koordinaten (m_1, m_2) wieder. Im Fall von 8-Bit-Bildern liegen die Grauwerte zwischen 0 und 255, wobei 0 ein schwarzes Pixel und 255 ein weißes Pixel darstellt.

Die beiden vielleicht wichtigsten Aufgaben in der Bildverarbeitung sind die Kompression und die Entstörung von digitalen Bildern. Wir werden beide Anwendungen kurz beschreiben, weil wir im Verlauf der Arbeit immer wieder darauf zurückgreifen werden.

Kompression Die Kompression von digitalen Bildern besteht aus drei Schritten: Zunächst wird das Bild mittels einer Transformation in eine „günstigere“ Form gebracht, danach wendet man eine Quantisierung an, bei der vernachlässigbare Bildinformation verworfen wird. Schließlich werden die verbleibenden Werte (binär) kodiert.

Wir werden uns in dieser Arbeit auf die Betrachtung der ersten beiden Schritte beschränken und den Kodierungsschritt, bei dem die eigentliche Speicherplatzreduktion vorgenommen wird, außen vor lassen. Deshalb sprechen wir weniger von Bildkompression als vielmehr von *Bildapproximation* oder von einer *dünnen Darstellung von Bildern*.

Die Transformation soll das Bild so umformen, dass möglichst viel Bildinformation in möglichst wenigen Werten enthalten ist. Dazu werden wir verschiedene Ideen für gerichtete Wavelettransformationen vorschlagen. Wenn die zugrunde liegenden Wavelets in der Lage sind, orientierte Bildkanten gut zu approximieren, führt das dazu, dass wir mit wenigen Waveletkoeffizienten diese Bildkanten darstellen können. Wie im vorigen Unterkapitel beschrieben, suchen wir zunächst eine geeignete Skalierungsfunktion ϕ und konstruieren daraus gerichtete Wavelets $\psi^l, l = 1, \dots, m-1$. Damit berechnen wir die Filterkoeffizienten

$$\begin{aligned} h^0[k] &= \langle m^{-1/2} \phi(\cdot A^{-1}), \phi(\cdot - k) \rangle \\ h^l[k] &= \langle m^{-1/2} \psi^l(\cdot A^{-1}), \phi(\cdot - k) \rangle, \quad l = 1, \dots, m-1, \end{aligned}$$

für die Tief- und Hochpassbänder der Filterbank. Mit dem digitalen Bild \mathbf{a} als Eingangssignal zerlegen wir das Bild und erhalten als Output die Koeffizienten gemäß (1.20)

$$[\mathbf{a}_{-J}, \mathbf{d}_{-J}, \mathbf{d}_{-J+1}, \dots, \mathbf{d}_{-1}],$$

wobei wir von den Hochpass-Koeffizienten in $\mathbf{d}_{-J}, \mathbf{d}_{-J+1}, \dots, \mathbf{d}_{-1}$ erwarten, dass viele verschwinden oder zumindestens betragsmäßig sehr klein sind.

Im Quantisierungsschritt unterwerfen wir die Waveletkoeffizienten einer Shrinkage-Funktion. Um eine beste M -Term-Bildapproximation zu erreichen, benutzt man das so genannte *globale Hard-Shrinkage*, bei dem die betragsmäßig kleinen Koeffizienten zu Null gesetzt werden:

$$S_\lambda(x) = \begin{cases} x, & |x| \geq \lambda, \\ 0, & |x| < \lambda, \end{cases}$$

wobei der Schwellwert $\lambda > 0$ so gewählt wird, dass nur noch die M betragsmäßig größten Koeffizienten $d_j^l[k]$ übrig bleiben (wobei auch die Tiefpass-Koeffizienten $a_j[k]$ mitgezählt werden müssen). Man erhält also die Koeffizienten

$$\tilde{\mathbf{d}}_j := \left\{ \{\tilde{d}_j^1[k]\}_{k \in \mathbb{Z}^d}, \dots, \{\tilde{d}_j^{m-1}[k]\}_{k \in \mathbb{Z}^d} \right\} \quad \text{mit} \quad \tilde{d}_j^l[k] := S_\lambda(d_j^l[k]). \quad (1.28)$$

Das aus $[\mathbf{a}_{-J}, \tilde{\mathbf{d}}_{-J}, \tilde{\mathbf{d}}_{-J+1}, \dots, \tilde{\mathbf{d}}_{-1}]$ rekonstruierte Bild $\tilde{\mathbf{a}}$ ist dann die beste M -Term-Approximation des Originalbildes \mathbf{a} .

Entstörung Auch die Bildentstörung mittels Waveletmethoden basiert auf einer Transformation mit anschließendem Shrinkage.

Wir betrachten ein Bild \mathbf{a} , das mit einem weißen Gauß'schen Rauschen gestört wird

$$\mathbf{u} = \mathbf{a} + \mathbf{v},$$

dabei ist die additive Störung $\mathbf{v} = (v_{m_1, m_2})_{(m_1, m_2) \in I}$ eine Zufallsmatrix mit identisch, unabhängig und normalverteilten Zufallsvariablen v_{m_1, m_2} mit dem Erwartungswert $\mu = 0$ und der Varianz σ^2 . Bei der Multiskalenzerlegung des beobachteten Bildes \mathbf{u} stellt man fest, dass die wesentlichen Bildkanten von \mathbf{a} in den groben Differenzräumen enthalten sind, während die Störung in den feinen Waveleträumen zu finden ist. Durch Elimination dieser Waveletkoeffizienten besteht die Hoffnung, das ungewünschte Rauschen zu minimieren. In [DJ94] haben Donoho und Johnstone bewiesen, dass ein fast optimaler Schätzer $\tilde{\mathbf{a}}$ für das Originalbild \mathbf{a} auf Grundlage des Bildes \mathbf{u} mittels des Hard-Shrinkage wie in (1.28) gefunden werden kann, wobei als globaler Schwellwert $\lambda = \sigma\sqrt{2\log(N^2)}$ zu wählen ist. Neben dem Hard-Thresholding haben sich aber auch andere Shrinkage-Methoden als erfolgreich erwiesen. Eine ausführliche Untersuchung über die theoretischen und numerischen Eigenschaften einer Vielzahl von Shrinkage-Funktionen findet man in [FK03].

Bei der Bildapproximation bevorzugt man in der Regel Waveletbasen gegenüber -frames, weil die Redundanz eines Frames der Absicht, die Bildinformation in nur möglichst wenig Koeffizienten zu konzentrieren, entgegen wirkt. Dagegen sind Waveletframes bei der Bildentstörung gegenüber Waveletbasen im Vorteil, weil redundante Information größere Hoffnung gibt, die verrauschten Bildwerte zu rekonstruieren.

PSNR-Wert Ein Maß für die Qualität des rekonstruierten Bildes ist der PSNR-Wert (engl. *peak signal-to-noise ratio*). Da viele Signale einen großen Dynamikumfang (Quotient zwischen maximalem und minimalem Signalwert) aufweisen, wird der PSNR-Wert in der logarithmischen Skala dargestellt. Er wird mithilfe der mittleren quadratischen Abweichung (MSE, engl. *mean square error*) definiert, die für ein Originalbild \mathbf{a} und eine Approximation $\tilde{\mathbf{a}}$ durch

$$MSE = MSE(\mathbf{a}, \tilde{\mathbf{a}}) = \frac{1}{N^2} \sum_{(m_1, m_2) \in I} |a_{m_1, m_2} - \tilde{a}_{m_1, m_2}|^2$$

beschrieben wird. Der PSNR-Wert ist dann definiert durch

$$PSNR = PSNR(\mathbf{a}, \tilde{\mathbf{a}}) = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right).$$

Je näher das rekonstruierte Bild $\tilde{\mathbf{a}}$ dem ursprünglichen Bild \mathbf{a} kommt, desto größer wird der PSNR-Wert, weil die MSE sinkt. Wir streben also stets einen hohen PSNR-Wert an; als Faustregel gilt, dass ein Bild ab einem PSNR-Wert von 30 dB akzeptabel ist. Allerdings ist der PSNR-Wert nur ein mathematisches Maß und beschreibt nicht in notwendiger Weise die visuelle Qualität des Bildes, wie das menschliche Auge sie wahrnimmt. Die Verwendung des PSNR-Werts macht nur Sinn, um verschiedene Approximationen miteinander zu vergleichen. Im optimalen Fall der perfekten Rekonstruktion ist die MSE wegen $\tilde{\mathbf{a}} = \mathbf{a}$ gleich Null und der PSNR-Wert unendlich.

Separabel versus nicht-separabel Es ist naheliegend, den multivariaten Fall auf den einfachen, univariaten Fall zurückzuführen, indem man eine eindimensionale Wavelettransformation in jeder Koordinatenrichtung durchführt. Sei ϕ_{1D} eine eindimensionale Skalierungsfunktion und ψ_{1D} das dazugehörige Wavelet. Für $d = 2$ definiert man die zweidimensionale Skalierungsfunktion über das Tensorprodukt

$$\phi(x_1, x_2) = \phi_{1D}(x_1) \phi_{1D}(x_2). \quad (1.29)$$

Diese zweidimensionale Skalierungsfunktion generiert mit der Dilatationsmatrix $A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$ eine MRA, mit der $|\det A| - 1 = 3$ Wavelets verknüpft sind. Und zwar sind die Wavelets gegeben durch

$$\psi^1(x_1, x_2) = \psi_{1D}(x_1) \psi_{1D}(x_2), \quad (1.30)$$

$$\psi^2(x_1, x_2) = \psi_{1D}(x_1) \phi_{1D}(x_2), \quad (1.31)$$

$$\psi^3(x_1, x_2) = \phi_{1D}(x_1) \psi_{1D}(x_2). \quad (1.32)$$

Die dadurch erzeugte Basis erfüllt die oben genannten MRA-Eigenschaften, insbesondere überträgt sich die Verfeinerbarkeit der Funktionen. Zweidimensionale Skalierungs- und Waveletfunktionen der Art (1.29) bzw. (1.30) bis (1.32) nennt man *separabel*.

Der Vorteil dieses Ansatzes ist die Einfachheit der Konstruktion mittels eindimensionaler MRAs, was sehr einfache Filterbank-Algorithmen nach sich zieht. Bei einem digitalen Bild kann eine eindimensionale Wavelettransformation getrennt auf die Zeilen und Spalten angewendet werden. Ein Nachteil ist, dass drei Wavelets benötigt werden, um das orthogonale Komplement von V_0 in V_1 aufzuspannen. Der wesentliche Nachteil der separablen Wavelets liegt jedoch in der unzureichenden Richtungssensitivität. Die drei Wavelets bevorzugen die x_1 - und x_2 -Richtung sowie die Diagonale. Um anders gerichtete Bildkanten in digitalen Bildern besser zu erkennen oder zu approximieren, benötigen wir Wavelets mit mehreren Orientierungen und anisotropen Trägern.

Ein klassisches Beispiel ist die Verwendung der anisotropen Quincunx-Matrix $A = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, die wegen $|\det A| = 2$ nur ein einziges Mutterwavelet benötigt, um eine ONB für den $L^2(\mathbb{R}^d)$ zu erzeugen. Die Wavelets

$$\psi_{1,k}(x_1, x_2) = \psi(x_1 + x_2 - k_1, x_1 - x_2 - k_2)$$

mit $k = (k_1, k_2) \in \mathbb{Z}^2$ sind nicht-separabel. Weitere nicht-separable Waveletmethoden findet man u. a. in [KV95, HL97, Rue02], dabei möchten wir insbesondere die bivariaten Box-Splines hervorheben [dBHR93, HL99]. Auch die gerichteten Curvelet- und Shearlet-Frames bestehen aus nicht-separablen Funktionen. Die Curvelets besitzen gemäß Konstruktion einen kompakten Träger auf Kreisscheibensegmenten in der Frequenzebene. Diese Tatsache sowie die Verwendung der Rotationsmatrix bewirkt, dass Curvelets nicht-separabel sind [CD04]. Ähnliches gilt für die Shearlets: Sie verwenden die zusammengesetzte, anisotrope Dilatationsmatrix

$$AB = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 4 \\ 0 & 2 \end{pmatrix},$$

und basieren auf dem im Frequenzbereich konstruierten, nicht-separablen Muttershearlet

$$\hat{\psi}(\omega) = \hat{\psi}(\omega_1, \omega_2) = \hat{\psi}_1(\omega_1) \hat{\psi}_2\left(\frac{\omega_2}{\omega_1}\right)$$

mit den eindimensionalen Wavelets $\hat{\psi}_1, \hat{\psi}_2 \in C^\infty(\hat{\mathbb{R}})$ für $\omega = (\omega_1, \omega_2) \in \hat{\mathbb{R}}^2, \omega_1 \neq 0$. Hierbei sollen $\hat{\psi}_1$ und $\hat{\psi}_2$ kompakten Träger haben, siehe [GL07].

Die Wavelets, die wir im folgenden konstruieren werden, sind alle nicht-separabel. Trotzdem liefern sie ähnlich schnelle Filterbank-Algorithmen wie im Falle von Tensorprodukt-Wavelets.

Kapitel 2

Orientierte Haarwavelet-Systeme auf Dreiecken

In diesem Kapitel konstruieren wir verschiedene Systeme von Haarwavelets auf Dreiecksgebieten, die wir in der Bildverarbeitung anwenden möchten. Sie alle haben gemeinsam, dass sie unmittelbar mit einer MRA-Struktur verbunden sind, dass sie nicht-separabel sind sowie einen kleinen und kompakten Träger im Zeitbereich (bzw. Ortsbereich) besitzen. Das führt zu einem guten Umgang mit gerichteten Bildkanten und zu einfachen Filterbank-Algorithmen.

Die verschiedenen Waveletsysteme unterscheiden sich in ihrer Redundanz. Ausgehend von einer 4-Richtungs-Triangulierung der reellen Ebene konstruieren wir zunächst eine gerichtete Haarwaveletbasis des $L^2(\mathbb{R}^2)$. Danach erweitern wir im Unterkapitel 2.2 das System um vier weitere Diagonalrichtungen, so dass ein Parseval-Waveletframe entsteht. Die konstruierten, gerichteten Wavelets gehören zu der Klasse der Wavelets mit zusammengesetzter Dilatation [GLL⁺06, Lim06]. Schließlich betrachten wir ein Shearletsystem mit beliebig vielen Richtungen, wobei die Anzahl der Orientierungen durch die Anzahl der Zerlegungslevel beeinflusst werden kann. Das Kapitel klingt mit einem Ausblick aus, indem wir stetige Wavelets auf den Dreiecksgebieten untersuchen.

2.1 Waveletbasen mit vier Richtungen

Bei der klassischen, zweidimensionalen Tensorprodukt-Wavelettransformation werden die beiden Achsenrichtungen bevorzugt. In unserem Dreiecksansatz nehmen wir zwei Diagonalrichtungen hinzu. Die Skalierungsfunktionen sind charakteristische Funktionen auf Dreiecken und liefern uns auf kanonischem Wege ein Waveletsystem, das eine ONB des $L^2(\mathbb{R}^2)$ bildet.

Im Wesentlichen sind diese Ergebnisse unserer Arbeit [Kro09a] entnommen. Unabhängig davon haben Kristhal et al. einen sehr ähnlichen Ansatz gewählt, um Haarwavelets mit zusammengesetzter Dilatation auf Dreiecken zu konstruieren [KRWW07]. Allerdings verwenden sie die Quincunx-Matrix als Dilatationsmatrix. Wir bevorzugen dagegen die Dilatation $A = 2I_2$,

um einfache Zerlegungs- und Rekonstruktionsalgorithmen zu gewährleisten, die wir in der Bildverarbeitung effizient einsetzen können.

2.1.1 Konstruktion von Skalierungs- und Waveletfunktionen

Wir betrachten das quadratische Gebiet $\Omega := [-\frac{1}{2}, \frac{1}{2}]^2$ und zerlegen es in acht Dreiecke gleichen Flächeninhalts wie in Abbildung 2.1(a) dargestellt. Dazu sei

$$U_0 = \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \begin{pmatrix} 0 \\ 1/2 \end{pmatrix} \right\} := \left\{ x \in \mathbb{R}^2 : 0 \leq x_2 \leq \frac{1}{2}, 0 \leq x_1 \leq x_2 \right\}$$

das Dreieck mit den Eckpunkten $\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}, \begin{pmatrix} 0 \\ 1/2 \end{pmatrix}$. Die übrigen Dreiecke erhalten wir durch Rotation bzw. Spiegelung, es gilt also $U_i := \{B_i^{-1}x : x \in U_0\} = B_i^{-1}U_0$ für $i = 0, \dots, 7$, wobei

$$\mathcal{B} := \{B_i : i = 0, \dots, 7\} \quad (2.1)$$

die Gruppe der Isometrien ist, mit

$$B_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, B_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, B_2 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, B_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

$$B_4 = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}, B_5 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}, B_6 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, B_7 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Diese Zerlegung weist neben den beiden Achsenrichtungen auch noch zwei Diagonalrichtungen auf, was im Folgenden zu insgesamt vier Orientierungen der Skalierungs- bzw. Waveletfunktionen führt. Durch \mathbb{Z}^2 -Translation von Ω entsteht ein Gitter, das eine spezielle Form der Triangulierung der reellen Ebene darstellt, die in der Literatur unter dem Begriff *triangulierte Quadrangulierung* (gleichmäßige Quadraterlegung, deren Quadrate durch zwei Diagonalen trianguliert werden) bekannt ist, siehe z. B. [GH06, LS07].

Wir definieren auf den acht Dreiecken in Ω nicht-separable Skalierungsfunktionen. Dabei liegt keine Verfeinerbarkeit im herkömmlichen Sinne vor, weil für die Verfeinerung auch die mit $B_i \in \mathcal{B}$ gespiegelten bzw. gedrehten Versionen der Funktionen benötigt werden. Das bedeutet, dass es sich bei unseren Funktionen um orthonormale \mathcal{AB} -Skalierungsfunktionen handeln wird, wobei $\mathcal{A} = \{A^j : j \in \mathbb{Z}\}$ für $A = 2I_2$ ist, vgl. [GLL⁺06, Lim06]. Es sei

$$\phi_0(x) = \phi_0(x_1, x_2) := \sqrt{8} \chi_{U_0}(x_1, x_2) = \sqrt{8} \chi_{[0,1]} \left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right) \chi_{[0, \frac{1}{2}]}(x_2), \quad (2.2)$$

wobei der Skalierungsfaktor $\sqrt{8}$ für die Normierung $\|\phi_0\|_2^2 = 1$ sorgt. Weiter sei

$$\phi_i(x) := \phi_0(B_i x) = \sqrt{8} \chi_{U_0}(B_i x) = \sqrt{8} \chi_{B_i^{-1}U_0}(x) = \sqrt{8} \chi_{U_i}(x), \quad i = 0, \dots, 7.$$

Nehmen wir jetzt auch Translationen $k \in \mathbb{Z}^2$ und Skalierungen $2^j, j \in \mathbb{Z}$, hinzu, so erhalten

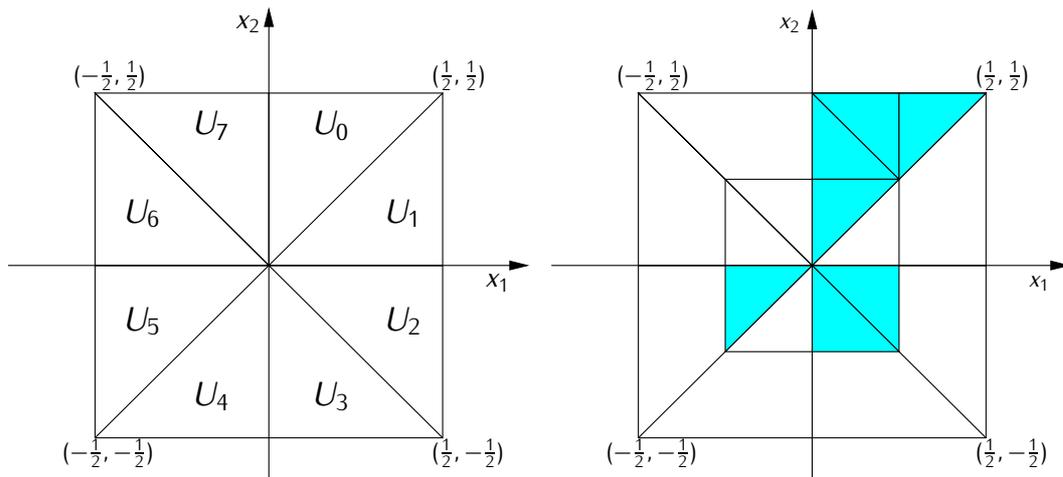


Abbildung 2.1: Dreieckszerlegung mit vier Richtungen. (a) Level V_0 , (b) Verfeinerung von U_0 .

wir die Funktionen mit zusammengesetzter Dilatation

$$\phi_{i,j,k}(x) := 2^j \phi_0(B_i(2^j x - k)), \quad i = 0, \dots, 7, j \in \mathbb{Z}, k \in \mathbb{Z}^2, \quad (2.3)$$

wobei der Faktor 2^j die \mathcal{AB} -Skalierungsfunktionen normiert, so dass $\langle \phi_{i,j,k}, \phi_{i',j',k'} \rangle = \delta_{i,i'} \delta_{k,k'}$ für alle $j \in \mathbb{Z}$ gilt. Wegen der „verfeinerbaren“ Träger-Dreiecke von $\phi_{i,j,k}$ sind auch die \mathcal{AB} -Skalierungsfunktionen verfeinerbar. Es gilt z. B.

$$\begin{aligned} \phi_0(x) &= \phi_0(2x) + \phi_2(2x - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_3(2x - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_5(2x - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \\ &= \frac{1}{2} \left(\phi_{0,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}}(x) + \phi_{2,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}}(x) + \phi_{3,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}}(x) + \phi_{5,1,\begin{pmatrix} 1 \\ 1 \end{pmatrix}}(x) \right), \end{aligned} \quad (2.4)$$

siehe dazu Abbildung 2.1(b). Analoge Verfeinerungsgleichungen gelten für alle \mathcal{AB} -Skalierungsfunktionen; allgemein gilt im Level $j \in \mathbb{Z}$ für $i = 0, \dots, 7$ und $k \in \mathbb{Z}^2$:

$$\begin{aligned} \phi_{i,j,k} &= \frac{1}{2} \left(\phi_{i,j+1,2k} + \phi_{(i+2(-1)^i) \bmod 8, j+1, 2k + B_i^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{(i+3(-1)^i) \bmod 8, j+1, 2k + B_i^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \right. \\ &\quad \left. + \phi_{(i+5(-1)^i) \bmod 8, j+1, 2k + B_i^{-1} \begin{pmatrix} 1 \\ 1 \end{pmatrix}} \right), \end{aligned}$$

wobei $a \bmod b = a - [a/b] \cdot b$ ist ($a, b \in \mathbb{R}$); dabei ist $[c]$ die größte ganze Zahl, die kleiner oder gleich c ist ($c \in \mathbb{R}$).

Diese Eigenschaften führen zusammenfassend zu

Satz 2.1. Die Folge $\{V_j\}_{j \in \mathbb{Z}}$ der Approximationsräume

$$V_j := \text{clos}_{L^2(\mathbb{R}^2)} \text{span}\{\phi_{i,j,k} : i = 0, \dots, 7; k \in \mathbb{Z}^2\}$$

bildet eine orthonormale \mathcal{AB} -MRA des $L^2(\mathbb{R}^2)$. Für alle $j \in \mathbb{Z}$ wird V_j durch eine ONB $\{\phi_{i,j,k} : i = 0, \dots, 7; k \in \mathbb{Z}^2\}$ aufgespannt.

Diese orthonormale \mathcal{AB} -MRA kann in einer alternativen Betrachtungsweise auch als vektorwertige MRA aufgefasst werden. Dazu betrachtet man wie in (1.12) beschrieben den Funktionenvektor

$$\Phi := (\phi_0, \phi_1, \dots, \phi_7)^T, \quad (2.5)$$

der die gedrehten und gespiegelten Versionen von ϕ_0 beinhaltet und deshalb im herkömmlichen Sinn verfeinerbar ist.

Außerdem sei erwähnt, dass die Dilatationsmatrix $A = 2I_2$ isotrop ist, das bedeutet, dass sie auf den Nebendiagonalen nur Null-Einträge hat. Wir stellen also fest, dass die Nicht-Separabilität unserer Skalierungsfunktionen allein auf die Dreiecksträger zurückzuführen ist (siehe (2.2)), und nicht auf eine anisotrope Dilatationsmatrix, wie es bei Curvelets oder Shearlets der Fall ist.

Ausgehend von der Verfeinerungsgleichung der \mathcal{AB} -Skalierungsfunktionen konstruieren wir orthonormale Haarwavelets auf den Dreiecken, die die Richtungssensitivität der \mathcal{AB} -Skalierungsfunktionen erben, weil sie die gleichen Träger besitzen. Betrachten wir zunächst wieder nur das Dreieck U_0 , so erhalten wir drei \mathcal{AB} -Wavelets ψ_0^l , $l = 1, 2, 3$, aus der Verfeinerungsgleichung von ϕ_0 in (2.4)

$$\begin{pmatrix} \phi_0 \\ \psi_0^1 \\ \psi_0^2 \\ \psi_0^3 \end{pmatrix} := \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} \phi_{0,1} \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \phi_{2,1} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \phi_{3,1} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \phi_{5,1} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \end{pmatrix}, \quad (2.6)$$

siehe Abbildung 2.2. Die erste Zeile der Verfeinerungsgleichung entspricht dabei genau (2.4). Mit Hilfe der Spiegelungen/Rotationen $B_i \in \mathcal{B}$, der Translationen $k \in \mathbb{Z}^2$ und der Skalierungen 2^j , $j \in \mathbb{Z}$, haben die nicht-separablen, gerichteten \mathcal{AB} -Haarwavelets die Gestalt

$$\psi_{i,j,k}^l(x) := 2^j \psi_0^l(B_i(2^j x - k)), \quad l = 1, 2, 3.$$

Die \mathcal{AB} -Wavelets $\psi_{i,j,k}^l$, $l = 1, 2, 3$, besitzen den gleichen Träger wie die entsprechende \mathcal{AB} -Skalierungsfunktion $\phi_{i,j,k}$ für alle i, j, k . Für die Waveleträume

$$W_j := \text{clos}_{L^2(\mathbb{R}^2)} \text{span}\{\psi_{i,j,k}^l : l = 1, 2, 3; i = 0, \dots, 7; k \in \mathbb{Z}^2\}$$

können wir alle klassischen Resultate von der zweidimensionalen Tensorwavelet-Transformation auf unseren nicht-separablen Fall übertragen. Es gilt nach Konstruktion $W_j \subset V_{j+1}$. Außerdem garantiert die Haarwavelet-Transformationsmatrix in (2.6) die Orthogonalität der Räume $V_j \perp W_j$ und die Invertierbarkeit $V_{j+1} \subset V_j + W_j$ (Rekonstruktionsformeln). Insgesamt erhalten wir also für $j \in \mathbb{Z}$ die Beziehung

$$V_{j+1} = V_j \oplus W_j,$$

so dass wir das angekündigte Resultat formulieren können:

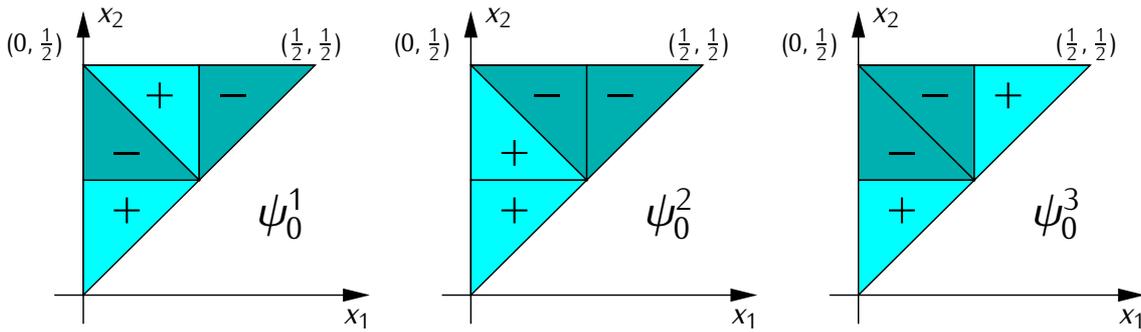


Abbildung 2.2: Konstruktion der gerichteten AB-Mutterwavelets ψ_0^1, ψ_0^2 und ψ_0^3 .

Satz 2.2. Es gilt $\bigoplus_{j \in \mathbb{Z}} W_j = L^2(\mathbb{R}^2)$, d. h. das System der gerichteten AB-Haarwavelets auf Dreiecken

$$\{\psi_{i,j,k}^l : j \in \mathbb{Z}; l = 1, 2, 3; i = 0, \dots, 7; k \in \mathbb{Z}^2\}$$

bildet eine ONB des $L^2(\mathbb{R}^2)$.

2.1.2 Anwendung in der Bildverarbeitung

In diesem Unterabschnitt setzen wir die oben konstruierten Wavelets zur Entstörung und Approximation von digitalen Bildern ein. Nach einer geeigneten Projektion in den Starraum V_{-1} wenden wir einen Zerlegungsalgorithmus und ein anschließendes Shrinkage-Verfahren an.

Projektion in V_{-1} Sei $\mathbf{a} = (a_{m_1, m_2})_{(m_1, m_2) \in I}$ ein quadratisches Bild dyadischer Größe, d. h. $I := \{0, \dots, N-1\}^2$ und $N = 2^J$ für festes $J \in \mathbb{N}$. Eine $L^2([0, N]^2)$ -Version des diskreten Bildes kann durch

$$f(x_1, x_2) = \sum_{m_1=0}^{N-1} \sum_{m_2=0}^{N-1} a_{m_1, m_2} \chi_{[0,1]^2}(x_1 - m_1, x_2 - m_2) = \sum_{k \in I} a_k \chi_{[0,1]^2}(x - k), \quad (2.7)$$

angegeben werden, wobei $\chi_{[0,1]^2}$ die charakteristische Funktion auf $[0, 1]^2$ bezeichnet. Um eine Multiskalen-Darstellung des Bildes f vom feinen zum groben Level zu berechnen, benötigen wir eine orthogonale Projektion von f in den Raum V_{-1} als Startpunkt für den Zerlegungsalgorithmus. Wir wählen als Starraum V_{-1} , um die unvermeidbare Redundanz minimal zu halten. Dazu benutzen wir die Vektornotation (2.5),

$$f_{-1}(x) = P_{V_{-1}} f(x) = \sum_{k \in I_1} c_k^T \frac{1}{2} \Phi \left(\frac{1}{2} x - k \right), \quad (2.8)$$

mit $c_k := \langle f, \frac{1}{2} \Phi \left(\frac{1}{2} \cdot -k \right) \rangle \in \mathbb{R}^8$ und $I_1 := \{0, \dots, \frac{N}{2} - 1\}^2$. Die Koeffizienten-Vektoren ergeben sich mit (2.7) aus

$$c_k = \langle f, \frac{1}{2} \Phi \left(\frac{1}{2} \cdot -k \right) \rangle = \sum_{k' \in I} \frac{1}{2} a_{k'} \langle \chi_{[0,1]^2}(\cdot - k'), \Phi \left(\frac{1}{2} \cdot -k \right) \rangle.$$

Eine einfache Rechnung ergibt

$$\begin{aligned} \langle \chi_{[0,1]^2}(\cdot - k'), \Phi\left(\frac{1}{2} \cdot -k\right) \rangle &= \int_{[0,1]^2} \Phi\left(\frac{y + k'}{2} - k\right) dy \\ &= \begin{cases} \sqrt{2}(1, 1, 0, 0, 0, 0, 0, 0)^T, & k' = 2k \\ \sqrt{2}(0, 0, 1, 1, 0, 0, 0, 0)^T, & k' = 2k + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \sqrt{2}(0, 0, 0, 0, 1, 1, 0, 0)^T, & k' = 2k + \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\ \sqrt{2}(0, 0, 0, 0, 0, 0, 1, 1)^T, & k' = 2k + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ 0, & \text{sonst} \end{cases} . \end{aligned}$$

Also bestehen die Koeffizienten-Vektoren c_k in (2.8) aus den Pixelwerten a_k des Bildes,

$$c_k^T = \frac{1}{\sqrt{2}} \left(a_{2k}, a_{2k}, a_{2k + \begin{pmatrix} 0 \\ 1 \end{pmatrix}}, a_{2k + \begin{pmatrix} 0 \\ 1 \end{pmatrix}}, a_{2k + \begin{pmatrix} 1 \\ 1 \end{pmatrix}}, a_{2k + \begin{pmatrix} 1 \\ 1 \end{pmatrix}}, a_{2k + \begin{pmatrix} 1 \\ 0 \end{pmatrix}}, a_{2k + \begin{pmatrix} 1 \\ 0 \end{pmatrix}} \right). \quad (2.9)$$

Ausgehend von der Projektion $f_{-1} \in V_{-1}$ zerlegen wir das Bild mittels eines Filterbank-Algorithmus in seine Waveletkoeffizienten. Nach einem geeigneten Shrinkage rekonstruieren wir das Bild wieder.

Bildentstörung Der Koeffizienten-Vektor in (2.9) enthält acht Einträge, die von vier Pixelwerten herrühren. Diese zweifache Redundanz lässt sich in der Bildentstörung ausnutzen. Wir betrachten wie gewohnt Gauß'sches Rauschen mit $\sigma = 15$. In Abbildung 2.3 wird die gerichtete Haarwavelet-Filterbank auf das synthetische Testbild der Größe 256×256 angewandt. Nach vollständiger Zerlegung ($J = 7$) in die Waveletkoeffizienten werden die Koeffizienten einem Shrinkage unterworfen. Das globale Hard-Thresholding-Verfahren mit dem Schwellwert $\lambda = \sigma \sqrt{\log(N^2)/2}$ liefert einen PSNR-Wert von 33.81 dB für das rekonstruierte Bild. Im Vergleich mit der klassischen Tensorprodukt-Haarwavelettransformation (Tensor-HWT) stellen wir eine verbesserte Darstellung der diagonalen Bildkanten fest.

Bildapproximation Im Gegensatz zur Bildentstörung ist die doppelte Informationsmenge, die durch die Projektion in den Startraum V_{-1} entstand, bei der Approximation von Bildern hinderlich. Dennoch geben wir in Abbildung 2.4 ein numerisches Beispiel zur Bildapproximation an, indem wir einen 256×256 -Ausschnitt des „Elaine“-Bildes mit nur 5% der Koeffizienten approximieren. Das bedeutet, dass der Shrinkage-Schwellwert λ so gewählt wurde, dass nur die 3.277 betragsmäßig größten Koeffizienten übrigbleiben. Dabei ist in dem approximierten Bild deutlich die Dreiecksstruktur der Träger zu erkennen. Die vergleichsweise schlechte Approximationsqualität bei natürlichen Bildern hat ihre Ursache einerseits in der bereits erwähnten Redundanz, andererseits aber auch in der Unstetigkeit der Haarwavelets.

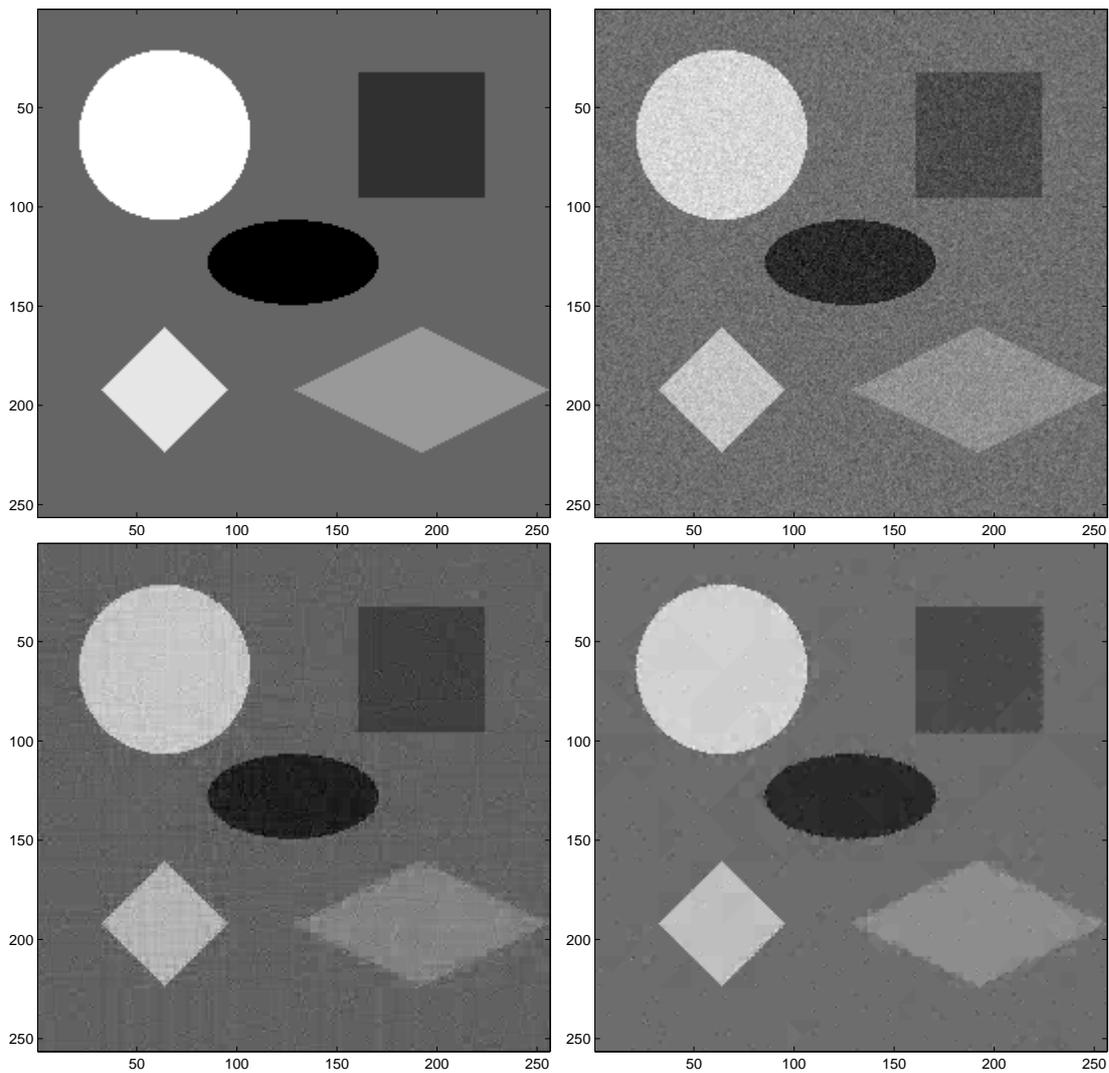


Abbildung 2.3: Entstörung eines synthetischen Bildes. (a) Originalbild, (b) Gestörtes Bild, PSNR 24.58 dB, (c) Tensor-HWT, PSNR 30.31 dB, (d) Gerichtete HWT, PSNR 33.81 dB.



Abbildung 2.4: Approximation des „Elaine“-Bildes mit 3.277 Koeffizienten. (a) Originalbild, (b) Tensor-HWT, PSNR 29.68 dB, (c) Tensor-D4, PSNR 30.81 dB, (d) Gerichtete HWT, PSNR 29.20 dB.

2.2 Waveletframes mit acht Richtungen

Im zweiten Teil des Kapitels modifizieren wir unseren Ansatz dahingehend, dass wir statt vier Hauptrichtungen nun acht Richtungen berücksichtigen. Dazu nehmen wir eine andere Dreieckszerlegung der Ebene vor. Die Erhöhung der Anzahl der Richtungen muss mit dem Verlust der Basiseigenschaft bezahlt werden. Wir werden sehen, dass das entstehende \mathcal{AB} -Waveletsystem lediglich ein Parseval-Frame mit vierfacher Redundanz ist. Neben den guten Richtungseigenschaften sind die kleinen, kompakten Träger im Zeitbereich und die niedrige Anzahl von Mutterwavelets sowie insbesondere die zugrunde liegende MRA-Struktur weitere Vorteile dieses Waveletsystems.

Die Ergebnisse dieses Unterkapitels haben wir in [KP09] vorgestellt.

2.2.1 Konstruktion der Approximationsräume

Die Konstruktion ist sehr ähnlich zu der im Abschnitt 2.1.1, jedoch starten wir hier mit dem Quadrat $\Omega := [-1, 1]^2$ und zerlegen es in 16 Dreiecke gleichen Flächeninhalts, so wie in Abbildung 2.5(a) dargestellt. Sei nun wieder $\phi_0(x) = \chi_{U_0}(x)$ die charakteristische Funktion auf dem Dreieck

$$U_0 = \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} := \left\{ x \in \mathbb{R}^2 : 0 \leq x_2 \leq 1, 0 \leq x_1 \leq \frac{x_2}{2} \right\}.$$

Es gilt also $\phi_0(x) = \phi_0(x_1, x_2) = \chi_{[0,1]} \left(\frac{2x_1}{x_2} \right) \chi_{[0,1]}(x_2)$. Analog dazu ist die zweite Mutter-Skalierungsfunktion ϕ_1 definiert durch

$$\phi_1(x) = \phi_1(x_1, x_2) := \chi_{U_1}(x_1, x_2) = \chi_{[1,2]} \left(\frac{2x_1}{x_2} \right) \chi_{[0,1]}(x_2),$$

mit $U_1 = \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1 \end{pmatrix} \right\}$. Beachte, dass mit der Schermatrix $S = \begin{pmatrix} 1 & 1/2 \\ 0 & 1 \end{pmatrix}$ die Beziehung $\phi_1(x) = \phi_0(S^{-1}x)$ gilt. Die übrigen Dreiecke U_2, \dots, U_{15} sind gespiegelte bzw. gedrehte Versionen von U_0 oder U_1 , je nach dem, ob der Index gerade oder ungerade ist. Mit der Gruppe \mathcal{B} der Isometrien in (2.1) können wir für $i = 0, \dots, 7$ die Dreiecke

$$U_{2i} = \{B_i^{-1}x : x \in U_0\} = B_i^{-1}U_0, \quad U_{2i+1} = \{B_i^{-1}x : x \in U_1\} = B_i^{-1}U_1$$

beschreiben, die die Träger der \mathcal{AB} -Skalierungsfunktionen $\phi_i, i = 0, \dots, 7$, liefern:

$$\phi_{2i}(x) := \phi_0(B_i x) = \chi_{U_0}(B_i x) = \chi_{B_i^{-1}U_0}(x) = \chi_{U_{2i}}(x),$$

$$\phi_{2i+1}(x) := \phi_1(B_i x) = \chi_{U_1}(B_i x) = \chi_{B_i^{-1}U_1}(x) = \chi_{U_{2i+1}}(x).$$

Mithilfe der Spiegelungen/Rotationen $B_i \in \mathcal{B}$, der Translationen $k \in \mathbb{Z}^2$ und der Skalierungen $A^j, j \in \mathbb{Z}$, mit $A = 2I_2$ haben die nicht-separablen, gerichteten \mathcal{AB} -Skalierungsfunktionen schließlich die allgemeine Gestalt

$$\begin{aligned} \phi_{2i,j,k}(x) &:= 2^j \phi_0(B_i(2^j x - k)), \\ \phi_{2i+1,j,k}(x) &:= 2^j \phi_1(B_i(2^j x - k)). \end{aligned}$$

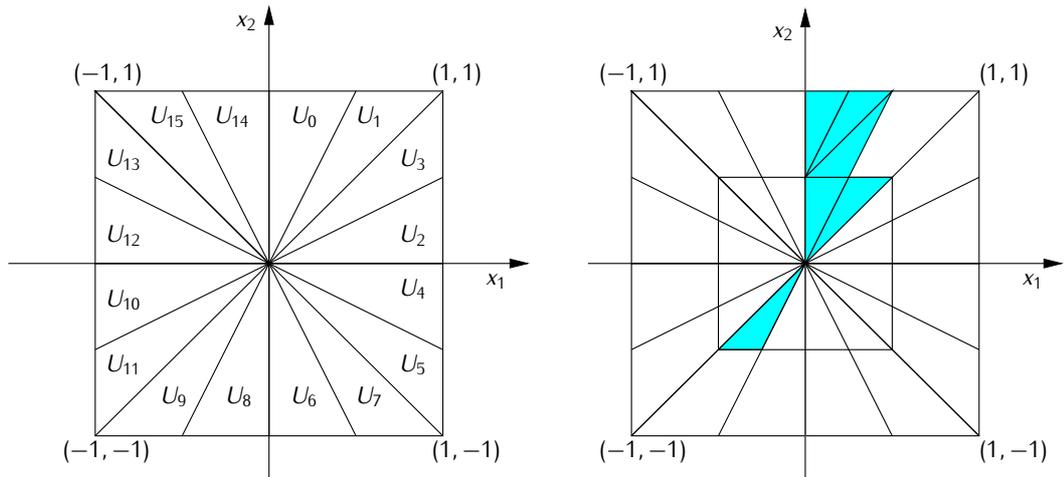


Abbildung 2.5: Dreieckszerlegung mit acht Richtungen. (a) Level V_0 , (b) Verfeinerung von U_0 .

Bei unseren nachfolgenden Untersuchungen werden wir bei Bedarf eine Vektor-Schreibweise dieser 16 \mathcal{AB} -Skalierungsfunktionen $\phi_i, i = 0, \dots, 15$, verwenden. Dazu verschieben wir die Funktionen ϕ_4, \dots, ϕ_{15} so, dass ihre Träger in $[0, 1]^2$ enthalten sind. D. h. wir betrachten den Funktionen-Vektor Φ der Länge 16,

$$\Phi := (\phi_0, \dots, \phi_3, \phi_4(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}), \dots, \phi_7(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}), \\ \phi_8(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \dots, \phi_{11}(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \phi_{12}(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}), \dots, \phi_{15}(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}))^T. \quad (2.10)$$

Nun sind wir in der Lage, die wesentliche \mathcal{AB} -MRA-Eigenschaft der Approximationsräume zu formulieren.

Satz 2.3. Die Folge $\{V_j\}_{j \in \mathbb{Z}}$ von Approximationsräumen

$$V_j := \text{clos}_{L^2(\mathbb{R}^2)} \text{span}\{\phi_{2i,j,k}, \phi_{2i+1,j,k} : i = 0, \dots, 7; k \in \mathbb{Z}^2\} \quad (2.11)$$

ist eine \mathcal{AB} -MRA des $L^2(\mathbb{R}^2)$, d. h. es gelten die folgenden Eigenschaften:

$$(M1) \quad V_j \subset V_{j+1} \quad \text{für alle } j \in \mathbb{Z}.$$

$$(M2) \quad f \in V_j \Leftrightarrow f(2 \cdot) \in V_{j+1} \quad \forall j \in \mathbb{Z}.$$

$$(M3) \quad \text{clos}_{L^2(\mathbb{R}^2)} \bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}^2).$$

$$(M4) \quad \bigcap_{j \in \mathbb{Z}} V_j = \{0\}.$$

$$(M5) \quad \{\phi_{2i}(\cdot - k), \phi_{2i+1}(\cdot - k) : i = 0, \dots, 7; k \in \mathbb{Z}^2\} \text{ ist ein Frame von } V_0.$$

Beweis. Zu (M1). Wegen der verfeinerbaren Träger (siehe Abbildung 2.5(b)) sind auch die \mathcal{AB} -Skalierungsfunktionen verfeinerbar. Es gilt

$$\begin{aligned}\phi_0 &= \phi_0(2\cdot) + \phi_0(2\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_1(2\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_9(2\cdot - \begin{pmatrix} 1 \\ 2 \end{pmatrix}) \\ &= \frac{1}{2} \left(\phi_{0,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \phi_{0,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{1,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{9,1,\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right),\end{aligned}\quad (2.12)$$

$$\begin{aligned}\phi_1 &= \phi_1(2\cdot) + \phi_1(2\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) + \phi_0(2\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) + \phi_8(2\cdot - \begin{pmatrix} 1 \\ 2 \end{pmatrix}) \\ &= \frac{1}{2} \left(\phi_{1,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \phi_{1,1,\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{0,1,\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{8,1,\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right).\end{aligned}\quad (2.13)$$

Die Verfeinerungsgleichungen für die anderen \mathcal{AB} -Skalierungsfunktionen folgen dann einfach

$$\begin{aligned}\phi_{2i} &= \phi_0(B_i \cdot) \\ &= \phi_0(2B_i \cdot) + \phi_0(2B_i \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_1(2B_i \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_1(B_4(2B_i \cdot - \begin{pmatrix} 1 \\ 2 \end{pmatrix})) \\ &= \frac{1}{2} \left(\phi_{2i,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \phi_{2i,1,B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{2i+1,1,B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{(2i+9) \bmod 16,1,B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right)\end{aligned}\quad (2.14)$$

ebenso wie

$$\phi_{2i+1} = \frac{1}{2} \left(\phi_{2i+1,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \phi_{2i+1,1,B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{2i,1,B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{(2i+8) \bmod 16,1,B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right).\quad (2.15)$$

Also gilt $V_0 \subset V_1$. Für beliebiges $j \in \mathbb{Z}$ und $k \in \mathbb{Z}^2$ gelten die allgemeinen Verfeinerungsgleichungen

$$\begin{aligned}\phi_{2i,j,k} &= \phi_0(B_i(2^j \cdot - k)) \\ &= \frac{1}{2} \left(\phi_{2i,j+1,2k} + \phi_{2i,j+1,2k+B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \phi_{2i+1,j+1,2k+B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} \right. \\ &\quad \left. + \phi_{(2i+9) \bmod 16,j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right)\end{aligned}$$

$$\begin{aligned}\phi_{2i+1,j,k} &= \phi_1(B_i(2^j \cdot - k)) \\ &= \frac{1}{2} \left(\phi_{2i+1,j+1,2k} + \phi_{2i+1,j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{2i,j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} \right. \\ &\quad \left. + \phi_{(2i+8) \bmod 16,j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right).\end{aligned}$$

Also gilt $V_j \subset V_{j+1}$ für alle $j \in \mathbb{Z}$.

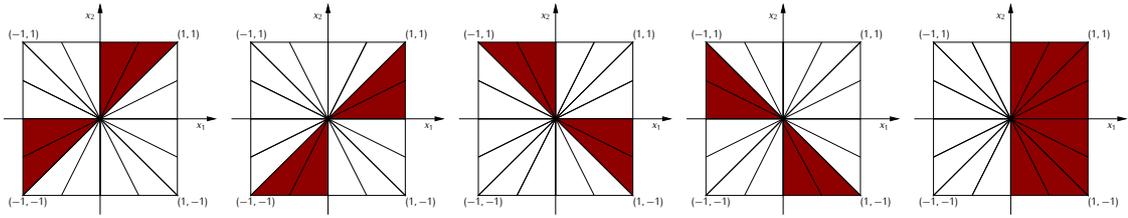
Zu (M2). Folgt aus der Definition der Approximationsräume in (2.11).

Zu (M3). Weil die Approximationsräume V_j gemäß der Konstruktion (2.11) die Unterräume der Haar-Skalierungsfunktionen enthalten, d. h. $V_j^H \subset V_j$ mit

$$V_j^H := \text{clos}_{L^2(\mathbb{R}^2)} \text{span} \{2^j \chi_{[0,1)^2}(2^j \cdot - k) : k \in \mathbb{Z}^2\},$$

ergibt sich sofort $\text{clos}_{L^2(\mathbb{R}^2)} \bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}^2)$.

Zu (M4). Diese Aussage folgt für eine stationäre Folge $\{V_j\}_{j \in \mathbb{Z}}$ nach [dBDV93, Korollar 4.14].

Abbildung 2.6: Redundanzen der \mathcal{AB} -Skalierungsfunktionen.

Zu (M5). Das Funktionensystem $\{\phi_{2i}(\cdot - k), \phi_{2i+1}(\cdot - k) : i = 0, \dots, 7; k \in \mathbb{Z}^2\}$ bildet offensichtlich keine Basis von V_0 , denn es herrschen folgende Abhängigkeiten:

$$\begin{aligned}
 \phi_0 + \phi_1 &= \phi_{10}(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) + \phi_{11}(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \\
 \phi_2 + \phi_3 &= \phi_8(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) + \phi_9(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \\
 \phi_4 + \phi_5 &= \phi_{14}(\cdot - \begin{pmatrix} 1 \\ -1 \end{pmatrix}) + \phi_{15}(\cdot - \begin{pmatrix} 1 \\ -1 \end{pmatrix}), \\
 \phi_6 + \phi_7 &= \phi_{12}(\cdot - \begin{pmatrix} 1 \\ -1 \end{pmatrix}) + \phi_{13}(\cdot - \begin{pmatrix} 1 \\ -1 \end{pmatrix}), \\
 \phi_0 + \phi_1 + \phi_2 + \phi_3 &= \phi_4(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_5(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_6(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_7(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}).
 \end{aligned} \tag{2.16}$$

Diese Gleichungen werden schnell klar, wenn man anhand von Abbildung 2.6 die entsprechenden Dreiecksträger der \mathcal{AB} -Skalierungsfunktionen betrachtet. In der Tat wird der Raum V_0 bereits durch die Translationen der 11 Funktionen $\{\phi_{2i} : i = 0, \dots, 7\} \cup \{\phi_1, \phi_3, \phi_5\}$ aufgespannt. Die *Gram'sche Matrix* $G := \langle \Phi, \Phi \rangle \in \mathbb{R}^{16 \times 16}$ mit Φ aus (2.10) ist eine Block-Toeplitz-Matrix

$$G = \frac{1}{4} \begin{pmatrix} I_4 & G_1 & G_2 & G_1^T \\ G_1^T & I_4 & G_1 & G_2 \\ G_2 & G_1^T & I_4 & G_1 \\ G_1 & G_2 & G_1^T & I_4 \end{pmatrix} \tag{2.17}$$

mit der Einheitsmatrix I_4 der Größe 4×4 und mit

$$G_1 = \begin{pmatrix} 1/5 & 2/15 & 1/2 & 1/6 \\ 7/15 & 1/5 & 1/6 & 1/6 \\ 0 & 1/3 & 1/5 & 7/15 \\ 1/3 & 1/3 & 2/15 & 1/5 \end{pmatrix}, \quad G_2 = \begin{pmatrix} 0 & 0 & 2/3 & 1/3 \\ 0 & 0 & 1/3 & 2/3 \\ 2/3 & 1/3 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 \end{pmatrix}.$$

Es ist $\text{Rang}(G) = 11$. Der größte und der kleinste von Null verschiedene Eigenwert von G liefern uns die Frame-Konstanten des Frames $\{\phi_i(\cdot - k) : i = 0, \dots, 15; k \in \mathbb{Z}^2\}$, vgl. [BHF98]. Genauer gesagt, die Frame-Ungleichung

$$A \|f\|_{L^2(\mathbb{R}^2)}^2 \leq \sum_{i=0}^{15} \sum_{k \in \mathbb{Z}^2} |\langle f, \phi_i(\cdot - k) \rangle|^2 \leq B \|f\|_{L^2(\mathbb{R}^2)}^2$$

ist für alle $f \in V_0$ mit $A \approx 0.0745$ und $B = 1$ erfüllt. \square

Der Rest des Unterkapitels ist der Berechnung eines Dualframes $\{\tilde{\phi}_i(\cdot - k) : i = 0, \dots, 15; k \in \mathbb{Z}^2\}$ von V_0 gewidmet, damit wir jedes $f \in V_0$ darstellen können als

$$f = \sum_{i=0}^{15} \sum_{k \in \mathbb{Z}^2} \langle f, \tilde{\phi}_i(\cdot - k) \rangle \phi_i(\cdot - k) = \sum_{i=0}^{15} \sum_{k \in \mathbb{Z}^2} \langle f, \phi_i(\cdot - k) \rangle \tilde{\phi}_i(\cdot - k). \quad (2.18)$$

Der zu Φ kanonische duale Frame $\tilde{\Phi} := (\tilde{\phi}_i)_{i=0}^{15}$ kann mithilfe des folgenden Satzes auf einfache Weise berechnet werden.

Satz 2.4. Sei $F = (f_i)_{i=0}^{N-1}$, $N \in \mathbb{N}$, ein Vektor mit Funktionen $f_i \in L^2(\Omega)$, $\Omega \subset \mathbb{R}^d$, der einen Frame für den endlichen Unterraum $V := \text{clos}_{L^2(\Omega)} \text{span}\{f_0, \dots, f_{N-1}\}$ von $L^2(\Omega)$ erzeugt. Einen zu F dualen Frame $\tilde{F} = (\tilde{f}_j)_{j=0}^{N-1}$ erhält man dann mittels

$$\tilde{F} = G^\dagger F, \quad (2.19)$$

wobei G^\dagger die eindeutig definierte Moore-Penrose-Inverse (auch verallgemeinerte Inverse oder Pseudoinverse genannt) der Gram'schen Matrix $G = \langle F, F \rangle = (\langle f_i, f_j \rangle)_{i,j=0}^{N-1}$ ist.

Beweis. Siehe Anhang A. □

Durch folgende Überlegungen kann man sich klar machen, dass die Wahl $\tilde{\Phi} = G^\dagger \Phi$ tatsächlich die Darstellung (2.18) erfüllt. Weil G symmetrisch und positiv semidefinit ist, existiert eine orthogonale Matrix $P \in \mathbb{R}^{16 \times 16}$ und eine Diagonalmatrix $D = \text{diag}(\lambda_1, \dots, \lambda_{11}, 0, \dots, 0) \in \mathbb{R}^{16 \times 16}$ derart, dass $G = P^T D P$ gilt. Das heißt, die Pseudoinverse G^\dagger hat die Gestalt $G^\dagger = P^T D^\dagger P$, wobei $D^\dagger = \text{diag}(1/\lambda_1, \dots, 1/\lambda_{11}, 0, \dots, 0)$. Nun wird aber auch V_0 durch $P\Phi$ erzeugt, und es gilt

$$\langle P\Phi, P\Phi \rangle = P \langle \Phi, \Phi \rangle P^T = P G P^T = D,$$

d. h., die letzten fünf Funktionen in dem 16er-Vektor $P\Phi$ sind Null-Funktionen.

Für beliebige Funktionen $g = c^T P\Phi \in V_0$ (auf $[0, 1]^2$ beschränkt) folgt jetzt

$$\sum_{i=0}^{15} \langle g, \tilde{\phi}_i \rangle \phi_i = \langle c^T P\Phi, G^\dagger \Phi \rangle \Phi = c^T P \langle \Phi, \Phi \rangle G^\dagger \Phi = c^T P G G^\dagger \Phi = c^T D D^\dagger P\Phi = c^T P\Phi = g,$$

was den ersten Teil von (2.18) erfüllt. Die zweite Gleichung folgt analog.

Die Block-Toeplitz-Struktur von G überträgt sich auf ihre Pseudo-Inverse G^\dagger ,

$$G^\dagger = \begin{pmatrix} \hat{G}_0 & \hat{G}_1 & \hat{G}_2 & \hat{G}_1^T \\ \hat{G}_1^T & \hat{G}_0 & \hat{G}_1 & \hat{G}_2 \\ \hat{G}_2 & \hat{G}_1^T & \hat{G}_0 & \hat{G}_1 \\ \hat{G}_1 & \hat{G}_2 & \hat{G}_1^T & \hat{G}_0 \end{pmatrix}$$

mit

$$\hat{G}_0 = \begin{pmatrix} 3.75 & -2.58 & -0.08 & -0.82 \\ -2.58 & 3.28 & -0.82 & 0.37 \\ -0.08 & -0.82 & 3.75 & -2.58 \\ -0.82 & 0.37 & -2.58 & 3.28 \end{pmatrix}, \quad \hat{G}_1 = \begin{pmatrix} 0.36 & 0.68 & -1.74 & 0.94 \\ -1.16 & 0.36 & 0.94 & 0.10 \\ 1.48 & -0.44 & 0.36 & -1.16 \\ -0.44 & -0.35 & 0.68 & 0.36 \end{pmatrix},$$

und

$$\hat{G}_2 = \begin{pmatrix} -1.21 & 0.29 & -0.66 & 1.83 \\ 0.29 & -0.74 & 1.83 & -1.12 \\ -0.66 & 1.83 & -1.21 & 0.29 \\ 1.83 & -1.12 & 0.29 & -0.74 \end{pmatrix}.$$

Wir haben hier G^T mit der üblichen Maple-Prozedur berechnet und auf zwei Nachkommastellen gerundet.

2.2.2 Konstruktion der Waveleträume

Nun betrachten wir die Waveleträume W_j , die die Bedingung $V_j + W_j = V_{j+1}$ für alle $j \in \mathbb{Z}$ erfüllen. Über die Verfeinerungsgleichungen der \mathcal{AB} -Mutter-Skalierungsfunktionen ϕ_0 und ϕ_1 in (2.12) und (2.13) erhalten wir die \mathcal{AB} -Mutterwavelets mittels

$$\psi_0^1 := \frac{1}{2} \left(\phi_{0,1,\binom{0}{0}} + \phi_{0,1,\binom{0}{1}} - \phi_{1,1,\binom{0}{1}} - \phi_{9,1,\binom{1}{2}} \right), \quad (2.20)$$

$$\psi_0^2 := \frac{1}{2} \left(\phi_{0,1,\binom{0}{0}} - \phi_{0,1,\binom{0}{1}} - \phi_{1,1,\binom{0}{1}} + \phi_{9,1,\binom{1}{2}} \right),$$

$$\psi_0^3 := \frac{1}{2} \left(\phi_{0,1,\binom{0}{0}} - \phi_{0,1,\binom{0}{1}} + \phi_{1,1,\binom{0}{1}} - \phi_{9,1,\binom{1}{2}} \right), \quad (2.21)$$

siehe Abbildung 2.7, und für ϕ_1

$$\psi_1^1 := \frac{1}{2} \left(\phi_{1,1,\binom{0}{0}} - \phi_{1,1,\binom{1}{1}} + \phi_{0,1,\binom{1}{1}} - \phi_{8,1,\binom{1}{2}} \right),$$

$$\psi_1^2 := \frac{1}{2} \left(\phi_{1,1,\binom{0}{0}} - \phi_{1,1,\binom{1}{1}} - \phi_{0,1,\binom{1}{1}} + \phi_{8,1,\binom{1}{2}} \right),$$

$$\psi_1^3 := \frac{1}{2} \left(\phi_{1,1,\binom{0}{0}} + \phi_{1,1,\binom{1}{1}} - \phi_{0,1,\binom{1}{1}} - \phi_{8,1,\binom{1}{2}} \right).$$

Die \mathcal{AB} -Wavelets ψ_i^1 , ψ_i^2 und ψ_i^3 haben nach Konstruktion denselben Träger wie die dazugehörige Skalierungsfunktion ϕ_i , $i = 0, 1$. Alle weiteren \mathcal{AB} -Wavelets erhält man durch Rotation bzw. Spiegelung dieser sechs Wavelets, und zwar

$$\psi_{2i}^l := \psi_0^l(B_i \cdot) \quad \text{und} \quad \psi_{2i+1}^l := \psi_1^l(B_i \cdot), \quad \text{für } i = 0, \dots, 7, l = 1, 2, 3.$$

Nun sind wir in der Lage, die gewünschten Waveleträume festzulegen,

$$W_j := \text{clos}_{L^2(\mathbb{R}^2)} \text{span} \{ \psi_{i,j,k}^l : i = 0, \dots, 15; l = 1, 2, 3; k \in \mathbb{Z}^2 \}.$$

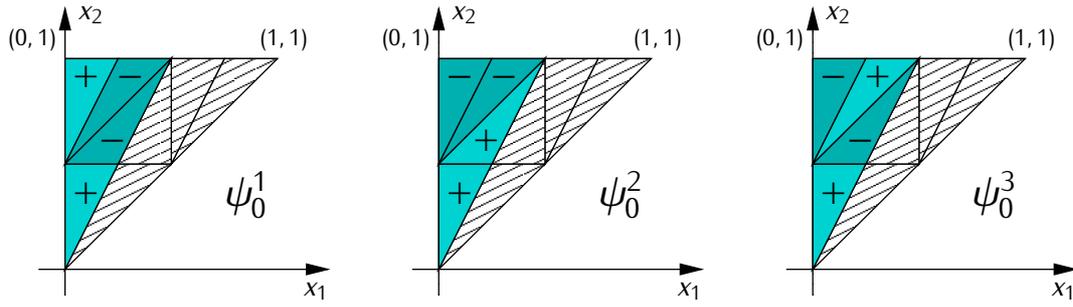


Abbildung 2.7: Konstruktion der gerichteten Mutterwavelets ψ_0^1 , ψ_0^2 und ψ_0^3 .

Hierbei sind die durch $k \in \mathbb{Z}^2$ verschobenen und mit $2^j, j \in \mathbb{Z}$, skalierten Versionen der \mathcal{AB} -Wavelets in der uns schon bekannten Weise definiert,

$$\begin{aligned}\psi_{2^i,j,k}^1 &:= 2^j \psi_0^1(B_i(2^j \cdot -k)), \\ \psi_{2^{i+1},j,k}^1 &:= 2^j \psi_1^1(B_i(2^j \cdot -k)),\end{aligned}\tag{2.22}$$

mit $B_i \in \mathcal{B}$. Die obigen Verfeinerungsgleichungen für ψ_i^1 , ψ_i^2 und ψ_i^3 , $i = 0, 1$, in W_0 lassen sich unmittelbar verallgemeinern zu

$$\begin{aligned}\psi_{2^i,j,k}^1 &= \frac{1}{2} \left(\phi_{2^i,j+1,2k} + \phi_{2^i,j+1,2k+B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} - \phi_{2^{i+1},j+1,2k+B_i^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}} \right. \\ &\quad \left. - \phi_{(2i+9) \bmod 16, j+1, 2k+B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right), \\ \psi_{2^{i+1},j,k}^1 &= \frac{1}{2} \left(\phi_{2^{i+1},j+1,2k} - \phi_{2^{i+1},j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \phi_{2^i,j+1,2k+B_i^{-1}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} \right. \\ &\quad \left. - \phi_{(2i+8) \bmod 16, j+1, 2k+B_i^{-1}\begin{pmatrix} 1 \\ 2 \end{pmatrix}} \right),\end{aligned}$$

für $k \in \mathbb{Z}^2, j \in \mathbb{Z}, i = 0, \dots, 7$. Analoge Gleichungen gelten für $\psi_{i,j,k}^2$ und $\psi_{i,j,k}^3$, und wir haben $W_j \subset V_{j+1}$ für alle $j \in \mathbb{Z}$. Es ist klar, dass die \mathcal{AB} -Wavelets $\psi_{i,j,k}^l$ denselben Träger wie die dazugehörige \mathcal{AB} -Skalierungsfunktion $\phi_{i,j,k}$ besitzen.

Schließlich machen wir uns noch klar, dass auch die Beziehung $V_{j+1} \subset V_j + W_j$, $j \in \mathbb{Z}$, gilt, indem wir die Rekonstruktionsformeln betrachten. Dabei beschränken wir uns wieder auf die \mathcal{AB} -Mutter-Skalierungsfunktionen. Es gilt

$$\begin{aligned}\phi_{0,j+1,2k} &= \frac{1}{2} \left(\phi_{0,j,k} + \psi_{0,j,k}^1 + \psi_{0,j,k}^2 + \psi_{0,j,k}^3 \right), \\ \phi_{0,j+1,2k+\begin{pmatrix} 0 \\ 1 \end{pmatrix}} &= \frac{1}{2} \left(\phi_{0,j,k} + \psi_{0,j,k}^1 - \psi_{0,j,k}^2 - \psi_{0,j,k}^3 \right), \\ \phi_{0,j+1,2k+\begin{pmatrix} 1 \\ 0 \end{pmatrix}} &= \frac{1}{2} \left(\phi_{9,j,k+\begin{pmatrix} 1 \\ 1 \end{pmatrix}} - \psi_{9,j,k+\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^1 + \psi_{9,j,k+\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^2 - \psi_{9,j,k+\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^3 \right), \\ \phi_{0,j+1,2k+\begin{pmatrix} 1 \\ 1 \end{pmatrix}} &= \frac{1}{2} \left(\phi_{1,j,k} + \psi_{1,j,k}^1 - \psi_{1,j,k}^2 - \psi_{1,j,k}^3 \right),\end{aligned}$$

siehe dazu auch wieder Abbildung 2.7, und ebenso gilt

$$\begin{aligned}\phi_{1,j+1,2k} &= \frac{1}{2} \left(\phi_{1,j,k} + \psi_{1,j,k}^1 + \psi_{1,j,k}^2 + \psi_{1,j,k}^3 \right), \\ \phi_{1,j+1,2k+\binom{1}{0}} &= \frac{1}{2} \left(\phi_{8,j,k+\binom{1}{1}} - \psi_{8,j,k+\binom{1}{1}}^1 + \psi_{8,j,k+\binom{1}{1}}^2 - \psi_{8,j,k+\binom{1}{1}}^3 \right), \\ \phi_{1,j+1,2k+\binom{0}{1}} &= \frac{1}{2} \left(\phi_{0,j,k} - \psi_{0,j,k}^1 - \psi_{0,j,k}^2 + \psi_{0,j,k}^3 \right), \\ \phi_{1,j+1,2k+\binom{1}{1}} &= \frac{1}{2} \left(\phi_{1,j,k} - \psi_{1,j,k}^1 - \psi_{1,j,k}^2 + \psi_{1,j,k}^3 \right).\end{aligned}$$

Insgesamt erhalten wir die gewünschte Zerlegung der Approximationsräume für $j \in \mathbb{Z}$

$$V_j + W_j = V_{j+1},$$

wobei hier weder eine direkte noch eine orthogonale Summe vorliegt. Zuletzt zeigen wir die Hauptaussage dieses Abschnitts, nämlich die Tight-Frame-Eigenschaft des Waveletsystems

$$\Psi_D := \{\psi_{i,j,k}^l : i = 0, \dots, 15; l = 1, 2, 3; j \in \mathbb{Z}; k \in \mathbb{Z}^2\}.$$

Dazu benötigen wir das folgende, wichtige Lemma.

Lemma 2.5. *Der Approximationsraum V_j kann für $j \in \mathbb{Z}$ in die vier Unterräume*

$$V_j^v := \text{clos}_{L^2(\mathbb{R}^2)} \text{span} \{ \phi_{2v,j,k}, \phi_{2v+1,j,k}, \phi_{2v+8,j,k}, \phi_{2v+9,j,k} : k \in \mathbb{Z}^2 \} \quad (2.23)$$

für $v = 0, 1, 2, 3$, zerlegt werden:

$$V_j = V_j^0 + V_j^1 + V_j^2 + V_j^3.$$

Außerdem bildet jede Folge $\{V_j^v\}_{j \in \mathbb{Z}}$ eine MRA des $L^2(\mathbb{R}^2)$, d.h. es ist insbesondere $\{ \phi_{2v,j,k}, \phi_{2v+1,j,k}, \phi_{2v+8,j,k}, \phi_{2v+9,j,k} : k \in \mathbb{Z}^2 \}$ eine Orthogonalbasis von V_j^v mit $\|\phi_{i,j,k}\|_2^2 = \frac{1}{4}$ für alle $i = 0, \dots, 15; j \in \mathbb{Z}; k \in \mathbb{Z}^2$.

Beweis. Wir müssen nur die MRA-Eigenschaften der Folgen $\{V_j^v\}_{j \in \mathbb{Z}}$ beweisen, alles andere folgt sofort daraus. Die Verfeinerungsgleichungen (2.14) und (2.15) der \mathcal{AB} -Skalierungsfunktionen zeigen, dass zur Verfeinerung von $\phi \in V_0^v$ nur Funktionen aus dem Unterraum V_1^v benötigt werden. Damit gilt also (M1). Aus der Definition der Approximationsräume in (2.23) folgt sofort (M2). Die beiden MRA-Bedingungen (M3) und (M4) folgen mit demselben Argument wie im Beweis von Satz 2.3. Anhand der Abbildung 2.8 erkennen wir leicht, dass die vier Basisfunktionen mit ihren \mathbb{Z}^2 -Translationen wegen der disjunkten Dreiecksträger die Orthogonalitätsbedingung

$$\langle \phi_{i,j,k}, \phi_{i',j',k'} \rangle = \delta_{i,i'} \delta_{k,k'}$$

erfüllen, mit $i, i' \in \{2v, 2v+1, 2v+8, 2v+9\}$ für $v = 0, 1, 2, 3$, und $k \in \mathbb{Z}^2$, also gilt auch (M5). \square

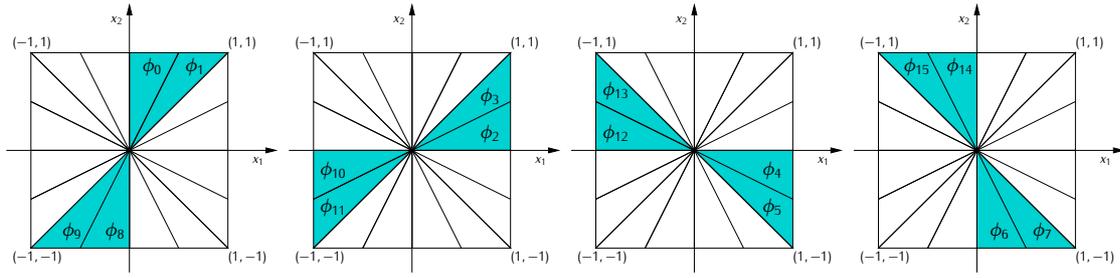


Abbildung 2.8: Die Basisfunktionen der vier Unterräume V_0^0, V_0^1, V_0^2 und V_0^3 .

Durch diese Aufteilung in Unterräume sortieren wir die \mathcal{AB} -Skalierungsfunktionen gewissermaßen nach Hauptrichtungen. Betrachtet man Abbildung 2.8, so sieht man, dass z. B. der Raum V_j^0 die \mathcal{AB} -Skalierungsfunktionen enthält, die von links unten nach rechts oben orientiert sind, während V_j^3 die von oben links nach unten rechts gerichteten Funktionen enthält.

Satz 2.6. *Das System Ψ_D der gerichteten \mathcal{AB} -Haarwavelets auf Dreiecken bildet einen Parseval-Frame des $L^2(\mathbb{R}^2)$, d. h. es gilt für alle $f \in L^2(\mathbb{R}^2)$*

$$\|f\|_2^2 = \sum_{\psi \in \Psi_D} |\langle f, \psi \rangle|^2.$$

Beweis. Entsprechend der vier Folgen $\{V_j^\nu\}_{j \in \mathbb{Z}}$, $\nu = 0, 1, 2, 3$, die gemäß Lemma 2.5 eine MRA des $L^2(\mathbb{R}^2)$ bilden, können wir die Differenzräume W_j^0, W_j^1, W_j^2 und W_j^3 betrachten, die Unterräume des W_j sind, d. h.,

$$W_j^\nu := \text{clos}_{L^2(\mathbb{R}^2)} \text{span} \{ \psi_{2^\nu j, k}^l, \psi_{2^\nu j+1, k}^l, \psi_{2^\nu j+8, k}^l, \psi_{2^\nu j+9, k}^l : l = 1, 2, 3; k \in \mathbb{Z}^2 \},$$

für $\nu = 0, 1, 2, 3$. Aus der Konstruktion der \mathcal{AB} -Wavelets über die Verfeinerungsgleichungen folgt sofort $W_j^\nu \perp V_j^\nu$, und das Erzeugendensystem ist eine Orthogonalbasis von W_j^ν für jedes $\nu = 0, 1, 2, 3$. Also ist jedes der vier Funktionensysteme

$$\Psi^\nu := \{ \psi_{2^\nu j, k}^l, \psi_{2^\nu j+1, k}^l, \psi_{2^\nu j+8, k}^l, \psi_{2^\nu j+9, k}^l : l = 1, 2, 3; j \in \mathbb{Z}; k \in \mathbb{Z}^2 \}, \quad \nu = 0, 1, 2, 3,$$

eine Orthogonalbasis des $L^2(\mathbb{R}^2)$, und die Parseval'sche Gleichung hat mit $\|\psi_{i, j, k}^l\|_2^2 = \frac{1}{4}$ die Gestalt

$$\|f\|_2^2 = \sum_{\psi \in \Psi^\nu} \frac{|\langle f, \psi \rangle|^2}{\langle \psi, \psi \rangle} = 4 \sum_{\psi \in \Psi^\nu} |\langle f, \psi \rangle|^2$$

für alle $f \in L^2(\mathbb{R}^2)$, $\nu = 0, 1, 2, 3$. Damit folgt, dass das gesamte Waveletsystem $\Psi_D = \bigcup_{\nu=0}^3 \Psi^\nu$ einen Parseval-Frame des $L^2(\mathbb{R}^2)$ bildet, mit

$$\|f\|_2^2 = \sum_{\psi \in \Psi_D} |\langle f, \psi \rangle|^2.$$

□

Bemerkung 2.7. 1. Wenn wir die \mathcal{AB} -Skalierungsfunktionen und damit auch die \mathcal{AB} -Wavelets normieren würden, so dass $\|\phi_{i,j,k}\|_2^2 = \|\psi_{i,j,k}^l\|_2^2 = 1$ für alle $i = 0, \dots, 15, j \in \mathbb{Z}, k \in \mathbb{Z}^2$ gelten würde, so würden wir statt eines Parseval-Frames einen Tight-Frame mit den Frame-Schranken $A = B = 4$ erhalten.

2. Wir betonen noch einmal die Tatsache, dass der gerichtete Waveletframe als Vereinigung von vier Orthogonalbasen angesehen werden kann und damit ein redundantes Dictionary ist. Diesen Umstand machen wir uns bei der Anwendung in der Bildverarbeitung zunutze, um eine effiziente Implementierung zu gewährleisten.

2.2.3 Gerichtete Haarwavelet-Filterbank

Wir betrachten wieder ein quadratisches Bild $\mathbf{a} = (a_{m_1, m_2})_{(m_1, m_2) \in I}$ dyadischer Größe, dessen Indexmenge durch $I := \{0, \dots, 2N - 1\}^2$ mit $N = 2^J$ für festes $J \in \mathbb{N}$ gegeben ist. Die entsprechende Bildfunktion $f \in L^2([0, 2N]^2)$ lautet

$$f(x_1, x_2) = \sum_{m_1=0}^{2N-1} \sum_{m_2=0}^{2N-1} a_{m_1, m_2} \chi_{[0,1]^2}(2x_1 - m_1, 2x_2 - m_2) = \sum_{k \in I} a_k \chi_{[0,1]^2}(2x - k). \quad (2.24)$$

Wir möchten unseren redundanten Haarwavelet-Frame benutzen, um eine effiziente Zerlegung von f zu erreichen. Bevor wir die einzelnen Schritte ausführlich beschreiben, skizzieren wir kurz unser weiteres Vorgehen in den nächsten beiden Unterkapiteln.

Zunächst berechnen wir eine orthogonale Projektion f_0 von f in den Approximationsraum V_0 , der in (2.11) definiert wurde. Alternativ können wir auch eine Projektion f_j in einen gröberen Raum V_j mit $j < 0$ betrachten. Danach wenden wir einen Filterbank-Algorithmus an, der durch die hergeleiteten Zerlegungs- und Rekonstruktionsformeln der \mathcal{AB} -Skalierungsfunktionen ϕ_i und \mathcal{AB} -Wavelets $\psi_i^1, \psi_i^2, \psi_i^3, i = 0, \dots, 15$, bestimmt wird. Dadurch zerlegen wir f_j in den Tiefpass-Anteil $f_{j-1} \in V_{j-1}$ und den Hochpass-Anteil $g_{j-1} \in W_{j-1}$. Dies kann durch vierfache Anwendung der Schnellen Wavelettransformation (FWT) geschehen, weil unser Waveletframe in vier Orthogonalbasen aufgeteilt werden kann.

Wenn wir unsere Filterbank zur Bildentstörung benutzen, erweist sich die vierfache Redundanz als vorteilhaft. Im Gegensatz dazu müssen bei der Bildapproximation die Redundanzen des Waveletframes verringert werden, um eine dünne Darstellung des Bildes zu gewährleisten. Diesen Überlegungen widmen wir das ganze Unterkapitel 2.2.4.

Projektion in den Raum V_0 Um die auf den vorhergehenden Seiten dargestellten Haarwavelet-Frames anzuwenden, benötigen wir zunächst eine geeignete Projektion f_0 der Bildfunktion f aus (2.24) in den Startraum V_0 aus (2.11). Wir stellen zwei Forderungen an diese Projektion. Erstens soll die zwangsläufig entstehende Redundanz so gering wie möglich sein. Zweitens soll es keinen Informationsverlust geben, d. h. die Bildfunktion f soll perfekt rekonstruierbar sein aus f_0 .

Die orthogonale Projektion von f in den Skalierungsraum V_0 hat die Gestalt

$$f_0 = \sum_{k \in I_1} (c_k^0)^T \Phi(\cdot - k), \quad (2.25)$$

dabei ist $I_1 := \{0, \dots, N-1\}^2$, $c_k^0 = (c_{0,k}^0, \dots, c_{15,k}^0)^T \in \mathbb{R}^{16}$. Beachte, dass alle Träger der \mathcal{AB} -Skalierungsfunktionen in Φ in dem Quadrat $[0, 1]^2$ enthalten sind, siehe (2.10). Die weitere Analyse der orthogonalen Projektion kann – was die Notation angeht – wesentlich vereinfacht werden, wenn wir uns nur auf dieses Quadrat $[0, 1]^2$ beschränken, anstatt die Funktion auf ganz $[0, 2N]^2$ zu betrachten. Diese Beschränkung ist zulässig, weil sowohl die Basisfunktionen $\chi_{[0,1]^2}(2 \cdot -k)$ in (2.24) als auch die \mathcal{AB} -Skalierungsfunktionen in $\Phi(\cdot - k)$ aus (2.25) einen kleinen kompakten Träger haben. Dann benötigen wir eine Projektion $f_0|_{[0,1]^2}(x) = (c_0^0)^T \Phi(x)$ der Funktion

$$f|_{[0,1]^2}(x) = a_{\binom{0}{0}} \chi_{[0,1]^2}(2x) + a_{\binom{1}{0}} \chi_{[0,1]^2}\left(2x - \binom{1}{0}\right) + a_{\binom{0}{1}} \chi_{[0,1]^2}\left(2x - \binom{0}{1}\right) + a_{\binom{1}{1}} \chi_{[0,1]^2}\left(2x - \binom{1}{1}\right),$$

$x \in [0, 1]^2$. Eine solche Projektion liefert offensichtlich den Redundanzfaktor 4, denn aus den vier Pixelwerten $a_{\binom{0}{0}}, a_{\binom{1}{0}}, a_{\binom{0}{1}}, a_{\binom{1}{1}}$ im Quadrat $[0, 1]^2$ entstehen 16 Koeffizienten in c_0^0 der \mathcal{AB} -Skalierungsfunktionen in Φ .

Wir leiten nun eine Berechnungsvorschrift für den Koeffizientenvektor $c_0^0 \in \mathbb{R}^{16}$ her. Mithilfe des kanonischen Dualframes $\tilde{\Phi} = G^T \Phi$ aus (2.19) ist

$$\begin{aligned} c_0^0 &= \langle \tilde{\Phi}, f \rangle = G^T \langle \Phi, f \rangle \\ &= a_{\binom{0}{0}} G^T \langle \Phi, \chi_{[0,1]^2}(2 \cdot) \rangle + a_{\binom{1}{0}} G^T \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{1}{0}\right) \rangle \\ &\quad + a_{\binom{0}{1}} G^T \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{0}{1}\right) \rangle + a_{\binom{1}{1}} G^T \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{1}{1}\right) \rangle. \end{aligned}$$

Die vier auftretenden Vektoren

$$\langle \Phi, \chi_{[0,1]^2}(2 \cdot - k') \rangle = \frac{1}{4} \int_{[0,1]^2} \Phi\left(\frac{y+k'}{2}\right) dy, \quad k' \in \left\{ \binom{0}{0}, \binom{1}{0}, \binom{0}{1}, \binom{1}{1} \right\},$$

in \mathbb{R}^{16} sind nun einfach zu berechnen. Wir erhalten

$$\begin{aligned} M &:= \left(\langle \Phi, \chi_{[0,1]^2}(2 \cdot) \rangle, \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{1}{0}\right) \rangle, \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{0}{1}\right) \rangle, \langle \Phi, \chi_{[0,1]^2}\left(2 \cdot - \binom{1}{1}\right) \rangle \right) \\ &= \frac{1}{16} \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 3 & 1 & 0 & 2 & 0 & 2 & 3 & 1 & 0 & 0 \\ 0 & 0 & 3 & 1 & 0 & 2 & 0 & 2 & 3 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 3 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 3 & 1 & 0 & 2 & 0 & 2 \\ 0 & 2 & 0 & 2 & 3 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 3 & 1 \end{pmatrix}^T. \end{aligned}$$

Damit folgt für die Koeffizienten c_0^0 schließlich

$$c_0^0 = G^T M \left(a_{\binom{0}{0}}, a_{\binom{1}{0}}, a_{\binom{0}{1}}, a_{\binom{1}{1}} \right)^T.$$

Die übrigen Koeffizienten-Vektoren c_k^0 der translatierten Framefunktionen $\Phi(\cdot - k)$ in (2.25) erhält man dann für alle $k \in I_1$ durch

$$c_k^0 = G^T M \left(a_{2k}, a_{2k+\binom{1}{0}}, a_{2k+\binom{0}{1}}, a_{2k+\binom{1}{1}} \right)^T.$$

Als nächstes werden wir zeigen, dass die Projektion f_0 in (2.25) auch unsere zweite Forderung erfüllt, da sie die volle Information über die Bildfunktion f enthält und somit eine perfekte Rekonstruktion von f gewährleistet. Wegen der Redundanz in Φ benötigen wir für die Rekonstruktion von f sogar nur vier Framefunktionen ϕ_i . Wir wählen die \mathcal{AB} -Skalierungsfunktionen ϕ_0, ϕ_1, ϕ_8 und ϕ_9 . Die \mathbb{Z}^2 -Translate dieser Funktionen spannen den Unterraum V_0^0 von V_0 auf. Die orthogonale Projektion von f aus (2.24) in V_0^0 hat die Form

$$f_0^0 = \sum_{k \in I_1} c_{0,k}^0 \phi_{0,0,k} + c_{1,k}^0 \phi_{1,0,k} + c_{8,k}^0 \phi_{8,0,k+\binom{1}{1}} + c_{9,k}^0 \phi_{9,0,k+\binom{1}{1}}.$$

Auf die gleiche Weise wie zuvor berechnen wir die vier Koeffizienten

$$\begin{pmatrix} c_{0,k}^0 \\ c_{1,k}^0 \\ c_{8,k}^0 \\ c_{9,k}^0 \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 1 & 0 & 3 & 0 \\ 1 & 0 & 1 & 2 \\ 0 & 3 & 0 & 1 \\ 2 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{2k} \\ a_{2k+\binom{1}{0}} \\ a_{2k+\binom{0}{1}} \\ a_{2k+\binom{1}{1}} \end{pmatrix}. \quad (2.26)$$

Die hierbei verwendete Koeffizientenmatrix setzt sich aus der 0ten, 1ten, 8ten und 9ten Zeile von M zusammen, denn eben diese Zeilen von M enthalten die Koeffizienten von $\phi_{0,0,k}, \phi_{1,0,k}, \phi_{8,0,k+\binom{1}{1}}$ und $\phi_{9,0,k+\binom{1}{1}}$. Die Gram'sche Matrix hat hier die Form $\frac{1}{4}I_4$, weil die vier Funktionen orthogonal zueinander sind. Die Gleichung (2.26) wird völlig klar, wenn wir anhand von Abbildung 2.9 beispielsweise den ersten Koeffizienten $c_{0,k}^0$ betrachten. Das linke Dreieck ist der Träger von $\phi_{0,0,k}$, der dazugehörige Koeffizient $c_{0,k}^0$ setzt sich also nur aus den beiden Pixelwerten a_{2k} und $a_{2k+\binom{0}{1}}$ zusammen, und zwar gemäß der überdeckten Fläche mit den Gewichten $\frac{1}{4}$ und $\frac{3}{4}$. Das ist genau die Gleichung für die erste Komponente des Koeffizientenvektors in (2.26),

$$c_{0,k}^0 = \frac{1}{4} a_{2k} + \frac{3}{4} a_{2k+\binom{0}{1}}.$$

Weil die Koeffizientenmatrix in (2.26) invertierbar ist, können wir f aus f_0^0 rekonstruieren. Aber kann man auch f aus f_0 rekonstruieren? Ja, denn V_0^0 ist ein Unterraum von V_0 , und es gilt insbesondere, dass f_0^0 eine orthogonale Projektion von f_0 in V_0^0 ist. D. h. also,

$$c_{0,k}^0 = \langle f_0, \phi_{0,0,k} \rangle = \langle (c_k^0)^T \Phi(\cdot - k), \phi_{0,0,k} \rangle = (c_k^0)^T g_0,$$

wobei g_0 den 0-ten Spaltenvektor der Gram'schen Matrix G bezeichnet. Analog bekommen wir mit der 1ten, 8ten und 9ten Spalte von G

$$c_{1,k}^0 = (c_k^0)^T g_1, \quad c_{8,k}^0 = (c_k^0)^T g_8, \quad c_{9,k}^0 = (c_k^0)^T g_9.$$

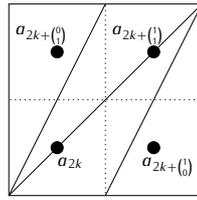


Abbildung 2.9: Berechnung der vier Koeffizienten $c_{0,k}^0, c_{1,k}^0, c_{8,k}^0, c_{9,k}^0$ aus vier Bildwerten.

Insgesamt erhalten wir also für alle $k \in I_1$

$$\begin{pmatrix} c_{0,k}^0 \\ c_{1,k}^0 \\ c_{8,k}^0 \\ c_{9,k}^0 \end{pmatrix} = (g_0, g_1, g_8, g_9)^T c_k^0 = G'^T c_k^0, \quad (2.27)$$

wobei $G' \in \mathbb{R}^{16 \times 4}$ aus der 0ten, 1ten, 8ten und 9ten Spalte von G besteht. Setzen wir (2.27) mit (2.26) gleich, so erhalten wir die gewünschte Rekonstruktionsformel

$$\begin{pmatrix} a_{2k} \\ a_{2k+\binom{1}{0}} \\ a_{2k+\binom{0}{1}} \\ a_{2k+\binom{1}{1}} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 1 & -3 & -3 & 9 \\ 1 & -3 & 5 & 1 \\ 5 & 1 & 1 & -3 \\ -3 & 9 & 1 & -3 \end{pmatrix} G'^T c_k^0.$$

An Stelle des Unterraums V_0^0 können wir auch einen der anderen Unterräume V_0^1, V_0^2 oder V_0^3 betrachten mit entsprechenden Koeffizienten-Matrizen.

Filterbank mit gerichteten Haarwavelets Sei nun $f_0 \in V_0$ wie in (2.25) gegeben. Wir möchten einen effizienten Filterbank-Algorithmus herleiten, um f_0 in $f_{-1} \in V_{-1}$ und $g_{-1} \in W_{-1}$ zu zerlegen sowie zur Rekonstruktion mittels $f_{-1} + g_{-1}$. Mit der Indexmenge $I_2 := \{0, \dots, \frac{N}{2} - 1\}^2$ können wir schreiben

$$\begin{aligned} f_0 = \sum_{k \in I_2} & (c_{2k}^0)^T \Phi(\cdot - 2k) + (c_{2k+\binom{1}{0}}^0)^T \Phi(\cdot - (2k + \binom{0}{1})) \\ & + (c_{2k+\binom{0}{1}}^0)^T \Phi(\cdot - (2k + \binom{0}{1})) + (c_{2k+\binom{1}{1}}^0)^T \Phi(\cdot - (2k + \binom{1}{1})). \end{aligned} \quad (2.28)$$

Wir beschränken uns bei der Zerlegung von f_0 wieder auf eine lokale Betrachtung. Weil sich die Trägergröße der Funktionen im nächsten, größeren Level $j = -1$ verdoppelt, betrachten wir nun das Quadrat $[0, 2)^2$. Dort gibt es 64 „feine“ Framefunktionen in V_0 , nämlich die 16 Komponenten des Skalierungsfunktions-Vektors Φ mit Trägern in $[0, 1)^2$ sowie die translatierten \mathcal{AB} -Skalierungsfunktionen in $\Phi(\cdot - \binom{1}{0})$, $\Phi(\cdot - \binom{0}{1})$ und $\Phi(\cdot - \binom{1}{1})$. Um einen effizienten Filterbank-Algorithmus zu erhalten, machen wir Gebrauch von dem Umstand, dass

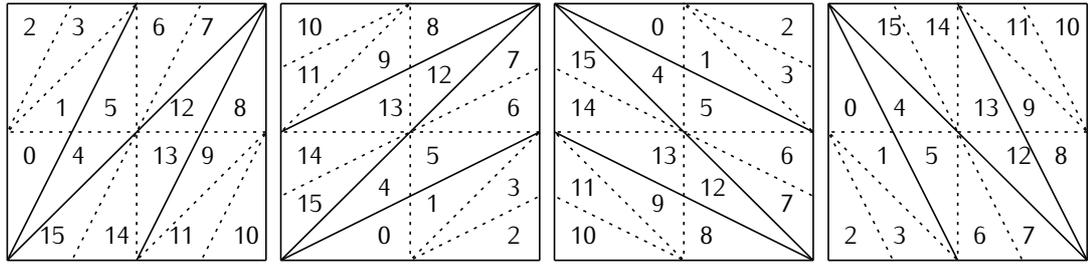


Abbildung 2.10: Reihenfolge der Komponenten in den Funktionen-Vektoren Φ_0^0 (links), Φ_0^1 (mitte links), Φ_0^2 (mitte rechts) und Φ_0^3 (rechts). Die Abbildungen zeigen die Indizierung der Träger der i -ten Komponente von Φ_0^ν für $i = 0, \dots, 15$.

V_j gemäß Lemma 2.5 in vier Unterräume V_j^ν , $\nu = 0, 1, 2, 3$, zerlegt werden kann, d. h.

$$V_j = V_j^0 + V_j^1 + V_j^2 + V_j^3.$$

Wir ordnen deshalb die 64 Framefunktionen in dem Skalierungsfunktions-Vektor $(\Phi^T, \Phi(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix})^T, \Phi(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix})^T, \Phi(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix})^T)$ nach den vier Richtungen der Unterräume, so dass wir einen neuen Funktionen-Vektor

$$\Phi_0 := (\Phi_0^{0T}, \Phi_0^{1T}, \Phi_0^{2T}, \Phi_0^{3T})^T \quad (2.29)$$

gleicher Länge erhalten. Hierbei enthält Φ_0^ν die 16 \mathcal{AB} -Skalierungsfunktionen in $[0, 2)^2$, die im Unterraum V_0^ν , $\nu = 0, 1, 2, 3$, liegen. Konkret bedeutet das

$$\Phi_0^0 := \left(\phi_{0,0,\begin{pmatrix} 0 \\ 0 \end{pmatrix}}, \phi_{9,0,\begin{pmatrix} 1 \\ 2 \end{pmatrix}}, \phi_{0,0,\begin{pmatrix} 0 \\ 1 \end{pmatrix}}, \phi_{1,0,\begin{pmatrix} 0 \\ 1 \end{pmatrix}}, \phi_{1,0,\begin{pmatrix} 0 \\ 0 \end{pmatrix}}, \phi_{8,0,\begin{pmatrix} 1 \\ 2 \end{pmatrix}}, \phi_{0,0,\begin{pmatrix} 1 \\ 1 \end{pmatrix}}, \phi_{1,0,\begin{pmatrix} 1 \\ 1 \end{pmatrix}}, \right. \\ \left. \phi_{8,0,\begin{pmatrix} 2 \\ 2 \end{pmatrix}}, \phi_{1,0,\begin{pmatrix} 1 \\ 0 \end{pmatrix}}, \phi_{8,0,\begin{pmatrix} 2 \\ 1 \end{pmatrix}}, \phi_{9,0,\begin{pmatrix} 2 \\ 1 \end{pmatrix}}, \phi_{9,0,\begin{pmatrix} 2 \\ 2 \end{pmatrix}}, \phi_{0,0,\begin{pmatrix} 1 \\ 0 \end{pmatrix}}, \phi_{8,0,\begin{pmatrix} 1 \\ 1 \end{pmatrix}}, \phi_{9,0,\begin{pmatrix} 1 \\ 1 \end{pmatrix}} \right)^T$$

und damit

$$\Phi_0^\nu := \Phi_0^0(B_\nu \cdot), \quad \nu = 1, 2, 3.$$

In der Abbildung 2.10 ist die neue Reihenfolge der Funktionen innerhalb der Vektoren Φ_0^ν anhand ihrer Trägerdreiecke dargestellt. Diese Umordnung der \mathcal{AB} -Skalierungsfunktionen erlaubt uns im Folgenden eine vereinfachte Zerlegung in \mathcal{AB} -Wavelets und gröbere \mathcal{AB} -Skalierungsfunktionen. In Lemma 2.5 hatten wir bewiesen, dass jede Folge $\{V_j^\nu\}_{j \in \mathbb{Z}}$ eine MRA des $L^2(\mathbb{R}^2)$ bildet. Das bedeutete insbesondere, dass die Verfeinerungsgleichungen der \mathcal{AB} -Skalierungsfunktionen (2.12), (2.13) und der dazugehörigen \mathcal{AB} -Wavelets (2.20) – (2.21) nur \mathcal{AB} -Skalierungsfunktionen aus dem gleichen Unterraum (d. h. gleicher Richtung) benötigen. Deshalb können wir jetzt den Zerlegungsalgorithmus getrennt für jeden Unterraum V_j^ν anwenden, also für $j = 0$

$$V_0^\nu = V_{-1}^\nu \oplus W_{-1}^\nu, \quad \nu = 0, 1, 2, 3.$$

Dies geschieht mittels der Haarwavelet-Transformationsmatrix W und der Blockdiagonal-Matrix A

$$W = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}, \quad A := \begin{pmatrix} W & & & \\ & W & & \\ & & W & \\ & & & W \end{pmatrix}$$

durch den Basiswechsel

$$\Psi_{-1}^{\nu} = A \Phi_0^{\nu}, \quad \nu = 0, 1, 2, 3,$$

wobei Ψ_{-1}^{ν} nun die Basisfunktionen von $V_{-1}^{\nu} \oplus W_{-1}^{\nu}$ im Quadrat $[0, 2)^2$ enthält, also vier Skalierungs- und zwölf Waveletfunktionen. Genauer gesagt ist

$$\Psi_{-1}^0 := \left(\phi_{0,-1,\binom{0}{0}}, \psi_{0,-1,\binom{0}{0}}^1, \psi_{0,-1,\binom{0}{0}}^2, \psi_{0,-1,\binom{0}{0}}^3, \phi_{1,-1,\binom{0}{0}}, \psi_{1,-1,\binom{0}{0}}^1, \psi_{1,-1,\binom{0}{0}}^2, \psi_{1,-1,\binom{0}{0}}^3, \right. \\ \left. \phi_{8,-1,\binom{1}{1}}, \psi_{8,-1,\binom{1}{1}}^1, \psi_{8,-1,\binom{1}{1}}^2, \psi_{8,-1,\binom{1}{1}}^3, \phi_{9,-1,\binom{1}{1}}, \psi_{9,-1,\binom{1}{1}}^1, \psi_{9,-1,\binom{1}{1}}^2, \psi_{9,-1,\binom{1}{1}}^3 \right)^T,$$

und

$$\Psi_{-1}^{\nu} := \Psi_{-1}^0(B_{\nu} \cdot), \quad \nu = 1, 2, 3.$$

Jetzt untersuchen wir, was diese Waveletzerlegung für die entsprechenden Koeffizienten bedeutet. Die Umordnung nach Richtungen, die wir im Vektor der \mathcal{AB} -Skalierungsfunktionen vorgenommen haben, führen wir analog in den Koeffizienten-Vektoren $c_{2k+k'}^0$, $k' \in \left\{ \binom{0}{0}, \binom{1}{0}, \binom{0}{1}, \binom{1}{1} \right\}$, aus (2.28) durch. Wir erhalten dann die lokale Darstellung

$$f_0|_{[0,2)^2} = (d_0^0)^T \Phi_0^0 + (d_0^1)^T \Phi_0^1 + (d_0^2)^T \Phi_0^2 + (d_0^3)^T \Phi_0^3 = D_{0,0}^T \Phi_0$$

mit $D_{0,0}^T := ((d_0^0)^T, (d_0^1)^T, (d_0^2)^T, (d_0^3)^T) \in \mathbb{R}^{64}$. Wegen $A = A^T = A^{-1}$ können wir nun die Frame-Transformation von V_0 nach $V_{-1} + W_{-1}$ vornehmen mittels

$$f_0|_{[0,2)^2} = D_{0,0}^T (I_4 \otimes A) \Psi_{-1}.$$

Hier enthält

$$\Psi_{-1} := (\Psi_{-1}^{0T}, \Psi_{-1}^{1T}, \Psi_{-1}^{2T}, \Psi_{-1}^{3T})^T$$

die Framefunktionen in $V_{-1} + W_{-1}$. Die neuen Koeffizienten haben bzgl. dieser Darstellung die Gestalt $D_{-1,0}^T := D_{0,0}^T (I_4 \otimes A) \in \mathbb{R}^{64}$.

Nachdem wir uns die Zerlegung für das erste Level lokal klar gemacht haben, können wir nun den gesamten (globalen) Zerlegungsalgorithmus angeben, der auf den gerichteten Haarwavelets auf Dreiecken basiert.

Algorithmus 1a: Zerlegung durch gerichtete Haarwavelet-Filterbank

Input: Startbild $\mathbf{a} = (a[m_1, m_2])_{m_1, m_2=0}^{2^N-1}$ mit $N = 2^J, J \in \mathbb{N}$, das die zugehörige Bildfunktion f in (2.24) liefert.

1. Berechne die orthogonale Projektion f_0 der Bildfunktion f in den Raum V_0 ,

$$f_0 = \sum_{k \in I_1} (c_{0,k}^0)^T \Phi(\cdot - k), \quad c_{0,k}^0 = \langle f, \tilde{\Phi}(\cdot - k) \rangle.$$

2. Ordne die Framefunktionen, die den Raum V_0 erzeugen, bzw. die entsprechenden Koeffizienten nach den vier Richtungen. Sei P die Permutationsmatrix, die diese Umordnung vornimmt,

$$\left(\Phi^T, \Phi \left(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right)^T, \Phi \left(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right)^T, \Phi \left(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right)^T \right) P = \Phi_0^T.$$

Ordne für jedes $k \in I_2$ die Koeffizienten-Vektoren um

$$\begin{aligned} D_{0,k}^T &:= ((d_{0,k}^0)^T, (d_{0,k}^1)^T, (d_{0,k}^2)^T, (d_{0,k}^3)^T) \\ &= ((c_{0,2k}^0)^T, (c_{0,2k+\binom{0}{1}}^0)^T, (c_{0,2k+\binom{0}{1}}^0)^T, (c_{0,2k+\binom{1}{1}}^0)^T) P, \end{aligned}$$

so dass

$$f_0 = \sum_{k \in I_2} D_{0,k}^T \Phi_0(\cdot - 2k). \quad (2.30)$$

3. Zerlege $f_0 \in V_0$ in $f_{-1} \in V_{-1}$ und $g_{-1} \in W_{-1}$ mittels $\Phi_0 = (I_4 \otimes A) \Psi_{-1}$,

$$f_{-1} + g_{-1} = \sum_{k \in I_2} D_{-1,k}^T \Psi_{-1}(\cdot - 2k). \quad (2.31)$$

Berechne die zugehörigen Koeffizienten durch

$$D_{-1,k}^T = D_{0,k}^T (I_4 \otimes A).$$

4. Mit der Definition von Ψ_{-1} ordne die Koeffizienten von

$$D_{-1,k}^T = ((d_{-1,k}^0)^T, (d_{-1,k}^1)^T, (d_{-1,k}^2)^T, (d_{-1,k}^3)^T), \quad k \in I_2,$$

nach Tiefpass- und Hochpasskoeffizienten, um schließlich f_{-1} und g_{-1} zu erhalten.

5. Iterative Anwendung: Wende die Zerlegungsschritte 2 bis 4 auf den Tiefpassteil f_{-1} an, während der Hochpassanteil g_{-1} abgespeichert wird.

Output: Zerlegtes Bild $\tilde{\mathbf{a}}$.

Wir stellen fest, dass der Zerlegungsalgorithmus lediglich aus Permutationen und Additionen bzw. Subtraktionen besteht. Weil die Transformationsmatrix $(I_4 \otimes A)$ orthogonal ist, ist der Algorithmus numerisch stabil. Der Rekonstruktionsalgorithmus der Filterbank folgt sofort, indem man die einzelnen Schritte rückwärts und invers ausführt.

2.2.4 Dünne Darstellung von Bildern in Waveleträumen

Das Problem, aus einem redundanten Dictionary $\{g_\mu\}_{\mu=0}^P$ mit $P > M$ die M Elemente auszuwählen, die die beste M -Term-Approximation f_M von f liefern, ist für allgemeine Dictionaries *NP-schwer* [DMA97]. Mithilfe von Pursuit-Algorithmen lässt sich der Berechnungsaufwand reduzieren, indem man auf effizientem Weg nach einer fast optimalen Lösung sucht. Eine übliche Pursuit-Methode ist das Orthogonal-Matching-Pursuit (OMP), siehe [DE03, Tro04] und die dort aufgeführten Referenzen. Bei diesem iterativen Greedy-Algorithmus wird in jedem Schritt das Dictionary-Element ausgewählt, das am besten mit dem Residuum des Signals aus dem vorhergehenden Schritt korreliert. Nach Hinzunahme dieser Frame- bzw. Basisfunktion erhält man eine neue Approximation des Signals, indem man eine Projektion in den Raum berechnet, der von den bereits ausgewählten Dictionary-Funktionen aufgespannt wird.

Nach [Tro04] hängt die Konvergenz des OMP-Algorithmus im Wesentlichen von der Kohärenz des Dictionaries ab, d. h. von der maximal möglichen Korrelation $|\langle g_\mu, g_{\mu'} \rangle|$ zwischen zwei Elementen. Wegen der hohen Kohärenz unseres Haarwavelet-Dictionaries liefert das OMP-Verfahren in unserem Fall leider keine guten Ergebnisse. Um eine dünne Darstellung des Bildes $f_0 \in V_0$ zu erhalten, müssen wir die bestehende Redundanz verringern. Dabei versuchen wir, unsere Kenntnis über das gerichtete Haarwavelet-Frame auszunutzen. Der OMP-Algorithmus benutzt eine solche Information nicht, bei vielen Dictionaries sind die Abhängigkeiten zwischen den Framefunktionen auch gar nicht bekannt. In unserem Fall dagegen wissen wir exakt, welche Abhängigkeiten vorliegen; im Beweis von Satz 2.3 hatten wir die Redundanz-Beziehungen eingehend untersucht. Das wollen wir uns nun bei der numerischen Anwendung zunutze machen.

Die Idee ist die folgende: Nach der Zerlegung eines gegebenen Bildes $f_j \in V_j$ in die beiden Anteile $f_{j-1} \in V_{j-1}$ und $g_{j-1} \in W_{j-1}$ versuchen wir die vorherrschenden Beziehungen zwischen den Framefunktionen, die die Räume V_{j-1} und W_{j-1} aufspannen, zu benutzen, um eine Darstellung von f_{j-1} und g_{j-1} zu erhalten, die möglichst viele Null-Koeffizienten enthält. Diese Prozedur wenden wir nach jedem Zerlegungslevel an. Schließlich unterwerfen wir – wie gewohnt – die übriggebliebenen Koeffizienten einer Shrinkage-Prozedur und erhalten auf diese Weise eine Approximation des Bildes, die aufgrund der gerichteten Framefunktionen verschiedene Orientierungen im Bild gut adaptiert.

Wir betrachten wieder die Zerlegung $V_0 = V_0^0 + V_0^1 + V_0^2 + V_0^3$, und erinnern daran, dass V_0 durch $\{\Phi_0(\cdot - 2k) : k \in \mathbb{Z}^2\}$ aufgespannt wird, wobei der in (2.29) definierte Skalierungsfunktionen-Vektor Φ_0 der Länge 64 die Framefunktionen geordnet nach Richtungen enthält. Außerdem war $V_0^v = \text{clos}_{L^2(\mathbb{R}^2)} \text{span} \{\Phi_0^v(\cdot - 2k) : k \in \mathbb{Z}^2\}$ für $v = 0, 1, 2, 3$, siehe Abbildung 2.8.

Die Darstellungen von f_0 in V_0 bzw. in $V_{-1} + W_{-1}$ sind in (2.30) bzw. (2.31) angegeben. Auf-

grund der Frame-Eigenschaft der aufspannenden Funktionen sind diese Darstellungen nicht eindeutig. Die Abhängigkeitsbeziehungen aus (2.6) zwischen den \mathcal{AB} -Skalierungsfunktionen in V_0 formulieren wir jetzt mithilfe der eingeführten Funktionen-Vektoren. Mit den 16 Einheitsvektoren $e_k := (\delta_{k,l})_{l=0}^{15}$ für $k = 0, \dots, 15$, gelten die Gleichungen

$$\begin{aligned} (e_0 + e_4)^T \Phi_0^v - (e_{14} + e_{15})^T \Phi_0^{v+1} &= 0, \\ (e_2 + e_3)^T \Phi_0^v - (e_{10} + e_{11})^T \Phi_0^{v+1} &= 0, \\ (e_9 + e_{13})^T \Phi_0^v - (e_1 + e_5)^T \Phi_0^{v+1} &= 0, \\ (e_6 + e_7)^T \Phi_0^v - (e_8 + e_{12})^T \Phi_0^{v+1} &= 0, \\ (e_0 + e_4)^T \Phi_0^{v+1} - (e_{14} + e_{15})^T \Phi_0^v &= 0, \\ (e_2 + e_3)^T \Phi_0^{v+1} - (e_{10} + e_{11})^T \Phi_0^v &= 0, \\ (e_9 + e_{13})^T \Phi_0^{v+1} - (e_1 + e_5)^T \Phi_0^v &= 0, \\ (e_6 + e_7)^T \Phi_0^{v+1} - (e_8 + e_{12})^T \Phi_0^v &= 0, \end{aligned}$$

für $v = 0, 1$, und

$$\begin{aligned} (e_0 + e_4)^T (\Phi_0^0 + \Phi_0^1) - (e_9 + e_{13})^T \Phi_0^2 - (e_2 + e_3)^T \Phi_0^3 &= 0, \\ (e_9 + e_{13})^T \Phi_0^0 + (e_2 + e_3)^T \Phi_0^1 - (e_6 + e_7)^T (\Phi_0^2 + \Phi_0^3) &= 0, \\ (e_2 + e_3)^T \Phi_0^0 + (e_9 + e_{13})^T \Phi_0^1 - (e_0 + e_4)^T (\Phi_0^2 + \Phi_0^3) &= 0, \\ (e_6 + e_7)^T (\Phi_0^0 + \Phi_0^1) - (e_2 + e_3)^T \Phi_0^2 - (e_9 + e_{13})^T \Phi_0^3 &= 0. \end{aligned}$$

Diese 20 Gleichungen lassen sich in einer Matrix-Vektor-Form schreiben. Dazu soll $U \in \mathbb{R}^{20 \times 64}$ diese Abhängigkeiten derart enthalten, dass gilt

$$U ((\Phi_0^0)^T, (\Phi_0^1)^T, (\Phi_0^2)^T, (\Phi_0^3)^T)^T = U \Phi_0 = 0 \quad \text{auf } [0, 2)^2. \quad (2.32)$$

Wir können nun der Darstellung von f_0 in (2.30) diese Redundanz-Beziehungen hinzufügen, und erhalten für beliebige Vektoren $g_k \in \mathbb{R}^{20}$, $k \in I_2$, eine redundante Darstellung von $f_0 \in V_0$ der Gestalt

$$f_0 = \sum_{k \in I_2} D_{0,k}^T \Phi_0(\cdot - 2k) = \sum_{k \in I_2} (D_{0,k}^T + g_k^T U) \Phi_0(\cdot - 2k).$$

Die Transformation von V_0 in $V_{-1} + W_{-1}$ liefert die Darstellung

$$f_0 = \sum_{k \in I_2} (D_{0,k}^T + g_k^T U) (I_4 \otimes A) \Psi_{-1}(\cdot - 2k).$$

Um unser Ziel zu erreichen, eine dünne Darstellung von f_0 in $V_{-1} + W_{-1}$ mit möglichst wenigen Nichtnull-Koeffizienten zu erhalten, wählen wir unter den unendlich vielen Darstellungsmöglichkeiten diejenige aus, bei der in dem Koeffizientenvektor $(D_{0,k}^T + g_k^T U) (I_4 \otimes A)$ möglichst viele Komponenten verschwinden. Wegen der lokalen Träger von Φ_0 bzw. Ψ_{-1} , kann

dieses Problem für jedes $k \in I_2$ einzeln gelöst werden. Also müssen wir für jedes $k \in I_2$ den Vektor $g_k \in \mathbb{R}^{20}$ finden, so dass die l^0 -Seminorm

$$\left\| \left(D_{0,k}^T + g_k^T U \right) (I_4 \otimes A) \right\|_0$$

minimal wird. Die l^0 -Seminorm eines Vektors ist die Anzahl der Nichtnull-Elemente; diese Abbildung nennen wir Seminorm, weil die Homogenität nicht gilt. Dieser zusätzliche Minimierungsschritt führt zu einem modifizierten Zerlegungsalgorithmus.

Algorithmus 1b: Zerlegung mit Redundanz-Reduktion

Input: Startbild $\mathbf{a} = (a[m_1, m_2])_{m_1, m_2=0}^{2N-1}$ mit $N = 2^J, J \in \mathbb{N}$, das die zugehörige Bildfunktion f in (2.24) liefert.

1. Berechne die orthogonale Projektion f_0 der Bildfunktion f in den Raum V_0 ,

$$f_0 = \sum_{k \in I_1} (c_{0,k}^0)^T \Phi(\cdot - k), \quad c_{0,k}^0 = \langle f, \tilde{\Phi}(\cdot - k) \rangle.$$

2. Ordne die Framefunktionen um (bzw. die entsprechenden Koeffizienten) nach Richtungen (siehe Schritt 2 von Algorithmus 1a),

$$f_0 = \sum_{k \in I_2} D_{0,k}^T \Phi_0(\cdot - 2k).$$

3. Füge Redundanzen in V_0 hinzu und führe die Transformation nach $V_{-1} + W_{-1}$ durch:

$$f_0 = \sum_{k \in I_2} (D_{0,k}^T + g_k^T U) \Phi_0(\cdot - 2k) = \sum_{k \in I_2} (D_{0,k}^T + g_k^T U)(I_4 \otimes A) \Psi_{-1}(\cdot - 2k).$$

4. Für jedes $k \in I_2$ berechne $g_k \in \mathbb{R}^{20}$ so, dass die l^0 -Seminorm

$$\left\| (D_{0,k}^T + g_k^T U)(I_4 \otimes A) \right\|_0 = \left\| D_{-1,k}^T + g_k^T U(I_4 \otimes A) \right\|_0$$

mit $D_{-1,k}^T = D_{0,k}^T(I_4 \otimes A)$ minimal wird.

5. Für jedes $k \in I_2$ sei $\tilde{D}_{-1,k}^T = D_{-1,k}^T + \tilde{g}_k^T U(I_4 \otimes A)$ dieser minimale Koeffizienten-Vektor, wobei

$$\tilde{g}_k := \arg \min_{g_k \in \mathbb{R}^{20}} \left\| D_{-1,k}^T + g_k^T U(I_4 \otimes A) \right\|_0.$$

Berechne die dünne Darstellung

$$f_0 = \sum_{k \in I_2} \tilde{D}_{-1,k}^T \Psi_{-1}(\cdot - 2k),$$

und entnehme $f_{-1} \in V_{-1}$ und $g_{-1} \in W_{-1}$ aus dieser Darstellung.

6. Iterative Anwendung: Wende die Zerlegungsschritte 2 bis 5 iterativ auf den Tiefpass-Anteil f_{-1} an, während der Hochpass-Anteil g_{-1} abgespeichert wird.

Output: Zerlegtes Bild $\tilde{\mathbf{a}}$.

Im Folgenden untersuchen wir den entscheidenden Schritt 4, in dem das Minimierungsproblem

$$\arg \min_{g_k \in \mathbb{R}^{20}} \{\|D_{-1,k} + (I_4 \otimes A) U^T g_k\|_0\}$$

für jedes $k \in I_2$ gelöst wird. Unsere Überlegungen werden in einen Minimierungsalgorithmus münden.

Zunächst vereinfachen wir die Notation, indem wir die Matrix $R := (I_4 \otimes A) U^T \in \mathbb{R}^{64 \times 20}$ definieren, die ebenso wie $D_{-1,k} \in \mathbb{R}^{64}$ gegeben ist. Beachte, dass R den vollen Rang 20 besitzt. Wir haben also das Minimierungsproblem

$$\arg \min_{g_k \in \mathbb{R}^{20}} \{\|D_{-1,k} + R g_k\|_0\} \quad (2.33)$$

zu lösen, wobei die Behandlung der l^0 -Seminorm die wesentliche Hürde ist. Leider erweist sich auch das Ersetzen der l^0 -Seminorm durch die l^1 -Norm nicht als zielführend. Denn betrachte zum Beispiel für einen konstanten Teil eines Bildes, d. h. $f_0 = c$ auf $[0, 2]^2$, die Koeffizienten-Vektoren von zwei möglichen Darstellungen von f_0 ,

$$f_0 = c(1_{16}^T, 0_{16}^T, 0_{16}^T, 0_{16}^T) \Phi_0 = c(w^T, 0_{16}^T, 0_{16}^T, 0_{16}^T) \Psi_{-1}$$

und

$$f_0 = \frac{c}{4}(1_{16}^T, 1_{16}^T, 1_{16}^T, 1_{16}^T) \Phi_0 = \frac{c}{4}(w^T, w^T, w^T, w^T) \Psi_{-1}.$$

Beide Vektoren haben dieselbe l^1 -Norm während sich ihre l^0 -Seminorm wesentlich unterscheidet! In diesem Beispiel ist $1_{16} := (1, \dots, 1)^T \in \mathbb{R}^{16}$, und 0_{16} bezeichnet den Null-Vektor der Länge 16, und

$$w^T := (2, 0, 0, 0, 2, 0, 0, 0, 2, 0, 0, 0, 2, 0, 0, 0) \in \mathbb{R}^{16}. \quad (2.34)$$

Ein einfacher Ansatz zur Lösung von (2.33) ist, alle Möglichkeiten zu untersuchen, um aus 20 linear unabhängigen Zeilen von R eine Matrix $C_k \in \mathbb{R}^{20 \times 20}$ zu bilden und das System

$$C_k g_k = -(D_{-1,k})_q,$$

zu lösen. Der Vektor $(D_{-1,k})_q \in \mathbb{R}^{20}$ ist hierbei ein Untervektor von $D_{-1,k}$, den man erhält, indem man die 20 Komponenten von $D_{-1,k}$ übernimmt, die den 20 Zeilen von R entsprechen, die C_k erzeugen. Dann enthält der Vektor

$$D_{-1,k} + R g_k = D_{-1,k} - R C_k^{-1} (D_{-1,k})_q$$

wenigstens 20 Nullen. Alle auf diese Weise berechneten Vektoren $D_{-1,k} + R g_k$ müsste man bezüglich ihrer l^0 -Seminorm miteinander vergleichen. Natürlich ist ein solches Vorgehen höchst ineffizient für unsere Zwecke. Wir möchten mit einem einfachen Algorithmus versuchen, a priori zu entscheiden, welche 20 Einträge sinnvoller Weise zu Null „gedrückt“ werden sollten.

Unser Algorithmus basiert auf einer einfachen Idee: Betrachte die lokalen, orthogonalen Projektionen des Bildes f_0 in die Unterräume $V_{-1}^\nu \oplus W_{-1}^\nu$, $\nu = 0, 1, 2, 3$, die die vier verschiedenen Hauptrichtungen repräsentieren, siehe Abbildung 2.8. Wir erinnern daran, dass für ein festes $k = (k_1, k_2)^T \in I_2$, und für jedes $\nu \in \{0, 1, 2, 3\}$, die erzeugenden Funktionen in $\Psi_{-1}^\nu(\cdot - 2k)$ eine Orthogonalbasis von $V_{-1}^\nu \oplus W_{-1}^\nu$ bilden (eingeschränkt auf $Q_k := [2k_1, 2k_1 + 2) \times [2k_2, 2k_2 + 2)$). Wir betrachten die 64 Koeffizienten, die durch die vier Projektionen entstehen, und wählen zunächst die betragsmäßig kleinsten Koeffizienten aus. Die Framenfunktionen, die zu diesen Koeffizienten gehören, sind unwesentlich für die Darstellung von f , so dass wir durch geeignete Wahl von g_k diese Koeffizienten gleich Null setzen können.

Bei diesem Vorgehen soll – wie bereits erwähnt – von unserem expliziten Wissen über das redundante Dictionary Gebrauch gemacht werden; insbesondere wollen wir die Abhängigkeitsbeziehungen des Systems ausnutzen, die in der Matrix U zusammengefasst sind.

Der folgende detailliert beschriebene Algorithmus liefert eine effiziente, lokale Darstellung eines Bildes. Wir werden später zeigen, dass diese neue Methode optimal für stückweise konstante Bilder ist und nicht von der Darstellung des Startbildes abhängt.

Algorithmus 2: Dünne, lokale Darstellung in $V_{-1} + W_{-1}$

Input: $D_{0,k} \in \mathbb{R}^{64}$ wie in Algorithmus 1b, Schritt 2, so dass $f_0|_{Q_k} = D_{0,k}^T \Phi_0(\cdot - 2k)$, und $R := (I_4 \otimes A)U^T \in \mathbb{R}^{64 \times 20}$.

1. Berechne die lokalen orthogonalen Projektionen von $f_0|_{Q_k} = D_{0,k}^T \Phi_0(\cdot - 2k)$ in die Unterräume $V_{-1}^\nu \oplus W_{-1}^\nu$, $\nu = 0, 1, 2, 3$.

Starte mit dem Ansatz $f_{-1}^\nu := (h_k^\nu)^T \Psi_{-1}^\nu(\cdot - 2k)$ in Q_k für die lokale, orthogonale Projektion von f_0 in $V_{-1}^\nu \oplus W_{-1}^\nu$. Die Orthogonalität der Basisfunktionen in Ψ_{-1}^ν führt zu

$$h_k^\nu := 4 \langle \Psi_{-1}^\nu(\cdot - 2k), f_0 \rangle \in \mathbb{R}^{16}.$$

Mit dem Vektor $h_k := ((h_k^0)^T, (h_k^1)^T, (h_k^2)^T, (h_k^3)^T)^T \in \mathbb{R}^{64}$ erhält man

$$h_k = 4 \langle \Psi_{-1}(\cdot - 2k), f_0 \rangle = 4 \langle (I_4 \otimes A) \Phi_0(\cdot - 2k), D_{0,k}^T \Phi_0(\cdot - 2k) \rangle = 4(I_4 \otimes A) \langle \Phi_0, \Phi_0 \rangle D_{0,k}.$$

Sei wiederum P die Permutationsmatrix (wie in Algorithmus 1a) zum Umordnen der erzeugenden \mathcal{AB} -Skalierungsfunktionen in V_0 , d. h.

$$(\Phi^T, \Phi(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}))^T, \Phi(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix})^T, \Phi(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix})^T) P = \Phi_0^T.$$

Dann ergibt sich mit der in (2.17) definierten Gram'schen Matrix G von Φ die Gleichung $\langle \Phi_0, \Phi_0 \rangle = P^T (I_4 \otimes G) P$. Also

$$h_k = 4(I_4 \otimes A)P^T(I_4 \otimes G)PD_{0,k}.$$

Der Vektor h_k enthält nun alle lokalen Koeffizienten der vier orthogonalen Projektionen von f_0 in $V_{-1}^v \oplus W_{-1}^v$, $v = 0, 1, 2, 3$.

2. Ordne die Komponenten von $h_k \in \mathbb{R}^{64}$ vom kleinsten Absolutbetrag bis hin zum größten, und berechne die entsprechende Permutation (p_1, \dots, p_{64}) der Indizes $(1, \dots, 64)$. Wenn gewisse Werte in h_k denselben Absolutbetrag haben, so nehme den mit dem kleinsten Index zuerst.
3. Berechne eine invertierbare Matrix $C_k \in \mathbb{R}^{20 \times 20}$, indem 20 Zeilen von R nach folgendem Schema ausgewählt werden:

(a) Die erste Zeile von C_k ist die p_1 -te Zeile von R .

(b) Die zweite Zeile von C_k ist die p_2 -te Zeile von R , sofern sie linear unabhängig von der p_1 -ten Zeile von R ist. Andernfalls, betrachte die p_3 -te Zeile von R etc. Allgemein verfähre für $i = 1, 2, \dots$ wie folgt: Wenn die p_i -te Zeile von R linear unabhängig von den bisher gewählten Zeilen ist, übernehme diese Zeile als nächste Zeile von C_k . Andernfalls gehe weiter zur p_{i+1} -ten Zeile.

(c) Beende diese Prozedur, wenn 20 linear unabhängige Zeilen von R übernommen wurden und C_k auf diese Weise vollständig berechnet wurde. Die Eigenschaft $\text{Rang}(R) = 20$ garantiert ein Ende dieser Prozedur.

4. Sei $q = (q_1, \dots, q_{20})$ der Vektor der Indizes der Zeilen von R , die in C_k übernommen wurden. Löse das lineare System

$$C_k g_k = -(D_{-1,k})_q,$$

wobei $(D_{-1,k})_q$ die Komponenten aus $D_{-1,k} = (I_4 \otimes A)D_{0,k}$ mit den Indizes q_1, \dots, q_{20} enthält, und zwar in dieser Reihenfolge. Mit dem Ergebnisvektor g_k sind wir in der Lage, den gewünschten dünnbesetzten Koeffizientenvektor $\check{D}_{-1,k}^T = D_{-1,k}^T + g_k^T R^T$ zu berechnen.

Output: Neue dünne, lokale Darstellung $\check{D}_{-1,k}^T \Psi_{-1}$ von $f|_{Q_k}$ in $V_{-1} + W_{-1}$.

Der vierte Schritt des Algorithmus bewirkt, dass in der neuen Darstellung von $f_0 \in V_0$, die durch $\check{D}_{-1,k}^T$ bestimmt ist, mindestens 20 Waveletkoeffizienten verschwinden, und zwar die, die zu den Indizes (q_1, \dots, q_{20}) gehören.

Abschließend zeigen wir zwei wesentliche Eigenschaften des vorgeschlagenen Algorithmus.

Satz 2.8. *Die lokale, dünne Darstellung von $f_0 \in V_0$ in $V_{-1} + W_{-1}$, die man mithilfe des Algorithmus 2 erhält, ist eindeutig, d. h., sie hängt nicht von der ursprünglichen redundanten Darstellung von f_0 in V_0 ab.*

Beweis. Die Komponenten des Funktionen-Vektors Ψ_{-1}^ν bilden eine Basis von $V_{-1}^\nu \oplus W_{-1}^\nu$ für jedes $\nu = 0, 1, 2, 3$. Deshalb sind die lokalen Projektionen von f_0 in $V_{-1}^\nu \oplus W_{-1}^\nu$ eindeutig darstellbar und hängen nicht von der Start-Darstellung von f_0 in V_0 ab. Das bedeutet, dass auch die Matrix C_k , die im Schritt 3 von Algorithmus 2 berechnet wird, eindeutig bestimmt ist. Unter Verwendung des Parameter-Vektors $g_k = -C_k^{-1}(D_{-1,k})_q$ wie in Schritt 4 von Algorithmus 2 berechnet, erhält man aus $f_0 = D_{-1,k}^T \Psi_{-1}$ die neue lokale Darstellung von f_0 in $V_{-1} + W_{-1}$,

$$f_0 = \left[D_{-1,k}^T - (D_{-1,k})_q^T (C_k^{-1})^T R^T \right] \Psi_{-1},$$

die nach Konstruktion 20 Null-Koeffizienten korrespondierend zu den Indizes (q_1, \dots, q_{20}) besitzt. D. h. die Komponenten $\psi_{q_1}, \dots, \psi_{q_{20}}$ in dem Funktionen-Vektor $\Psi_{-1} = (\psi_\mu)_{\mu=1}^{64}$ werden nicht mehr für die Darstellung von f_0 benötigt. Sei nun $f_0 = \hat{D}_{-1,k}^T \Psi_{-1}$ eine zweite Darstellung von f_0 in $V_{-1} + W_{-1}$, dann existiert ein Vektor $b \in \mathbb{R}^{20}$ mit $\hat{D}_{-1,k}^T = D_{-1,k}^T + b^T R^T$. Algorithmus 2 liefert die Darstellung

$$\begin{aligned} f_0 &= \left[\hat{D}_{-1,k}^T - (\hat{D}_{-1,k})_q^T (C_k^{-1})^T R^T \right] \Psi_{-1} \\ &= \left[D_{-1,k}^T + b^T R^T - [(D_{-1,k})_q^T + b^T R_q^T] (C_k^{-1})^T R^T \right] \Psi_{-1} \\ &= \left[D_{-1,k}^T - (D_{-1,k})_q^T (C_k^{-1})^T R^T \right] \Psi_{-1}, \end{aligned}$$

wegen $b^T R_q^T (C_k^{-1})^T R^T = b^T C_k^T (C_k^{-1})^T R^T = b^T R^T$ nach Konstruktion. Hier bezeichnet $b^T R_q^T \in \mathbb{R}^{20}$ den Teilvektor von $b^T R^T \in \mathbb{R}^{64}$ bestehend aus den Einträgen (q_1, \dots, q_{20}) . \square

Satz 2.9. *Sei f konstant auf dem Bildquadrat $Q_k = [2k_1, 2k_1 + 2) \times [2k_2, 2k_2 + 2)$ für $k = (k_1, k_2)^T \in I_2$. Dann liefert Algorithmus 2 eine optimale Darstellung von f in $V_{-1} + W_{-1}$.*

Beweis. Sei $f \equiv c$ auf Q_k mit einem gewissen $c \in \mathbb{R}$. Dann kann f in V_0 dargestellt werden durch

$$f = c \cdot (1_{16}^T, 0_{16}^T, 0_{16}^T, 0_{16}^T) \Phi_0.$$

Nach Satz 2.8 können wir unsere Untersuchung auf diese Darstellung von f reduzieren. Nun wenden wir Algorithmus 2 an und zeigen, dass der resultierende Koeffizientenvektor eine minimale Anzahl von Nichtnull-Elementen enthält. Wenn wir den ersten Schritt von Algorithmus 2 ausführen und die orthogonalen Projektionen von f in $V_{-1}^\nu \oplus W_{-1}^\nu$, $\nu = 0, 1, 2, 3$, berechnen, erhalten wir den Vektor $h_k = c(w^T, w^T, w^T, w^T)$ mit $w^T \in \mathbb{R}^{16}$ aus (2.34). Man

kann sich schnell klar machen, dass im zweiten Schritt des Algorithmus der Vektor h_k die Permutation

$$p = (2, 3, 4, 6, 7, 8, 10, 11, 12, \dots, 62, 63, 64, 1, 5, 9, \dots, 57, 61)$$

liefert. O.B.d.A. seien die Zeilen von U durch die Abhängigkeitsgleichungen auf Seite 55 gegeben, und zwar auch in der dort aufgeführten Reihenfolge. Eine einfache Rechnung gemäß Schritt 3 des Algorithmus 2 zeigt, dass die 48 Zeilen mit den Indizes $(2, 3, 4, \dots, 62, 63, 64)$ von $R = (I_4 \otimes A)U^T$ genau 15 linear unabhängige Zeilen enthalten. Diese werden für die Berechnung der ersten 15 Zeilen von C_k benutzt. Die letzten 5 Zeilen von C_k kommen aus den Zeilen aus R mit den Indizes $(1, 9, 17, 33, 41)$.

Schließlich wenden wir den vierten Schritt von Algorithmus 2 an, und erhalten mit

$$D_{-1,k} = c(I_4 \otimes A)(1_{16}^T, 0_{16}^T, 0_{16}^T, 0_{16}^T) = c(w^T, 0_{16}^T, 0_{16}^T, 0_{16}^T)$$

den Vektor $(D_{-1,k})_q = 2c(e_{16} + e_{17})$, weil nur die Komponenten in $D_{-1,k}$ mit den Indizes $q_{16} = 1$ und $q_{17} = 9$, die die 16te und 17te Zeile von C_k angeben, Nichtnull-Werte sind. Hier bezeichnen wir mit $e_\mu := (\delta_{\mu,v})_{v=1}^{20}$ den μ -ten Einheitsvektor der Länge 20. Mit

$$w_1^T := (2, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \in \mathbb{R}^{16}, \quad w_2^T := w^T - w_1^T,$$

haben die 16te und 17te Zeile der Matrix $(R C_k^{-1})^T$ die Form $\frac{1}{2}(w_1^T, -w_2^T, 0_{16}^T, 0_{16}^T)$ und $\frac{1}{2}(w_2^T, w_2^T, 0_{16}^T, -w^T)$. Man erhält schließlich

$$\begin{aligned} f &= \left[D_{-1,k}^T - (D_{-1,k})_q^T (C_k^{-1})^T R^T \right] \Psi_{-1} \\ &= c \cdot \left[(w^T, 0_{16}^T, 0_{16}^T, 0_{16}^T) - (w_1^T, -w_2^T, 0_{16}^T, 0_{16}^T) - (w_2^T, w_2^T, 0_{16}^T, -w^T) \right] \Psi_{-1} \\ &= c \cdot (0_{16}^T, 0_{16}^T, 0_{16}^T, w^T) \Psi_{-1}. \end{aligned}$$

Dies ist eine optimale dünne Darstellung von f mit nur vier Nichtnull-Koeffizienten. \square

2.2.5 Anwendung in der Bildverarbeitung

In diesem Abschnitt wenden wir die hergeleiteten Algorithmen zur Entstörung und Approximation von digitalen Bildern an. Beide Anwendungen basieren auf der effizienten Multiskalen-Zerlegung mittels der entwickelten Filterbank. Als Testbilder dienen uns wieder das stückweise konstante Bild mit geometrischen Figuren und das „pepper“-Bild aus der Klasse der natürlichen Bilder. Wir vergleichen unsere Ergebnisse mit der Curvelet- und Contourlet-Methode. Zu diesem Zweck haben wir die MATLAB-Toolboxen von den Internetseiten www.curvelet.org und www.ifp.uiuc.edu/~minhdo/software/ genutzt.

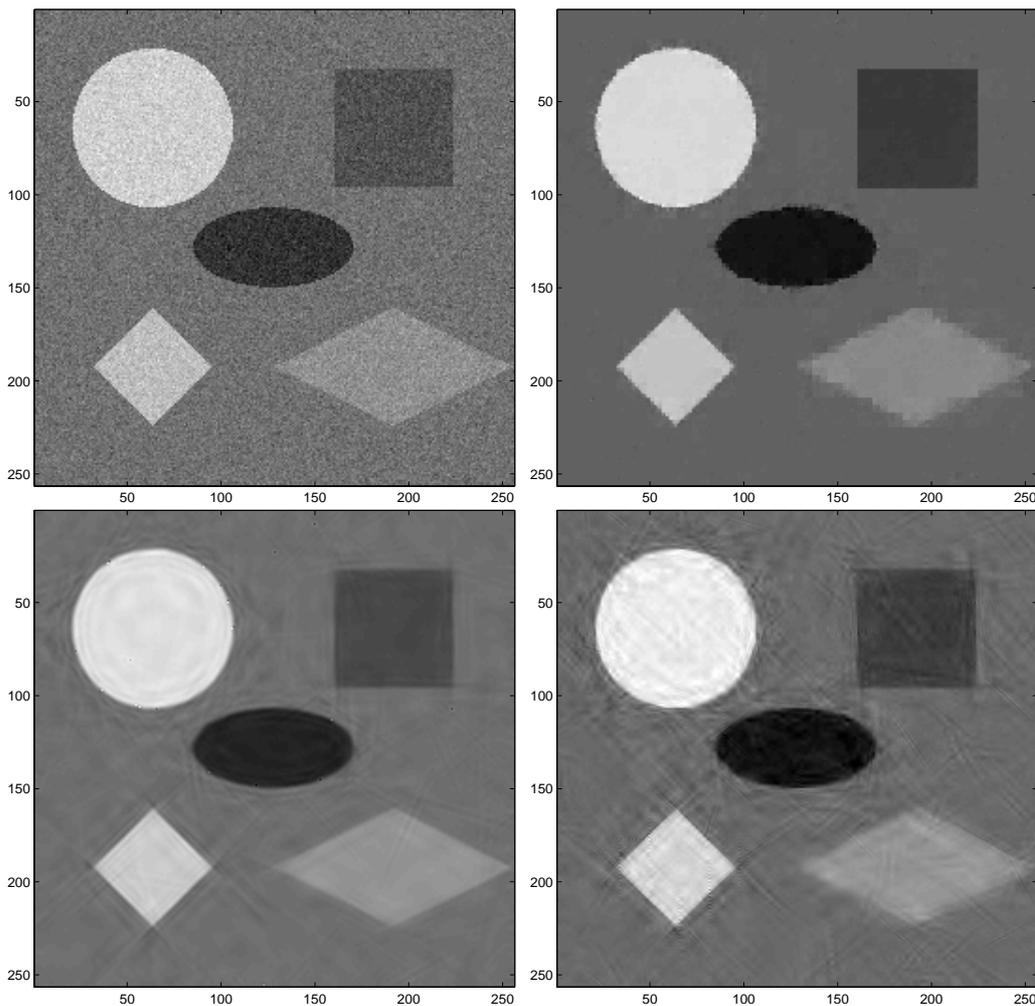


Abbildung 2.11: Entstörung eines synthetischen Bildes. (a) Gestört, PSNR 22.08 dB, (b) Unsere Methode, PSNR 33.12 dB, (c) Curvelets, PSNR 32.18 dB, (d) Contourlets, PSNR 29.91 dB.

Bildentstörung Die beiden Testbilder der Größe 256×256 in Abbildung 2.11 und 2.12 sind mit einem Gauß'schen Rauschen der Stärke $\sigma = 20$ gestört. Nach der Anwendung des Zerlegungsalgorithmus 1a werden die Waveletkoeffizienten einem globalen Hard-Thresholding mit dem Schwellwert $\lambda = \sigma\sqrt{\log(N)/2}$ unterzogen. Dabei gibt N die Anzahl der Pixel des Bildes an. Auf diese Weise erhalten wir ein gutes Entstörungsergebnis im synthetischen Bild 2.11, weil die verschiedenen gerichteten Bildkanten gut detektiert werden. Selbst für natürliche Bilder wie das „pepper“-Bild liefern die gerichteten Haarwavelets akzeptable Resultate, wenn auch keine exzellenten. Immerhin wird in Abbildung 2.12 deutlich, dass der PSNR-Wert unserer Methode in der gleichen Größenordnung wie bei der Curvelet-Methode liegt und die Contourlet-Methode sogar übertrifft.

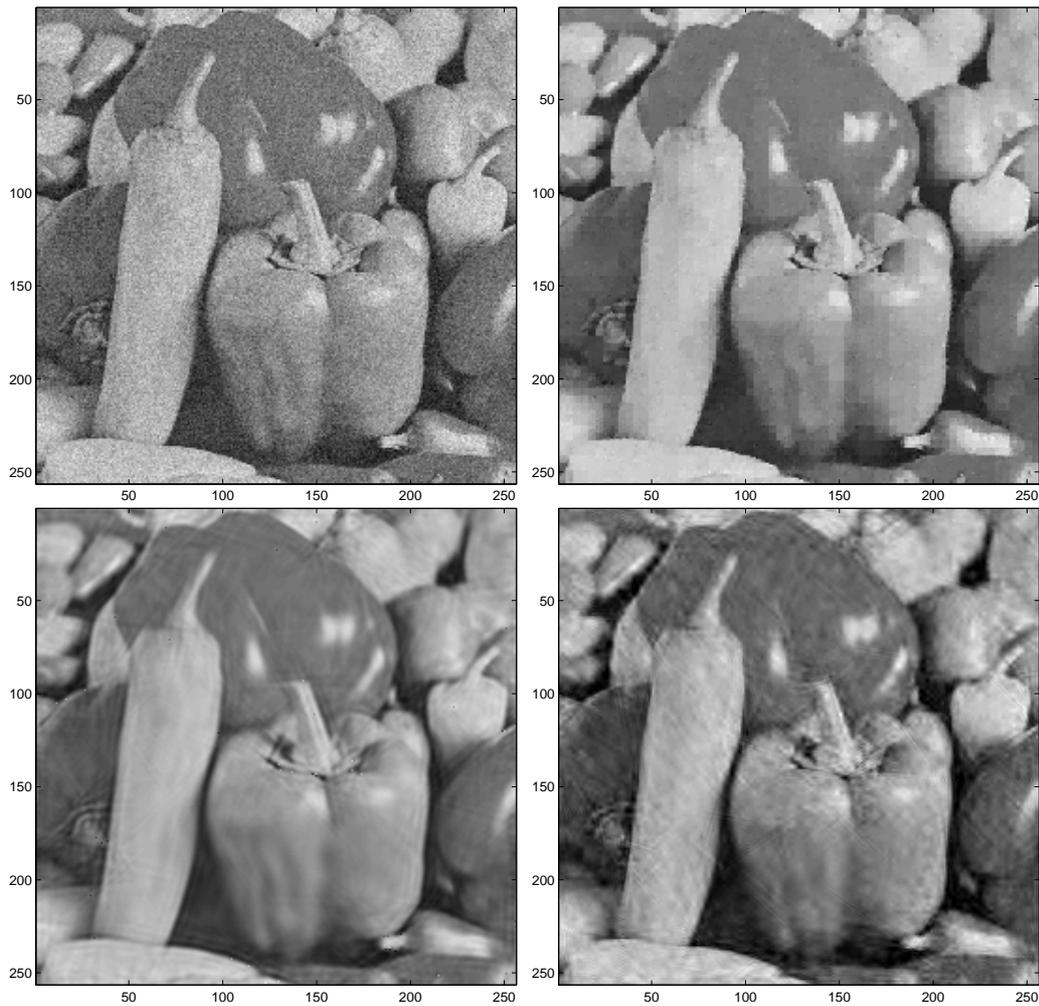


Abbildung 2.12: Entstörung des „pepper“-Bildes. (a) Gestört, PSNR 22.08 dB, (b) Unsere Methode, PSNR 28.00 dB, (c) Curvelets, PSNR 28.14 dB, (d) Contourlets, PSNR 26.16 dB.

Bildapproximation Während bei der Bildentstörung die vierfache Redundanz unseres Waveletframes hilfreich war, reduzieren wir bei der Bildapproximation die vorherrschenden Redundanzen mithilfe des Algorithmus 1b. Auf diese Weise können wir die Redundanz von 4 auf 2,74 verringern. Konkret bedeutet dies z. B. für das „pepper“-Bild in Abbildung 2.14, dass wir nach der Zerlegung 179.780 Koeffizienten haben. (Das Ausgangsbild hat $256^2 = 65.536$ Pixelwerte). Die gleiche Größenordnung beobachten wir auch bei der diskreten Curvelet-Transformation: Dort erhält man 185.344 Koeffizienten, was einer Redundanz von 2,83 entspricht. Die Contourlet-Transformation erreicht eine bemerkenswert geringe Redundanz von 1,3 (das sind 86.016 Koeffizienten) und liefert damit auch die beste Approximation für unsere Testbilder. In Abbildung 2.13 rekonstruieren wir das synthetische Bild aus den 1.311 betragsmäßig größten Koeffizienten; das „pepper“-Bild in Abbildung 2.14 wird mit 6.554 Koeffizienten dargestellt.

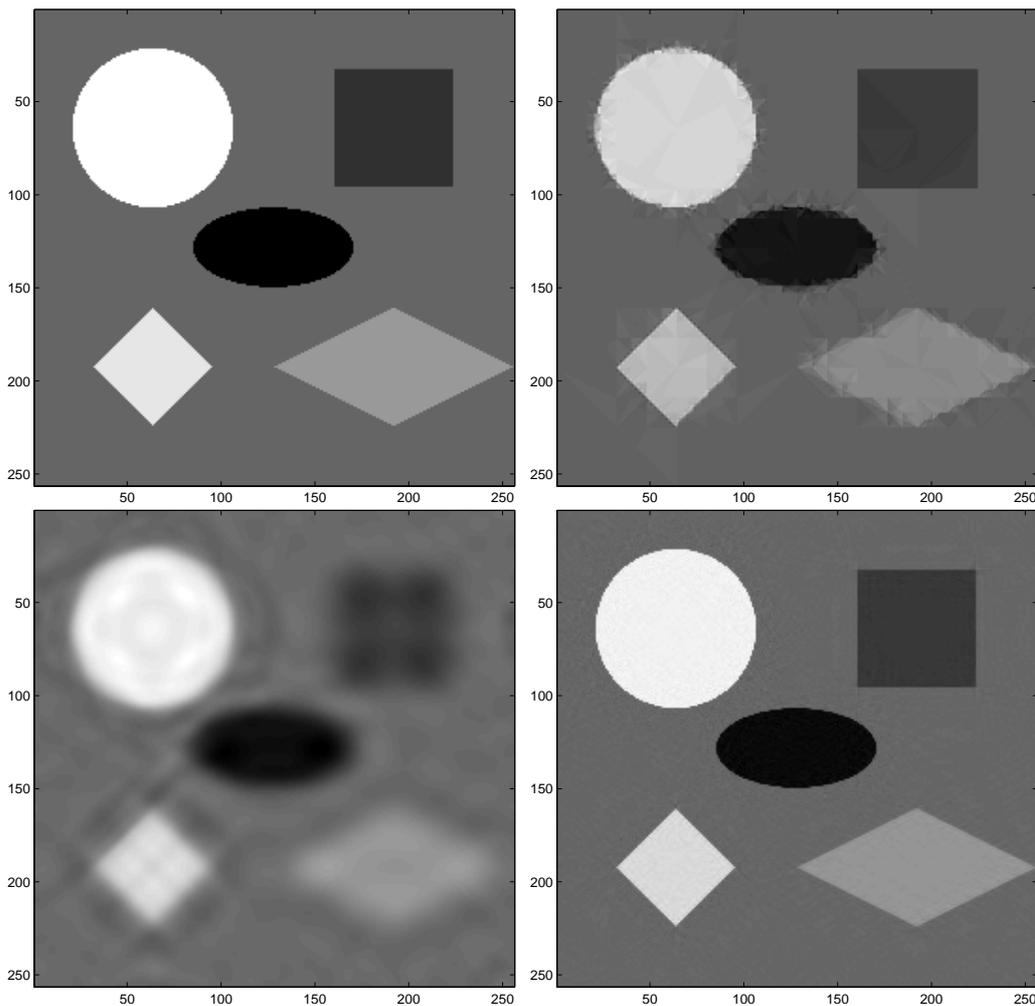


Abbildung 2.13: Approximation eines synthetischen Bildes mit 2% der ursprünglichen Koeffizienten. (a) Originalbild, (b) Unsere Methode, PSNR 31.82 dB, (c) Curvelets, PSNR 26.77 dB, (d) Contourlets, PSNR 34.13 dB.

Insbesondere bei dem „pepper“-Bild werden die Dreiecksstrukturen der verwendeten \mathcal{AB} -Wavelets deutlich. Diese Artefakte könnten mit einem geeigneten Postprocessing-Schritt behoben werden, vergleiche Kapitel 4.

2.3 Charakterisierung von Tight-Frames mittels der Polyphasen-Matrix

Das im vorhergehenden Abschnitt konstruierte Waveletsystem kann als vektorwertiger Parseval-Frame verstanden werden, wenn wir die 16 Mutterwavelets in einen Funktionsvektor schreiben. Das motiviert uns, die Frame-Untersuchungen für vektorwertige Wavelets (Multiwavelets) vorzunehmen, deren Notation wir schon auf Seite 17 eingeführt haben.

In diesem Unterkapitel zeigen wir, dass die Unitarität der Polyphasen-Matrix ein Kri-

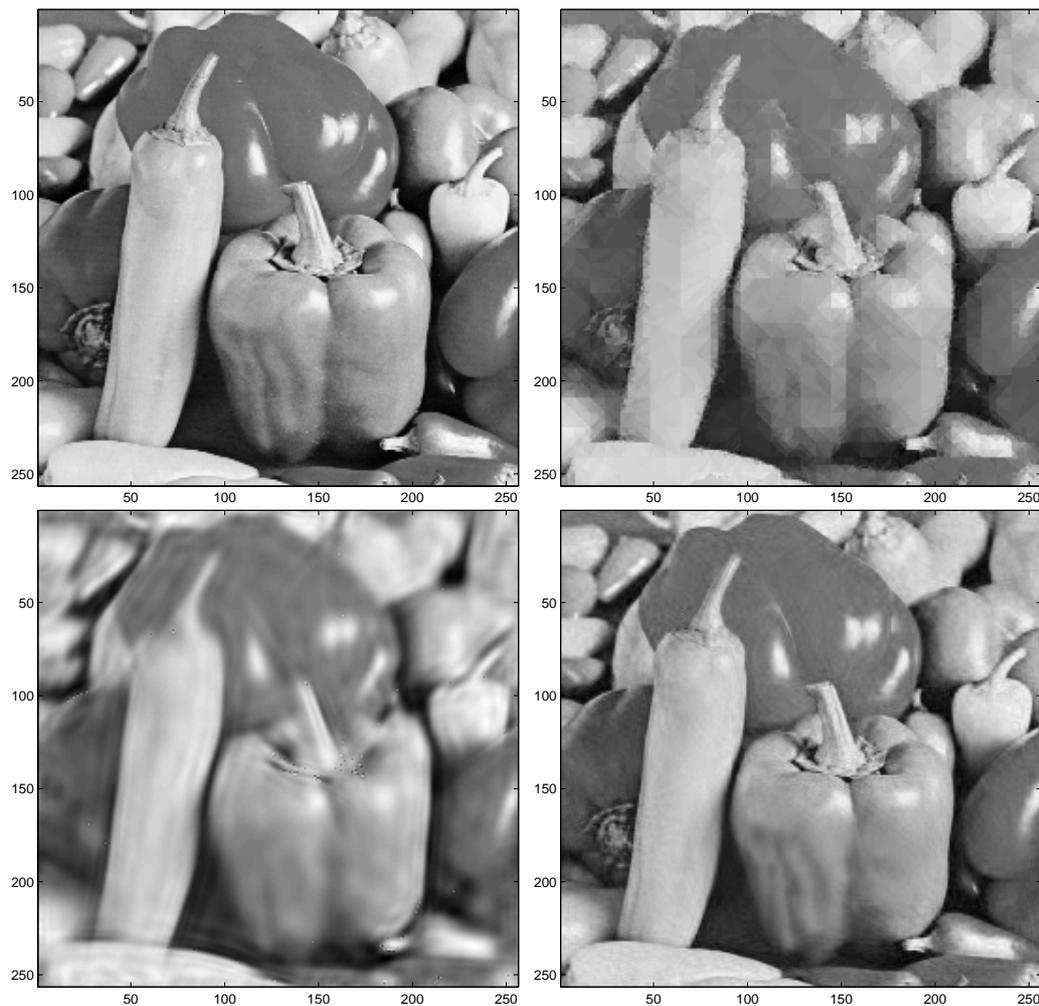


Abbildung 2.14: Approximation des „pepper“-Bildes mit 10% der ursprünglichen Koeffizienten. (a) Originalbild, (b) Unsere Methode, PSNR 27.80 dB, (c) Curvelets, PSNR 25.51 dB, (d) Contourlets, PSNR 32.12 dB.

terium ist, um die Tight-Frame-Eigenschaft eines vektorwertigen Waveletsystems im Frequenzbereich zu überprüfen. Diese Aussage ist eine Variante der *Extension Principles*, die wir in Kapitel 1.2 beleuchtet hatten. Nach einer kurzen Erläuterung der *Extension Principles* im vektorwertigen Fall beweisen wir die Hauptaussage des Kapitels und erläutern dann den engen Zusammenhang zwischen der Modulations- und Polyphasen-Darstellung. Ganz am Ende benutzen wir das Resultat, um auch im Frequenzbereich zu zeigen, dass die im vorhergehenden Abschnitt betrachteten, gerichteten Haarwavelets auf Dreiecken in der Tat einen Parseval-Frame bilden.

2.3.1 Die Extension Principles

Die üblichen Methoden zur Konstruktion und Charakterisierung von Waveletframes sind die *Extension Principles*. Wie bereits oben erwähnt, sind diese Ansätze Verallgemeinerungen der

klassischen Aussagen, dass die ONB-Eigenschaft eines Waveletsystems äquivalent ist zur Unitarität der entsprechenden Modulationsmatrix. Im Abschnitt 1.2 hatten wir die Extension Principles für skalarwertige Wavelets im multivariaten Fall formuliert.

Das Zusammenspiel der Symbole lässt sich auch auf den vektorwertigen Fall übertragen. Zunächst betrachten wir wieder den klassischen Fall einer MRA, bei der die Skalierungsfunktionen eine ONB des V_0 generieren. Hier gilt folgende Charakterisierung des Waveletsystems im Frequenzbereich.

Satz 2.10 (Multiwavelets im multivariaten Fall). *Der Funktionenvektor $\Phi = (\phi_0, \dots, \phi_{r-1}) \in (L^2(\mathbb{R}^d))^r$ generiere eine ONB-MRA für $L^2(\mathbb{R}^d)$, und Φ erfülle eine Verfeinerungsgleichung*

$$\hat{\Phi}(\omega) = H^0(\omega A^{-1}) \hat{\Phi}(\omega A^{-1}), \quad \omega \in \hat{\mathbb{R}}^d,$$

für eine $r \times r$ -Matrix H^0 , deren Einträge 2π -periodische Polynome sind. Dann gilt:

Die durch $\hat{\Psi}^l(\omega) = H^l(\omega A^{-1}) \hat{\Phi}(\omega A^{-1})$ definierten Multiwavelets generieren eine ONB des $L^2(\mathbb{R}^d)$ genau dann, wenn

$$\mathcal{M}(\omega) = [H^l(\omega + 2\pi\gamma_j^T A^{-1})]_{l,j=0}^{|\det A|-1}$$

unitär ist, wobei die Matrizen H^l , $l = 1, \dots, |\det A| - 1$, 2π -periodisch sind.

Dieser Satz ist die Verallgemeinerung von Satz 1.9 im vektorwertigen Fall, er kann z. B. in [CHM04] gefunden werden. In den letzten Jahren sind auch die Extension Principles auf den Fall von Multiwavelets übertragen worden. Diese Übertragung liegt nicht unmittelbar auf der Hand, weil eine Faktorisierung der Symbole, wie man sie von den skalaren Funktionen kennt, für matrixwertige Symbole schwierig ist. In [Han08] ist das allgemeinste der vier Extension Principles, das MOEP, für vektorwertige Wavelets im multivariaten Fall bewiesen worden.

Wir werden in diesem Kapitel das UEP und das OEP für Multiwavelets im multivariaten Fall mit Hilfe der *Polyphasen-Matrix* formulieren, dabei stützen wir uns auf unsere Arbeit [Kro09b].

Die Betrachtung der Polyphasen-Matrix $\mathcal{P}(\omega)$ aus (1.15) hat sowohl in der Filterbank-Theorie als auch in der Waveletanalyse gewisse Vorteile gegenüber der Verwendung der Modulationsmatrix $\mathcal{M}(\omega)$. Wie bereits in Kapitel 1.3 erwähnt, liefert die Polyphasen-Darstellung unmittelbar eine effiziente Filterbank, indem das Eingangssignal in zwei Phasen zerlegt wird und die Wavelettransformation auf zwei Signale halber Länge angewendet werden kann (für $d = 1$ und $A = 2$). In der Waveletanalyse hat die Polyphasen-Matrix eine große Bedeutung durch das Lifting-Schema bei vektorwertigen Wavelets bekommen. Man stellt nämlich fest, dass die Modulationsmatrix eine besondere Struktur hat: Die gesamte Information ist bereits in der ersten Spalte enthalten, die anderen Spalten erhält man aus der ersten Spalte durch Translation der trigonometrischen Polynome. Wenn wir eine neue Matrix, deren Einträge trigonometrische Polynome sind, mit der Modulationsmatrix multiplizieren, um neue Wavelets

über diese neue Modulationsmatrix zu konstruieren, so muss diese Matrix eine bestimmte Struktur aufweisen. Im Gegensatz dazu, ist dies bei der Polyphasen-Matrix nicht der Fall. Sie ist unstrukturiert, und damit erhält man mehr Freiheit, einen Matrix-Faktor zu finden, mit dem die neuen Wavelets erzeugt werden können. Wir verweisen an dieser Stelle auf das Buch von Keinert [Kei04] über vektorwertige Wavelets. In Kapitel 9 werden dort die verschiedenen Faktorisierungs-Methoden der Symbole vorgestellt, um im Frequenzbereich neue Wavelets mit gewünschten Eigenschaften zu generieren. Außerdem findet man dort eine Fülle von weiterführenden Literaturangaben.

2.3.2 Die Polyphasen-Matrix und die Tight-Frame-Eigenschaft

Der folgende Satz, das Hauptresultat des gesamten Unterkapitels 2.3, zeigt, dass die Unitarität der Polyphasen-Matrix die Tight-Frame-Eigenschaft eines Multiwaveletsystems garantiert.

Satz 2.11 (UEP in der Polyphasen-Darstellung). *Sei Φ ein Skalierungsfunktionen-Vektor, der eine Matrix-Verfeinerungsgleichung $\hat{\Phi}(\omega) = H^0(\omega A^{-1}) \hat{\Phi}(\omega A^{-1})$ erfüllt. Weiter soll $\|\hat{\Phi}(0)\|_2^2 = 1$ und $\lim_{j \rightarrow \infty} \|\hat{\Phi}(\omega A^j)\|_2^2 = 0$ gelten.*

Dann gilt: Wenn die Polyphasen-Matrix $\mathcal{P}(\omega)$ unitär ist für fast alle $\omega \in \hat{\mathbb{R}}^d$, so erzeugen die Multiwavelets

$$\{|\det A|^{j/2} \psi_i^l(A^j \cdot -k) : l = 1, \dots, L-1; i = 0, \dots, r-1; j \in \mathbb{Z}; k \in \mathbb{Z}^d\},$$

die mittels $\hat{\Psi}^l(\omega) = H^l(\omega A^{-1}) \hat{\Phi}(\omega A^{-1})$ definiert sind, einen Tight-Frame des $L^2(\mathbb{R}^d)$, d. h. es gibt eine Konstante $C > 0$ mit

$$C \|f\|_2^2 = \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^d} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\langle f, |\det A|^{j/2} \psi_i^l(A^j \cdot -k) \rangle|^2 \quad \forall f \in L^2(\mathbb{R}^d).$$

Beweis. Wir verwenden die Parseval'sche Gleichung $\langle f, g \rangle = \frac{1}{(2\pi)^{d/2}} \langle \hat{f}, \hat{g} \rangle$, wobei $\langle \cdot, \cdot \rangle$ das übliche Skalarprodukt in $L^2(\mathbb{R}^d)$ ist, und erhalten damit für die rechte Seite

$$\begin{aligned} & \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^d} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\langle f, |\det A|^{j/2} \psi_i^l(A^j \cdot -k) \rangle|^2 \\ &= \sum_{j,k,l,i} \left| \frac{1}{(2\pi)^{d/2}} \langle \hat{f}, |\det A|^{-j/2} \hat{\psi}_i^l(\cdot A^{-j}) e^{-i \cdot A^{-j} k} \rangle \right|^2 \\ &= \frac{1}{(2\pi)^d} \sum_{j,l,i} |\det A|^j \sum_{k \in \mathbb{Z}^d} |\langle \hat{f}(\cdot A^j) \hat{\psi}_i^l, e^{-i \cdot k} \rangle|^2 \\ &= \frac{1}{(2\pi)^d} \sum_{j,l,i} |\det A|^j \int_{\hat{\mathbb{R}}^d} |\hat{f}(\omega A^j)|^2 |\hat{\psi}_i^l(\omega)|^2 d\omega. \quad (2.35) \end{aligned}$$

Im letzten Schritt haben wir wieder die Parseval'sche Gleichung in der Gestalt

$$\sum_{k \in \mathbb{Z}^d} |\langle \hat{g}, e^{-i \cdot k} \rangle|^2 = \|\hat{g}\|_2^2 = \int_{\hat{\mathbb{R}}^d} |\hat{g}(\omega)|^2 d\omega, \quad \forall g \in L^2(\mathbb{R}^d),$$

benutzt. Im Folgenden betrachten wir in (2.35) zunächst die Summe $\sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\hat{\psi}_i^l(\omega)|^2$. Mit der Vektor-Schreibweise $\|\hat{\Psi}^l(\omega)\|_2^2 = \sum_{i=0}^{r-1} |\hat{\psi}_i^l(\omega)|^2$ bekommen wir den Ausdruck

$$\begin{aligned} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\hat{\psi}_i^l(\omega)|^2 &= \sum_{l=1}^{L-1} \|\hat{\Psi}^l(\omega)\|_2^2 = \sum_{l=1}^{L-1} \|H^l(\omega A^{-1}) \hat{\Phi}(\omega A^{-1})\|_2^2 \\ &= \sum_{l=1}^{L-1} \overline{\hat{\Phi}(\omega A^{-1})}^T \overline{H^l(\omega A^{-1})}^T H^l(\omega A^{-1}) \hat{\Phi}(\omega A^{-1}). \end{aligned}$$

Jetzt machen wir Gebrauch von der Zerlegung des Symbols $H^l(\omega A^{-1})$ in (1.14) in seine m Polyphasen-Komponenten, wobei $\Gamma = \{\gamma_0, \dots, \gamma_{m-1}\}$ mit $m = |\det A|$ die Repräsentantenmenge des Gitters \mathbb{Z}^2 war.

$$\begin{aligned} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\hat{\psi}_i^l(\omega)|^2 &= \overline{\hat{\Phi}(\omega A^{-1})}^T \sum_{l=1}^{L-1} \overline{\left(\frac{1}{|\det A|^{1/2}} \sum_{\gamma \in \Gamma} e^{-i\omega A^{-1} \gamma} H_{\gamma}^l(\omega) \right)}^T \\ &\quad \left(\frac{1}{|\det A|^{1/2}} \sum_{\gamma' \in \Gamma} e^{-i\omega A^{-1} \gamma'} H_{\gamma'}^l(\omega) \right) \hat{\Phi}(\omega A^{-1}) \\ &= \overline{\hat{\Phi}(\omega A^{-1})}^T \frac{1}{|\det A|} \sum_{\gamma, \gamma' \in \Gamma} e^{-i\omega A^{-1}(\gamma' - \gamma)} \left(\sum_{l=1}^{L-1} \overline{H_{\gamma}^l(\omega)}^T H_{\gamma'}^l(\omega) \right) \hat{\Phi}(\omega A^{-1}). \end{aligned}$$

Wegen der vorausgesetzten Unitarität der Polyphasen-Matrix wird dieser Ausdruck mit Hilfe von (1.17) zu

$$\begin{aligned} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\hat{\psi}_i^l(\omega)|^2 &= \overline{\hat{\Phi}(\omega A^{-1})}^T \frac{1}{|\det A|} \sum_{\gamma, \gamma' \in \Gamma} e^{-i\omega A^{-1}(\gamma' - \gamma)} \\ &\quad \cdot \left(\delta_{\gamma, \gamma'} I_{r \times r} - \overline{H_{\gamma}^0(\omega)}^T H_{\gamma'}^0(\omega) \right) \hat{\Phi}(\omega A^{-1}) \\ &= \overline{\hat{\Phi}(\omega A^{-1})}^T \frac{m}{|\det A|} \hat{\Phi}(\omega A^{-1}) \\ &\quad - \overline{\hat{\Phi}(\omega A^{-1})}^T \frac{1}{|\det A|} \sum_{\gamma, \gamma' \in \Gamma} e^{-i\omega A^{-1}(\gamma' - \gamma)} \overline{H_{\gamma}^0(\omega)}^T H_{\gamma'}^0(\omega) \hat{\Phi}(\omega A^{-1}) \\ &= \|\hat{\Phi}(\omega A^{-1})\|_2^2 - \overline{\hat{\Phi}(\omega A^{-1})}^T \overline{H^0(\omega A^{-1})}^T H^0(\omega A^{-1}) \hat{\Phi}(\omega A^{-1}) \\ &= \|\hat{\Phi}(\omega A^{-1})\|_2^2 - \overline{\hat{\Phi}(\omega)}^T \hat{\Phi}(\omega) = \|\hat{\Phi}(\omega A^{-1})\|_2^2 - \|\hat{\Phi}(\omega)\|_2^2 \\ &= \sum_{i=0}^{r-1} \left(|\hat{\phi}_i(\omega A^{-1})|^2 - |\hat{\phi}_i(\omega)|^2 \right). \end{aligned}$$

Setzen wir diesen Ausdruck nun in (2.35) ein, erhalten wir

$$\begin{aligned}
& \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^d} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\langle f, |\det A|^{j/2} \psi_i^l(A^j \cdot -k) \rangle|^2 \\
&= \frac{1}{(2\pi)^d} \sum_{j \in \mathbb{Z}} |\det A|^j \int_{\hat{\mathbb{R}}^d} |\hat{f}(\omega A^j)|^2 \sum_{i=0}^{r-1} \left(|\hat{\phi}_i(\omega A^{-1})|^2 - |\hat{\phi}_i(\omega)|^2 \right) d\omega \\
&= \frac{1}{(2\pi)^d} \int_{\hat{\mathbb{R}}^d} |\hat{f}(\omega)|^2 \sum_{j \in \mathbb{Z}} \sum_{i=0}^{r-1} \left(|\hat{\phi}_i(\omega A^{-(j+1)})|^2 - |\hat{\phi}_i(\omega A^{-j})|^2 \right) d\omega.
\end{aligned}$$

Für die Teleskopsumme kann man gemäß der Voraussetzungen an $\hat{\Phi}$ und A folgende Umformungen vornehmen:

$$\begin{aligned}
& \sum_{j \in \mathbb{Z}} \sum_{i=0}^{r-1} \left(|\hat{\phi}_i(\omega A^{-(j+1)})|^2 - |\hat{\phi}_i(\omega A^{-j})|^2 \right) \\
&= \lim_{j \rightarrow \infty} \sum_{i=0}^{r-1} |\hat{\phi}_i(\omega A^{-(j+1)})|^2 - \lim_{j \rightarrow -\infty} \sum_{i=0}^{r-1} |\hat{\phi}_i(\omega A^{-j})|^2 \\
&= \lim_{j \rightarrow \infty} \sum_{i=0}^{r-1} |\hat{\phi}_i(\omega A^{-j})|^2 - \lim_{j \rightarrow \infty} \sum_{i=0}^{r-1} |\hat{\phi}_i(\omega A^j)|^2 \\
&= \|\hat{\Phi}(0)\|_2^2 - \lim_{j \rightarrow \infty} \|\hat{\Phi}(\omega A^j)\|_2^2 = 1.
\end{aligned}$$

Also erhalten wir insgesamt durch Verwendung von Plancherels Formel im letzten Schritt

$$\sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^d} \sum_{l=1}^{L-1} \sum_{i=0}^{r-1} |\langle f, |\det A|^{j/2} \psi_i^l(A^j \cdot -k) \rangle|^2 = \frac{1}{(2\pi)^d} \int_{\hat{\mathbb{R}}^d} |\hat{f}(\omega)|^2 d\omega = \frac{1}{(2\pi)^d} \|\hat{f}\|_2^2 = \|f\|_2^2.$$

□

Mit einer analogen Beweisführung kann man das allgemeinere Konzept des OEPs für multivariate verfeinerbare Funktionen-Vektoren beweisen.

Satz 2.12 (OEP in der Polyphasen-Darstellung). *Sei Φ ein Skalierungsfunktionen-Vektor, der die Matrix-Verfeinerungsgleichung $\hat{\Phi}(\omega) = H^0(\omega A^{-1})\hat{\Phi}(\omega A^{-1})$ erfüllt. Darüber hinaus nehmen wir an, dass $S(\omega)$ eine $r \times r$ -Matrix sei, deren Einträge trigonometrische Polynome sind, so dass $\|S(0)\hat{\Phi}(0)\|_2^2 = 1$ und $\lim_{j \rightarrow \infty} \|S(\omega A^j)\hat{\Phi}(\omega A^j)\|_2^2 = 0$ erfüllt sind.*

Dann gilt: Wenn für alle $\gamma, \gamma' \in \Gamma$ und für fast alle $\omega \in \hat{\mathbb{R}}^d$ die Bedingung

$$\overline{H_\gamma^0(\omega)}^T \overline{S(\omega)}^T S(\omega) H_{\gamma'}^0(\omega) + \sum_{l=1}^{L-1} \overline{H_\gamma^l(\omega)}^T H_{\gamma'}^l(\omega) = \delta_{\gamma, \gamma'} \overline{S(\omega A^{-1})}^T S(\omega A^{-1}),$$

erfüllt ist, bilden die Multiwavelets $\{|\det A|^{j/2} \psi_i^l(A^j \cdot -k) : l = 1, \dots, L-1; i = 0, \dots, r-1; j \in \mathbb{Z}; k \in \mathbb{Z}^d\}$, die mittels $\hat{\Psi}^l(\omega) = H^l(\omega A^{-1})\hat{\Phi}(\omega A^{-1})$ definiert sind, einen Tight-Frame des $L^2(\mathbb{R}^d)$.

2.3.3 Zusammenhang zwischen Polyphasen- und Modulations-Darstellung

Es besteht wegen (1.14) ein enger Zusammenhang zwischen der Modulationsmatrix $\mathcal{M}(\omega)$ und der Polyphasen-Matrix $\mathcal{P}(\omega)$. Im Fall von skalaren, verfeinerbaren Funktionen können wir diesen Zusammenhang mithilfe einer Matrixmultiplikation ausdrücken.

Lemma 2.13. Sei $\mathcal{M}(\omega) = [H^l(\omega + 2\pi\tilde{\gamma}^T A^{-1})]_{l=0, \tilde{\gamma} \in \Gamma}^{L-1}$ die Modulationsmatrix und $\mathcal{P}(\omega) = [H_{\gamma}^l(\omega)]_{l=0, \gamma \in \Gamma}^{L-1}$ die Polyphasen-Matrix der skalaren, verfeinerbaren Skalierungs- und Waveletfunktionen. Dann gilt die Beziehung

$$\mathcal{M}(\omega) = \mathcal{P}(\omega A) U, \quad (2.36)$$

wobei

$$U = \frac{1}{|\det A|^{1/2}} [e^{-i(\omega + 2\pi\tilde{\gamma}^T A^{-1})\gamma}]_{\gamma, \tilde{\gamma} \in \Gamma}$$

eine unitäre Matrix der Größe $m \times m$ ist. Deshalb gilt: $\mathcal{M}(\omega)$ ist unitär genau dann, wenn $\mathcal{P}(\omega)$ unitär ist für fast alle $\omega \in \hat{\mathbb{R}}^d$.

Beweis. Wir betrachten den (l, j) -ten Eintrag der Matrixgleichung (2.36). Es ist

$$H^l(\omega + 2\pi\tilde{\gamma}_j^T A^{-1}) = \sum_{s=0}^{m-1} H_{\gamma_s}^l(\omega A) \cdot \frac{1}{m^{1/2}} e^{-i(\omega + 2\pi\tilde{\gamma}_j^T A^{-1})\gamma_s} = \frac{1}{m^{1/2}} \sum_{s=0}^{m-1} e^{-i(\omega + 2\pi\tilde{\gamma}_j^T A^{-1})\gamma_s} H_{\gamma_s}^l(\omega A)$$

Wir nutzen die 2π -Periodizität der Polyphasen-Symbole $H_{\gamma}^l(\omega)$ aus und erhalten

$$H^l(\omega + 2\pi\tilde{\gamma}_j^T A^{-1}) = \frac{1}{m^{1/2}} \sum_{s=0}^{m-1} e^{-i(\omega + 2\pi\tilde{\gamma}_j^T A^{-1})\gamma_s} H_{\gamma_s}^l(\omega A + 2\pi\tilde{\gamma}_j^T).$$

Das ist genau die Polyphasen-Zerlegung der Symbole $H^l(\omega + 2\pi\tilde{\gamma}_j^T A^{-1})$ in (1.27). \square

Beispiel 2.14. Wir betrachten die Verfeinerungsgleichung im univariaten Fall ($d = 1, L = 2, A = m = 2$), auch 2-Skalen-Gleichung genannt. Hier zerlegen wir die beiden Symbole H^l , $l = 0, 1$, in gerade und ungerade Polyphasen-Komponenten, so dass gilt

$$\mathcal{M}(\omega) = \begin{pmatrix} H^0(\omega) & H^0(\omega + \pi) \\ H^1(\omega) & H^1(\omega + \pi) \end{pmatrix} = \begin{pmatrix} H_0^0(2\omega) & H_1^0(2\omega) \\ H_0^1(2\omega) & H_1^1(2\omega) \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ e^{-i\omega} & -e^{-i\omega} \end{pmatrix}.$$

Im Fall von vektorwertigen Funktionen ist eine solche Matrix-Multiplikations-Darstellung zwischen \mathcal{M} und \mathcal{P} nicht möglich. Trotzdem existiert eine enge Verbindung, die unter Ausnutzung der besonderen Blockstruktur beider Matrizen beschrieben werden kann. Das formulieren wir im Lemma 2.16, für dessen Beweis wir die folgende Proposition benötigen. Um die Notation zu vereinfachen, definieren wir den Ausdruck $e_{k,j} := e^{-i(\omega + 2\pi\gamma_j^T A^{-1})\gamma_k}$ für alle $k, j = 0, \dots, m-1$, wobei $m = |\Gamma| = |\det A|$ war.

Proposition 2.15. (i) Es gilt: $\sum_{j=0}^{m-1} \overline{e_{k,j}} e_{k',j} = m \delta_{k,k'} \quad \forall k, k' = 0, \dots, m-1$.

(ii) Für festes $j = 0, \dots, m-1$ sind die $e_{k,j}$, $k = 0, \dots, m-1$, linear unabhängig.

(iii) Für Matrizen $A_{k,k'} \in \mathbb{R}^{r \times r}$, $k, k' = 0, \dots, m-1$, gilt

$$\begin{aligned} \sum_{k'=0}^{m-1} A_{k,k'} e_{k',j} &= e_{k,j} I_{r \times r} \quad \forall j, k = 0, \dots, m-1 \\ \Leftrightarrow A_{k,k'} &= \delta_{k,k'} I_{r \times r} \quad \forall k, k' = 0, \dots, m-1. \end{aligned}$$

Beweis. (i) folgt aus [Jia99, Lemma 2.1]; (ii) und (iii) sind Folgerungen aus (i). \square

Lemma 2.16. Sei $\mathcal{M}(\omega)$ die Modulationsmatrix der Größe $rL \times rm$ und $\mathcal{P}(\omega)$ die Polyphasen-Matrix der verfeinerbaren, vektorwertigen Funktionen mit Vielfachheit r .

Dann gilt: $\mathcal{M}(\omega)$ ist unitär genau dann, wenn $\mathcal{P}(\omega)$ unitär ist für fast alle $\omega \in \hat{\mathbb{R}}^d$.

Beweis. Sei $\mathcal{M}(\omega)$ unitär, d. h.

$$\sum_{l=0}^{L-1} \overline{H^l(\omega + 2\pi\tilde{\gamma}_i^T A^{-1})}^T H^l(\omega + 2\pi\tilde{\gamma}_j^T A^{-1}) = \delta_{i,j} I_{r \times r},$$

für $i, j = 0, \dots, m-1$. Zerlegen wir nun die Symbole H^l in ihre Polyphasen-Anteile, so erhalten wir mit (1.16)

$$\begin{aligned} \frac{1}{|\det A|} \sum_{l=0}^{L-1} \overline{\left(\sum_{k=0}^{m-1} e_{k,i} H_{\gamma_k}^l(\omega A) \right)}^T \left(\sum_{k'=0}^{m-1} e_{k',j} H_{\gamma_{k'}}^l(\omega A) \right) &= \delta_{i,j} I_{r \times r} \\ \Leftrightarrow \frac{1}{m} \sum_{k,k'=0}^{m-1} \overline{e_{k,i}} \left(\sum_{l=0}^{L-1} \overline{H_{\gamma_k}^l(\omega A)}^T H_{\gamma_{k'}}^l(\omega A) \right) e_{k',j} &= \delta_{i,j} I_{r \times r}. \end{aligned}$$

Mit $A_{k,k'} := \sum_{l=0}^{L-1} \overline{H_{\gamma_k}^l(\omega A)}^T H_{\gamma_{k'}}^l(\omega A)$ und Proposition 2.15(i) bekommt man

$$\frac{1}{m} \sum_{k=0}^{m-1} \overline{e_{k,i}} \left(\sum_{k'=0}^{m-1} A_{k,k'} e_{k',j} \right) = \frac{1}{m} \sum_{k=0}^{m-1} \overline{e_{k,i}} e_{k,j} I_{r \times r},$$

was gemäß Proposition 2.15(ii) und (iii) äquivalent ist zu $A_{k,k'} = \delta_{k,k'} I_{r \times r}$ für alle $k, k' = 0, \dots, m-1$. Nach Definition von $A_{k,k'}$ bedeutet das gerade Unitarität von $\mathcal{P}(\omega A)$, und damit auch Unitarität von $\mathcal{P}(\omega)$. \square

Das UEP in der üblichen Modulations-Darstellung ist also äquivalent zu dem UEP, das wir mit der Polyphasen-Matrix formuliert haben.

2.3.4 Polyphasen-Untersuchung des gerichteten Haarwavelet-Frames

Im Kapitel 2.2 haben wir ein Waveletsystem konstruiert, das aus nicht-separablen, gerichteten Haarwavelets besteht, die kompakte Dreiecksträger im Zeitbereich besitzen. Die Tight-Frame-Eigenschaft des Systems haben wir in Satz 2.6 mit Argumenten im Zeitbereich bewiesen. Die Ergebnisse des vorigen Abschnitts erlauben uns nun eine Tight-Frame-Charakterisierung mittels der Polyphasen-Matrix im Frequenzbereich.

Dafür benutzen wir die Vektor-Schreibweise der konstruierten \mathcal{AB} -Skalierungsfunktionen und \mathcal{AB} -Wavelets.

Für jedes $i = 0, \dots, 15$, $j \in \mathbb{Z}$ und $k \in \mathbb{Z}^2$, erfüllen die Haar-artigen \mathcal{AB} -Skalierungsfunktionen die Verfeinerungsgleichungen (2.14) bzw. (2.15), die in der Vektorschreibweise mit $\Phi := (\phi_i)_{i=0}^{15}$ die Gestalt

$$\Phi(x) = 2 \sum_{k \in \mathbb{Z}^2} M_k^0 \Phi(2x - k), \quad x \in \mathbb{R}^2, \quad (2.37)$$

haben, wobei M_k^0 Matrix-Koeffizienten der Größe 16×16 der Maske $\{M_k^0\}_{k \in \mathbb{Z}^2}$ sind, und als Einträge nur 0 oder $1/2$ enthalten. Für die Multiwavelets $\Psi^l := (\psi_i^l)_{i=0}^{15}$ gilt analog

$$\Psi^l(x) = 2 \sum_{k \in \mathbb{Z}^2} M_k^l \Phi(2x - k), \quad x \in \mathbb{R}^2, l = 1, 2, 3, \quad (2.38)$$

wobei die Matrix-Koeffizienten M_k^l in diesem Fall die Werte 0, $1/2$ und $-1/2$ enthalten.

Um das UEP anzuwenden, benötigen wir die Darstellung der Funktionen im Frequenzbereich. Die Fouriertransformierten der einzelnen \mathcal{AB} -Skalierungsfunktionen $\phi_{i,j,k} = 2^j \phi_i(2^j \cdot -k)$ können einfach berechnet werden durch

$$\hat{\phi}_{i,j,k}(\omega) = 2^{-j} e^{-i \frac{\omega}{2^j} k} \hat{\phi}_i \left(\frac{\omega}{2^j} \right).$$

Wir beschränken uns wieder exemplarisch auf das Dreieck U_0 . Dort haben z. B. die Verfeinerungsgleichungen (2.12) und (2.20)-(2.21) im Frequenzbereich die Form

$$\begin{aligned} \hat{\phi}_0(\omega) &= \frac{1}{4} \left(\hat{\phi}_0 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_0 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_1 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 1 \\ 2 \end{pmatrix}} \hat{\phi}_9 \left(\frac{\omega}{2} \right) \right), \\ \hat{\psi}_0^1(\omega) &= \frac{1}{4} \left(\hat{\phi}_0 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_0 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_1 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 1 \\ 2 \end{pmatrix}} \hat{\phi}_9 \left(\frac{\omega}{2} \right) \right), \\ \hat{\psi}_0^2(\omega) &= \frac{1}{4} \left(\hat{\phi}_0 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_0 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_1 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 1 \\ 2 \end{pmatrix}} \hat{\phi}_9 \left(\frac{\omega}{2} \right) \right), \\ \hat{\psi}_0^3(\omega) &= \frac{1}{4} \left(\hat{\phi}_0 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_0 \left(\frac{\omega}{2} \right) + e^{-i \frac{\omega}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} \hat{\phi}_1 \left(\frac{\omega}{2} \right) - e^{-i \frac{\omega}{2} \begin{pmatrix} 1 \\ 2 \end{pmatrix}} \hat{\phi}_9 \left(\frac{\omega}{2} \right) \right). \end{aligned}$$

In Multiwavelet-Schreibweise sind die Symbole von (2.37) und (2.38) mit $\Psi^0 := \Phi$ durch

$$\hat{\Psi}^l(\omega) = H^l \left(\frac{\omega}{2} \right) \hat{\Psi}^0 \left(\frac{\omega}{2} \right), \quad l = 0, \dots, 3,$$

gegeben. Die Symbole H^l sind wegen des kompakten Trägers der Funktionen endliche Summen

$$H^l(\omega) = \frac{1}{2} \sum_{k \in \mathbb{Z}^2} M_k^l e^{-i \omega k},$$

die in ihre Polyphasen-Anteile zerlegt werden können. Und zwar ist gemäß (1.14)

$$\begin{aligned} H^l(\omega) &= \frac{1}{2} \left(H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^l e^{-i \omega \begin{pmatrix} 0 \\ 0 \end{pmatrix}} + H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^l e^{-i \omega \begin{pmatrix} 1 \\ 0 \end{pmatrix}} + H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^l e^{-i \omega \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^l e^{-i \omega \begin{pmatrix} 1 \\ 1 \end{pmatrix}} \right) \\ &= \frac{1}{2} \left(H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^l + H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^l e^{-i \omega_1} + H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^l e^{-i \omega_2} + H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^l e^{-i(\omega_1 + \omega_2)} \right), \end{aligned}$$

weil für die Dilatationsmatrix $A = 2I_2$ das Gitter \mathbb{Z}^2 in $m = 4$ Co-Mengen zerlegt werden kann, die jeweils durch einen Vektor aus $\Gamma = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}$ repräsentiert werden.

Mit der Absicht, eine möglichst einfache Polyphasen-Matrix zu erhalten, verschieben wir die Framefunktionen wieder so, dass ihre Träger in $[0, 1]^2$ liegen, d.h. wir betrachten die translatierten Versionen von ϕ_i bzw. von ψ_i^l in dem Funktionenvektor

$$\Psi_t^l := \left(\psi_0^l, \dots, \psi_3^l, \psi_4^l(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}), \dots, \psi_7^l(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}), \right. \\ \left. \psi_8^l(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \dots, \psi_{11}^l(\cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}), \psi_{12}^l(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}), \dots, \psi_{15}^l(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) \right)^T,$$

für $l = 0, 1, 2, 3$ und $\psi_i^0 = \phi_i$. Es ist klar, dass dieser Funktionenvektor dieselbe Frame-MRA erzeugt und damit dasselbe Waveletsystem. Dann hat die zu Ψ_t^l , $l = 0, 1, 2, 3$, gehörende Polyphasen-Matrix der Größe 64×64 die folgende Blockstruktur

$$\mathcal{P}(\omega) := [H_\gamma^l(\omega)]_{l=0, \gamma \in \Gamma}^3 = \begin{pmatrix} H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^0 & H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^0 & H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^0 & H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^0 \\ H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^1 & H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^1 & H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^1 & H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^1 \\ H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^2 & H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^2 & H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^2 & H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^2 \\ H_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}^3 & H_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^3 & H_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}^3 & H_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}^3 \end{pmatrix},$$

Die verschiedenen Blöcke H_γ^l sind 16×16 -Matrizen gemäß dieses Schemas:

	$\gamma = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\gamma = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\gamma = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$	$\gamma = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$l = 0$	I_4 0 0 0 0 D_1 0 D_2 0 0 D_4 0 0 D_3 0 D_0	D_1 0 D_2 0 0 D_4 0 0 D_3 0 D_0 0 0 0 0 I_4	D_0 0 D_3 0 0 I_4 0 0 D_2 0 D_1 0 0 0 0 D_4	D_4 0 0 0 0 D_0 0 D_3 0 0 I_4 0 0 D_2 0 D_1
$l = 1$	I_4 0 0 0 0 \tilde{D}_1 0 $-D_2$ 0 0 \tilde{D}_4 0 0 $-D_3$ 0 \tilde{D}_0	\tilde{D}_1 0 $-D_2$ 0 0 \tilde{D}_4 0 0 $-D_3$ 0 \tilde{D}_0 0 0 0 0 I_4	\tilde{D}_0 0 $-D_3$ 0 0 I_4 0 0 $-D_2$ 0 \tilde{D}_1 0 0 0 0 \tilde{D}_4	\tilde{D}_4 0 0 0 0 \tilde{D}_0 0 $-D_3$ 0 0 I_4 0 0 $-D_2$ 0 \tilde{D}_1
$l = 2$	I_4 0 0 0 0 $-D_1$ 0 D_2 0 0 $-D_4$ 0 0 D_3 0 $-D_0$	$-D_1$ 0 D_2 0 0 $-D_4$ 0 0 D_3 0 $-D_0$ 0 0 0 0 I_4	$-D_0$ 0 D_3 0 0 I_4 0 0 D_2 0 $-D_1$ 0 0 0 0 $-D_4$	$-D_4$ 0 0 0 0 $-D_0$ 0 D_3 0 0 I_4 0 0 D_2 0 $-D_1$
$l = 3$	I_4 0 0 0 0 $-\tilde{D}_1$ 0 $-D_2$ 0 0 $-\tilde{D}_4$ 0 0 $-D_3$ 0 $-\tilde{D}_0$	$-\tilde{D}_1$ 0 $-D_2$ 0 0 $-\tilde{D}_4$ 0 0 $-D_3$ 0 $-\tilde{D}_0$ 0 0 0 0 I_4	$-\tilde{D}_0$ 0 $-D_3$ 0 0 I_4 0 0 $-D_2$ 0 $-\tilde{D}_1$ 0 0 0 0 $-\tilde{D}_4$	$-\tilde{D}_4$ 0 0 0 0 $-\tilde{D}_0$ 0 $-D_3$ 0 0 I_4 0 0 $-D_2$ 0 $-\tilde{D}_1$

mit

$$D_0 := \frac{1}{2} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tilde{D}_0 := \frac{1}{2} \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, D_1 := \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\tilde{D}_1 := \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, D_2 := \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, D_3 := \frac{1}{2} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$D_4 := \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \tilde{D}_4 := \frac{1}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}.$$

In dem Schema kann man leicht sehen, dass die verschiedenen Spalten der Polyphasen-Matrix orthogonal zueinander sind. Dies liegt an der speziellen Blockstruktur. Damit ist also $\mathcal{P}(\omega)$ unitär. Die Anwendung von Satz 2.11 (mit $d = 2, r = 16, A = 2I_2, L = m = 4$) liefert nun in der Tat, dass die konstruierten Haarwavelets auf Dreiecken einen Parseval-Frame für den Raum $L^2(\mathbb{R}^2)$ bilden.

2.4 Waveletframes mit beliebig vielen Richtungen

In diesem Unterkapitel wollen wir Waveletsysteme betrachten, die eine noch höhere Richtungssensitivität aufweisen als die bisher konstruierten Waveletframes. Es ist sinnvoll, die Anzahl der Richtungen von dem jeweiligen Zerlegungslevel abhängig zu machen. Auf diese Weise ist man bei der zugehörigen Filterbank in der Lage, die Richtungssensitivität (und damit die Redundanz) des Waveletsystems zu regulieren. Genauer gesagt, in jedem Level $j \geq 0$ verwenden wir 2^{j+2} Richtungen; das bedeutet, die Anzahl der Richtungen vom groben zum feineren Level wird verdoppelt. Diese Eigenschaft zeichnet auch die Shearletsysteme aus (siehe [ELL08]), bei den Curvelet- und Contourletsystemen verdoppeln sich die Richtungen nur in jedem zweiten Level.

Wir verallgemeinern unseren Dreiecksansatz aus Kapitel 2.1 und 2.2, indem wir in den feineren Leveln schmalere Dreieckszerlegungen zulassen. Dazu verwenden wir die Matrizen

$$A = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix} \quad \text{und} \quad S = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Die parabolische Dilatationsmatrix A bewirkt, dass die Dreiecksflächen in jedem Level um den Faktor vier in x_1 -Richtung und um den Faktor zwei in x_2 -Richtung kleiner werden. Durch die Schermatrix S erzeugen wir die verschiedenen Richtungen, wir konstruieren also ein Shearletframe, wobei die Shearlets im Zeitbereich einen kompakten Träger auf Dreiecken haben.

2.4.1 Konstruktion von Skalierungs- und Waveletfunktionen

Wie in den vorigen Kapiteln beginnen wir mit der Zerlegung der \mathbb{R}^2 -Ebene in geeignete Dreiecke, auf denen wir die Skalierungsfunktionen als Indikatorfunktionen definieren. Allerdings ändern wir hier die bisherige Indizierung der Dreiecke bzw. der korrespondierenden

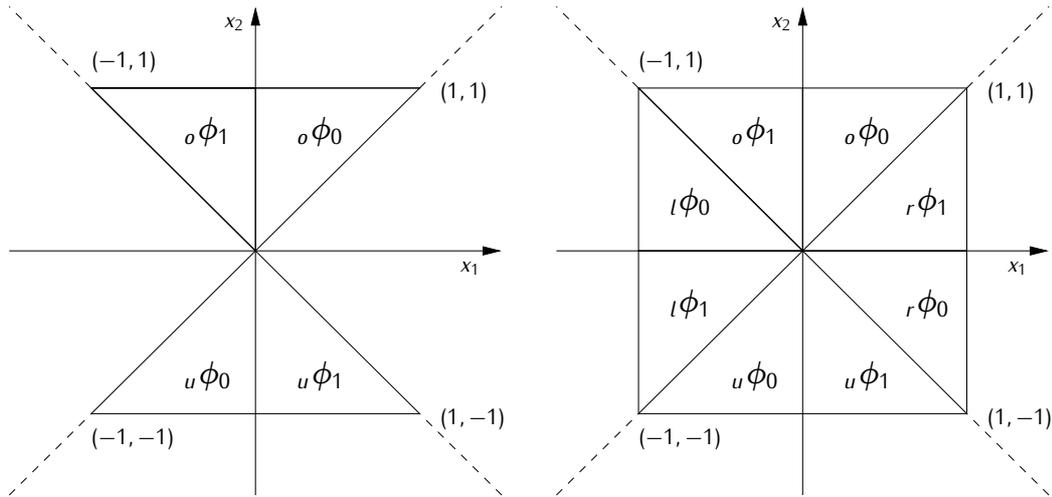


Abbildung 2.15: Dreiecksträger der Mutter-Skalierungsfunktionen im Level $j = 0$. (a) Träger von ${}_o\phi_0$ und ${}_o\phi_1$. (b) Träger der übrigen Skalierungsfunktionen.

Skalierungs- und Waveletfunktionen. Zunächst zerlegen wir die Ebene in vier Keile, die durch die beiden Hauptdiagonalen $x_1 = x_2$ und $x_1 = -x_2$ begrenzt sind. Die beiden Mutter-Skalierungsfunktionen im oberen Kegel sind die charakteristische Funktion auf dem Dreieck

$$U_0 = \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} := \left\{ x \in \mathbb{R}^2 : 0 \leq x_2 \leq 1, 0 \leq x_1 \leq x_2 \right\},$$

also

$${}_o\phi_0(x) = {}_o\phi_0(x_1, x_2) := 2\chi_{[0,1]} \left(\frac{x_1}{x_2} \right) \chi_{[0,1]}(x_2),$$

sowie ihre gescherzte Version ${}_o\phi_1(x) := {}_o\phi_0(S^{-1}x)$. Die an der x_1 -Achse gespiegelten Versionen von ${}_o\phi_0$ und ${}_o\phi_1$ sind die Mutter-Skalierungsfunktionen des unteren Kegels

$${}_u\phi_0(x) := {}_o\phi_0(-x) \quad \text{und} \quad {}_u\phi_1(x) := {}_o\phi_1(-x),$$

siehe Abbildung 2.15(a). Das voran- und tiefgestellte „u“ soll die Skalierungsfunktionen im unteren Kegel bezeichnen, ebenso wie das „o“ die des oberen Kegels bezeichnete. Analog sind die Mutter-Skalierungsfunktionen in den horizontalen Kegeln durch

$$\begin{aligned} {}_l\phi_0(x_1, x_2) &= {}_o\phi_0(x_2, -x_1) \quad \text{und} \quad {}_l\phi_1(x_1, x_2) = {}_o\phi_1(-x_2, -x_1), \quad \text{sowie} \\ {}_r\phi_0(x_1, x_2) &= {}_o\phi_0(-x_2, x_1) \quad \text{und} \quad {}_r\phi_1(x_1, x_2) = {}_o\phi_1(x_2, x_1) \end{aligned}$$

definiert, wie Abbildung 2.15(b) anschaulich darstellt. Das entspricht dem Ansatz in Kapitel 2.1; vergleiche dazu Abbildung 2.15(b) mit Abbildung 2.1(a).

Für jede dieser Funktionen können wir uns die verfeinerten Versionen anschauen, indem wir mit A^j , $j \geq 0$, skalieren. Zusätzlich erlauben wir Scherungen mit S^i für $-2^j + 1 \leq i \leq 2^j$ und Translationen um $k \in \mathbb{Z}^2$. Die allgemeinen Skalierungsfunktionen im Level $j \geq 0$ sind also wieder Funktionen mit zusammengesetzter Dilatation [GLL⁺06, Lim06]

$${}_o\phi_{i,j,k}(x) := |\det A|^{j/2} {}_o\phi_i(A^j x - k) = |\det A|^{j/2} {}_o\phi_0(S^i(A^j x - k)), \quad (2.39)$$

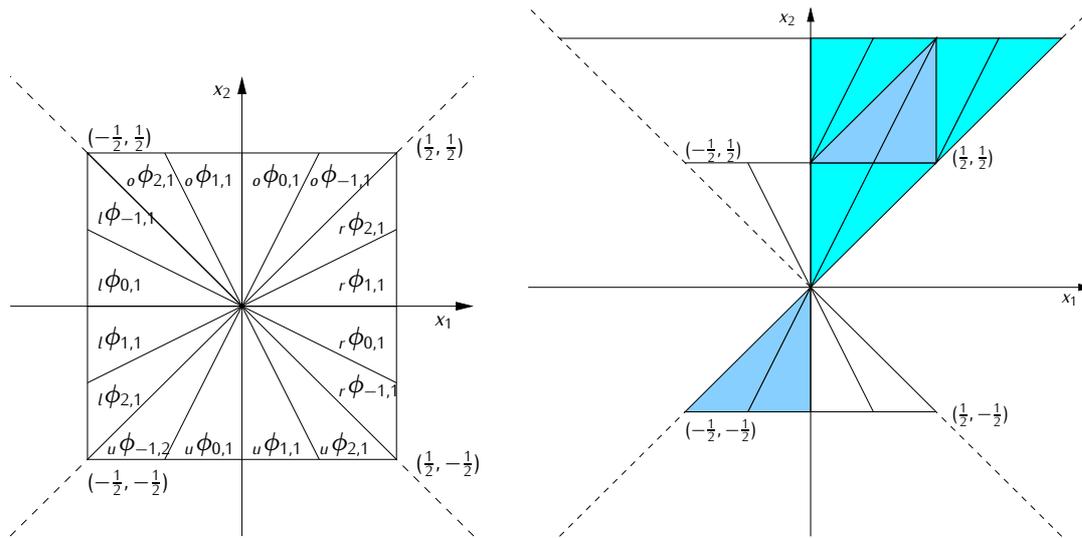


Abbildung 2.16: (a) Die verfeinerten Skalierungsfunktionen im Level $j = 1$. (b) Illustration der Verfeinerung von ${}_o\phi_0$.

und für die Skalierungsfunktionen der anderen drei Kegel „u“, „l“ und „r“ analog. So erhalten wir für $j = 1$ in jedem Kegel vier feinere Dreiecksfunktionen durch die Scherungen $i = -1, 0, 1, 2$, siehe Abbildung 2.16(a). Das entspricht dem Ansatz in Kapitel 2.2; vergleiche dazu Abbildung 2.16(a) mit Abbildung 2.5(a).

Bei den Verfeinerungsgleichungen benötigen wir entweder Funktionen aus dem vertikalen Doppelkegel oder aus dem horizontalen. Deshalb können wir uns im Folgenden auf den vertikalen Doppelkegel („o“ und „u“) beschränken. Anhand der Verfeinerungsgleichungen werden wir erkennen können, dass es sich bei unserer Konstruktion um „echte“ Shearlets handelt, also um Funktionen, die nur mit Hilfe ihrer gescherten Versionen verfeinerbar sind.

Für die Verfeinerungen benötigen wir eine Art Kommutativität der beiden Matrizen A und S . Es gilt

$$AS^i = S^{2i}A \quad \text{für } i \in \mathbb{Z}. \quad (2.40)$$

Damit sind wir in der Lage die Verfeinerungsgleichungen anzugeben, wir beschränken uns hier auf den Fall des oberen Kegels, die anderen Gleichungen lauten ähnlich. Die umfangreiche allgemeine Gleichung wird deutlicher, wenn man sich zuvor in Abbildung 2.16(b) anschaulich macht, wie ${}_o\phi_0$ mittels 8 feinerer Skalierungsfunktionen ${}_o\phi_{i,1,k}$ und ${}_u\phi_{i,1,k}$ für $i = -1, 0, 1, 2$ und $k \in \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 2 \end{pmatrix} \right\}$ verfeinert wird. Es gilt hier nämlich mit $|\det A|^{j/2} = 2^{3j/2}$

$$\begin{aligned} {}_o\phi_0 = 2^{-3/2} & \left({}_o\phi_{0,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + {}_o\phi_{0,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_o\phi_{0,1,\begin{pmatrix} 2 \\ 1 \end{pmatrix}} + {}_o\phi_{-1,1,\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + {}_o\phi_{-1,1,\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_o\phi_{-1,1,\begin{pmatrix} 2 \\ 1 \end{pmatrix}} \right. \\ & \left. + {}_u\phi_{0,1,\begin{pmatrix} 2 \\ 2 \end{pmatrix}} + {}_u\phi_{-1,1,\begin{pmatrix} 2 \\ 2 \end{pmatrix}} \right). \quad (2.41) \end{aligned}$$

Allgemein gilt nun für $j \geq 0$, $-2^j + 1 \leq i \leq 2^j$ und $k \in \mathbb{Z}^2$ nach (2.39)

$${}_o\phi_{i,j,k} = 2^{3j/2} {}_o\phi_0(S^i(A^j \cdot -k)).$$

Wir können also die Verfeinerung von ${}_o\phi_{i,j,k}$ auf die von ${}_o\phi_0$ zurückführen. Dazu setzen wir das Argument $S^i(A^j \cdot -k)$ in die Verfeinerungsgleichung von ϕ_0 in (2.41) ein

$$\begin{aligned} {}_o\phi_{i,j,k} &= 2^{3j/2} \left({}_o\phi_0(AS^i(A^j \cdot -k)) + {}_o\phi_0(AS^i(A^j \cdot -k) - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + {}_o\phi_0(AS^i(A^j \cdot -k) - \begin{pmatrix} 2 \\ 1 \end{pmatrix}) \right. \\ &+ {}_o\phi_0(S^{-1}AS^i(A^j \cdot -k)) + {}_o\phi_0(S^{-1}(AS^i(A^j \cdot -k) - \begin{pmatrix} 0 \\ 1 \end{pmatrix})) + {}_o\phi_0(S^{-1}(AS^i(A^j \cdot -k) - \begin{pmatrix} 2 \\ 1 \end{pmatrix})) \\ &\left. + {}_u\phi_0(AS^i(A^j \cdot -k) - \begin{pmatrix} 2 \\ 2 \end{pmatrix}) + {}_u\phi_0(S^{-1}(AS^i(A^j \cdot -k) - \begin{pmatrix} 2 \\ 2 \end{pmatrix})) \right). \end{aligned}$$

Unter Verwendung von (2.40) und anschließendem Ausklammern von S^{2i} bzw. S^{2i+1} können wir diesen Ausdruck umschreiben zu

$$\begin{aligned} {}_o\phi_{i,j,k} &= 2^{3j/2} \left({}_o\phi_0(S^{2i}(A^{j+1} \cdot -Ak)) + {}_o\phi_0(S^{2i}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}))) \right. \\ &+ {}_o\phi_0(S^{2i}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}))) + {}_o\phi_0(S^{2i-1}(A^{j+1} \cdot -Ak)) \\ &+ {}_o\phi_0(S^{2i-1}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}))) + {}_o\phi_0(S^{2i-1}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}))) \\ &\left. + {}_u\phi_0(S^{2i}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}))) + {}_u\phi_0(S^{2i-1}(A^{j+1} \cdot -(Ak + S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}))) \right). \end{aligned}$$

Wenden wir (2.39) an, so erhalten wir schließlich

$$\begin{aligned} {}_o\phi_{i,j,k} &= 2^{-3j/2} \left({}_o\phi_{2i,j+1,Ak} + {}_o\phi_{2i,j+1,Ak+S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_o\phi_{2i,j+1,Ak+S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}} \right. \\ &+ {}_o\phi_{2i-1,j+1,Ak} + {}_o\phi_{2i-1,j+1,Ak+S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_o\phi_{2i-1,j+1,Ak+S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}} \\ &\left. + {}_u\phi_{2i,j+1,Ak+S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}} + {}_u\phi_{2i-1,j+1,Ak+S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}} \right). \quad (2.42) \end{aligned}$$

Analog ergibt sich

$$\begin{aligned} {}_u\phi_{i,j,k} &= 2^{-3j/2} \left({}_u\phi_{2i,j+1,Ak} + {}_u\phi_{2i,j+1,Ak-S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_u\phi_{2i,j+1,Ak-S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}} \right. \\ &+ {}_u\phi_{2i-1,j+1,Ak} + {}_u\phi_{2i-1,j+1,Ak-S^{-2i} \begin{pmatrix} 0 \\ 1 \end{pmatrix}} + {}_u\phi_{2i-1,j+1,Ak-S^{-2i} \begin{pmatrix} 2 \\ 1 \end{pmatrix}} \\ &\left. + {}_o\phi_{2i,j+1,Ak-S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}} + {}_o\phi_{2i-1,j+1,Ak-S^{-2i} \begin{pmatrix} 2 \\ 2 \end{pmatrix}} \right). \quad (2.43) \end{aligned}$$

Wir führen jetzt die bevorzugte Vektor-Notation ein. Der Vektor der Skalierungsfunktionen (für den vertikalen Doppelkegel) im Level $j = 0$ hat die Form

$$\Phi_0(\cdot - k) := ({}_o\phi_{0,0,k}, {}_o\phi_{1,0,k}, {}_u\phi_{0,0,k}, {}_u\phi_{1,0,k})^T.$$

Im nächstfeineren Level verdoppelt sich die Anzahl der Skalierungsfunktionen, man erhält

$$\Phi_1(\cdot - k) := ({}_o\phi_{-1,1,k}, {}_o\phi_{0,1,k}, {}_o\phi_{1,1,k}, {}_o\phi_{2,1,k}, {}_u\phi_{-1,1,k}, {}_u\phi_{0,1,k}, {}_u\phi_{1,1,k}, {}_u\phi_{2,1,k})^T.$$

Die vektorwertige Verfeinerungsgleichung lautet

$$\Phi_0 = 2^{3/2} \sum_{k \in \mathbb{Z}^2} M_{1,k}^0 \Phi_1(\cdot - k)$$

mit den Matrix-Koeffizienten $M_{j+1,k}^0 \in \mathbb{R}^{4 \times 8}$ der Maske $\{M_{1,k}^0\}_{k \in \mathbb{Z}^2}$, die sich aus (2.42) und (2.43) ergeben. Die Verdoppelung der Anzahl der Skalierungsfunktionen führt zu einer *instationären MRA*, bei der sich in jedem Skalierungslevel die Dimension der Matrix-Koeffizienten verdoppelt. Für ein beliebiges Level $j \in \mathbb{Z}$ gilt die Verfeinerung

$$\Phi_j = 2^{3/2} \sum_{k \in \mathbb{Z}^2} M_{j+1,k}^0 \Phi_{j+1}(\cdot - k)$$

mit $M_{1,k}^0 \in \mathbb{R}^{2^{j+2} \times 2^{j+3}}$ und den Funktionen-Vektoren

$$\Phi_j(\cdot - k) = ({}_o\phi_{-2j+1,j,k}, \dots, {}_o\phi_{2j,j,k}, {}_u\phi_{-2j+1,j,k}, \dots, {}_u\phi_{2j,j,k})^T \in (L^2(\mathbb{R}^2))^{2^{j+2}}$$

und $\Phi_{j+1} \in (L^2(\mathbb{R}^2))^{2^{j+3}}$ entsprechend. Im Frequenzbereich hat die Skalierungsgleichung die Form

$$\hat{\Phi}_j(\omega) = H_{j+1}^0(\omega A^{-1}) \hat{\Phi}_{j+1}(\omega A^{-1})$$

mit dem Symbol $H_j^0(\omega) = 2^{-3/2} \sum_{k \in \mathbb{Z}^d} M_{j,k}^0 e^{-i\omega k}$. Die zugehörigen Wavelet-Vektoren für $l = 1, \dots, 7$ (wegen $|\det A| - 1 = 7$),

$$\Psi_j^l(\cdot - k) := ({}_o\psi_{-2j+1,j,k}^l, \dots, {}_o\psi_{2j,j,k}^l, {}_u\psi_{-2j+1,j,k}^l, \dots, {}_u\psi_{2j,j,k}^l)^T \in (L^2(\mathbb{R}^2))^{2^{j+2}},$$

erklären wir in naheliegender Weise über die Verfeinerungsgleichung

$$\Psi_j^l = 2^{3/2} \sum_{k \in \mathbb{Z}^2} M_{j+1,k}^l \Phi_{j+1}(\cdot - k), \quad l = 1, \dots, 7,$$

mit den matrixwertigen Waveletkoeffizienten $M_{j+1,k}^l \in \mathbb{R}^{2^{j+2} \times 2^{j+3}}$ der sieben Wavelet-Masken $\{M_{j+1,k}^l\}_{k \in \mathbb{Z}^2}$, $l = 1, \dots, 7$. Und zwar besteht jede Koeffizientenmatrix $M_{j+1,k}^l$ aus den Einträgen $\frac{1}{\sqrt{8}}$ oder $-\frac{1}{\sqrt{8}}$, die so angeordnet sind, dass sie der Haarwavelet-Konstruktion für alle $j \in \mathbb{Z}$, $-2^j + 1 \leq i \leq 2^j$, $k \in \mathbb{Z}^2$ entsprechen:

$$\begin{pmatrix} {}_o\phi_{i,j,k} \\ {}_o\psi_{i,j,k}^1 \\ {}_o\psi_{i,j,k}^2 \\ {}_o\psi_{i,j,k}^3 \\ {}_o\psi_{i,j,k}^4 \\ {}_o\psi_{i,j,k}^5 \\ {}_o\psi_{i,j,k}^6 \\ {}_o\psi_{i,j,k}^7 \end{pmatrix} := \frac{1}{\sqrt{8}} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} {}_o\phi_{2i,j+1, Ak} \\ {}_o\phi_{2i,j+1, Ak+S^{-2i}} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ {}_o\phi_{2i,j+1, Ak+S^{-2i}} \begin{pmatrix} 2 \\ 1 \end{pmatrix} \\ {}_o\phi_{2i-1,j+1, Ak} \\ {}_o\phi_{2i-1,j+1, Ak+S^{-2i}} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ {}_o\phi_{2i-1,j+1, Ak+S^{-2i}} \begin{pmatrix} 2 \\ 1 \end{pmatrix} \\ {}_u\phi_{2i,j+1, Ak+S^{-2i}} \begin{pmatrix} 2 \\ 2 \end{pmatrix} \\ {}_u\phi_{2i-1,j+1, Ak+S^{-2i}} \begin{pmatrix} 2 \\ 2 \end{pmatrix} \end{pmatrix}.$$

Beachte, dass die erste Zeile genau der Verfeinerungsgleichung (2.42) entspricht. Die Verfeinerungsgleichung der Wavelets im Frequenzbereich lautet

$$\hat{\Psi}_j^l(\omega) = H_{j+1}^l(\omega A^{-1}) \hat{\Phi}_{j+1}(\omega A^{-1}), \quad l = 1, \dots, 7,$$

mit den Waveletsymbolen $H_j^l(\omega) = 2^{-3j/2} \sum_{k \in \mathbb{Z}^d} M_{j,k}^l e^{-i\omega k}$. Wir erkennen, dass es sich bei den Wavelets in der Tat um Shearlets handelt, wenn wir in Analogie zu (2.39) die Schreibweise

$${}_o\psi_{i,j,k}^l(x) = 2^{3j/2} {}_o\psi_0^l(S^i(A^j x - k)), \quad l = 1, \dots, 7,$$

verwenden, wobei $j \geq 0$, $-2^j + 1 \leq i \leq 2^j$ und $k \in \mathbb{Z}^d$ gilt. Ebenso erhält man die Shearletdarstellung für die anderen drei Kegel „u“, „l“ und „r“.

2.4.2 Die Polyphasen-Matrix und die Tight-Frame-Eigenschaft

Wir wollen die Frame-Eigenschaft des Shearletsystems wieder mit der Unitarität der Polyphasen-Matrix beweisen. Wir verallgemeinern dazu Satz 2.11 für den Fall von *instationären* MRAs. Weil sich die Anzahl der Skalierungsfunktionen bzw. der Shearlets in jedem Level verändert, verändern sich auch die Symbole und mithin die Polyphasen-Matrix. Anstatt einmal die Unitarität der Polyphasen-Matrix zu überprüfen, müssen wir jetzt in jedem Level $j \geq 0$ aufs Neue die Unitarität der Polyphasen-Matrix $\mathcal{P}_j(\omega)$ zeigen. Mit $r(j)$ bezeichnen wir im folgenden Satz die Länge der Funktionenvektoren im Level j . In unserem Fall ist $r(j) = 2^{j+2}$, wie wir oben gesehen haben.

Satz 2.17 (UEP in der Polyphasen-Darstellung im instationären Fall). *Sei Φ_j der Vektor, der die Skalierungsfunktionen im Level $j \geq 0$ enthält und der eine Matrix-Verfeinerungsgleichung*

$$\hat{\Phi}_j(\omega) = H_{j+1}^0(\omega A^{-1}) \hat{\Phi}_{j+1}(\omega A^{-1})$$

erfüllt. Weiter soll $\|\hat{\Phi}_j(0)\|_2^2 = 1$ und $\lim_{j \rightarrow \infty} \|\hat{\Phi}_j(\omega A^j)\|_2^2 = 0$ gelten. Dann gilt: Wenn die Polyphasen-Matrizen $\mathcal{P}_j(\omega)$ in jedem Level $j \geq 0$ unitär sind für fast alle $\omega \in \hat{\mathbb{R}}^d$, so erzeugen die Multiwavelets

$$\{ |\det A|^{j/2} \psi_i^l(A^j \cdot -k) : l = 1, \dots, L-1; j \geq 0; i = 0, \dots, r(j) - 1; k \in \mathbb{Z}^d \},$$

die mittels $\hat{\Psi}_j^l(\omega) = H_{j+1}^l(\omega A^{-1}) \hat{\Phi}_{j+1}(\omega A^{-1})$ definiert sind, einen instationären Tight-Frame des $L^2(\mathbb{R}^d)$, d. h. es gibt eine Konstante $C > 0$ mit

$$C \|f\|_2^2 = \sum_{j \geq 0} \sum_{k \in \mathbb{Z}^d} \sum_{l=1}^{L-1} \sum_{i=0}^{r(j)-1} |\langle f, |\det A|^{j/2} \psi_i^l(A^j \cdot -k) \rangle|^2 \quad \forall f \in L^2(\mathbb{R}^d).$$

Beweis. Der Beweis verläuft analog zu dem Beweis von Satz 2.11. Wir müssen lediglich anstatt der Vektoren $\hat{\Phi}$, $\hat{\Psi}^l$ und Symbole H^l die levelabhängigen Versionen $\hat{\Phi}_j$, $\hat{\Psi}_j^l$ und H_j^l betrachten, $l = 0, \dots, L-1$. □

Um diesen Satz auf unser Shearletsystem anwenden zu können, müssen wir die Polyphasen-Matrizen $\mathcal{P}_j(\omega) := [H_{j,\nu}^l]_{l=0,\nu \in \Gamma}^7$ bestimmen. Unsere Untersuchungen aus den Abschnitten 2.1 bis 2.3 legen folgende Vermutung nahe.

Vermutung 2.18. *Das Shearletsystem*

$$\{o\psi_{i,j,k}^l, u\psi_{i,j,k}^l, l\psi_{i,j,k}^l, r\psi_{i,j,k}^l : l = 1, \dots, 7; j \geq 0; i = -2^j + 1, \dots, 2^j; k \in \mathbb{Z}^2\},$$

ist ein instationärer Tight-Frame des $L^2(\mathbb{R}^d)$.

2.5 Ausblick: Stetige Wavelets auf Dreiecken

Die konstruierten Wavelets auf Dreiecken sind wegen ihrer hohen Richtungssensitivität gut geeignet um gerichtete Bildkanten zu adaptieren. Allerdings besitzen die Haar-ähnlichen Wavelets nur ein verschwindendes Moment. Anders ausgedrückt: Die stückweise konstanten Skalierungsfunktionen reproduzieren nur stückweise konstante Funktionen. Die Approximationsqualität könnte weiter verbessert werden, wenn wir die Regularität der Wavelets bzw. der zugehörigen Skalierungsfunktionen erhöhen könnten. In diesem Abschnitt stellen wir einen ersten Ansatz in diese Richtung vor. Wir konstruieren dazu auf einem Dreiecksgitter ein Paar aus linearen Splinefunktionen, deren Verfeinerbarkeit auf natürlichem Weg zu einer Folge von Approximationsräumen führt, die eine Riesz-MRA des $L^2(\mathbb{R}^2)$ bildet. Aber weder Skalierungs- noch Waveletfunktionen sind nunmehr orthogonal zu ihren Translationen, so dass duale Funktionen betrachtet werden müssen. Dabei stellt es sich als schwierig heraus, solche dualen Skalierungsfunktionen und Wavelets zu finden, denen eine biorthogonale MRA zugrunde liegt.

4-Richtungs-Gitter Unsere erste Dreieckskonstruktion im Abschnitt 2.1 bevorzugte neben den beiden Achsenrichtungen die Hauptdiagonalen, so dass ein *4-Richtungs-Gitter* entstand, das in der Literatur *triangulierte Quadrangulierung* [GH06, LS07] genannt wird, weil man es durch das Einfügen von Diagonalen in einem Quadrat konstruieren kann. Genauer gesagt, haben wir das Quadrat $[-1/2, 1/2]^2$ mithilfe der Diagonalen in acht Dreiecke gleichen Flächeninhalts zerlegt. Unter Berücksichtigung der Translationen wurde auf diese Weise ein Gitter erzeugt, das durch die Gitterlinien

$$x = i/2, y = i/2, x + y = i, x - y = i \quad (i \in \mathbb{Z}),$$

definiert ist. Im nächst größeren Level haben wir dann konstante Skalierungsfunktionen betrachtet, die auf dem größeren Gitter

$$x = i, y = i, x + y = 2i, x - y = 2i \quad (i \in \mathbb{Z}),$$

„lebten“. Das zeigt, dass die Verfeinerung bzw. Vergrößerung des Gitters mithilfe der Dilationsmatrix $A = 2I_2$ geschah.

In diesem Kapitel verwenden wir zur Konstruktion der linearen Spline-Wavelets erneut ein 4-Richtungs-Gitter, allerdings leicht modifiziert gemäß dem Ansatz von Goodman et al. in [GLT93]. Das Gitter Δ besteht dabei aus Dreiecksflächen $T \in \Delta$, die durch die Gitterlinien

$$x = i, y = i, x + y = i, x - y = i \quad (i \in \mathbb{Z}) \quad (2.44)$$

bestimmt sind. Dieses Gitter ist bekannt unter den Bezeichnungen „crisscross lattice“ oder „ $\Delta^{(2)}$ -Triangulierung“ bzw. „Typ-2-Triangulierung“. Durch das Gitter Δ wird ein Spline-Raum

$$S_1^0(\Delta) := \left\{ s \in C^0(\mathbb{R}^2) \cap L^2(\mathbb{R}^2) : s|_T \in \Pi_1, T \in \Delta \right\},$$

generiert, wobei $\Pi_1 = \{p : p(x, y) = \alpha x + \beta y + \gamma, \alpha, \beta, \gamma \in \mathbb{R}\}$ der Raum der linearen bivariaten Funktionen ist.

Verbindung zu anderen Methoden Einen schönen Überblick über verschiedene Ansätze in der Literatur für die Konstruktion von verfeinerbaren, multivariaten Splinefunktionen mit kompakten Träger bieten die Arbeiten [GH06] und [LS07].

Was die Konstruktion von stückweise, linearen Wavelets auf Triangulierungen angeht, sind zunächst die Box-Splines [dBHR93] zu nennen. Der erhebliche Nachteil ist, dass ihre \mathbb{Z}^2 -Translationen keine Basis für den Raum $S_1^0(\Delta)$ bilden, weil sie kein Erzeugendensystem sind [dBHR93, Kapitel 2]. Außerdem sind die Box-Splines auf dem 4-Richtungs-Gitter nicht mehr stabil, d. h. sie erfüllen nicht die Riesz-Bedingung (1.3), vgl. [CJS94].

In den letzten Jahren gab es verschiedene Versuche, ein Splinesystem auf dem 4-Richtungs-Gitter zu finden, das die Basiseigenschaft erfüllt. Die ersten Konstruktionen eines solchen Splinepaares (ϕ_1, ϕ_2) , das den Raum $S_1^0(\Delta)$ generiert, finden sich in den schon erwähnten Arbeiten von Goodman et al. [GLT93, Goo97]. Bereits in [GLT93] berechnen Goodman, Lee und Tang auch ein dazugehöriges Prewavelet-Paar, das allerdings einen großen Träger besitzt und nicht explizit angegeben wird.

In [DGH00] wurde die Riesz-MRA, die von dem Splinepaar erzeugt wird, in eine ONB-MRA überführt. Dazu bediente man sich einer verflochtenen MRA (*intertwining MRA*). Allerdings werden die Approximationsräume nun nicht mehr nur von zwei, sondern von zehn Mutter-Skalierungsfunktionen generiert. Die zugehörigen Wavelets sind nicht betrachtet worden.

Floater und Quak [FQ01] nehmen die fehlende Orthogonalität der Basis-Knotenfunktionen von $S_1^0(\Delta)$ in Kauf und konstruieren auf elegante Weise eindeutige Prewavelets, die aus der Summe von zwei sogenannten „Semiwavelets“ entstehen (nicht zu verwechseln mit semiorthogonalen Wavelets!). Auch hier besteht der Nachteil in den verhältnismäßig großen Trägern der Funktionen.

Wir ziehen ebenso eine einfache Rieszbasis einer komplizierten ONB vor, und versuchen Prewavelets mit sehr kleinem Träger und mit einer vollständigen MRA-Struktur zu konstruieren, die auch in der Bildverarbeitung anwendbar sind. Dazu gehören insbesondere die

dualen Skalierungsfunktionen und Wavelets, deren Masken für den Zerlegungsalgorithmus der Filterbank notwendig sind. Dieser Aspekt ist in allen bisherigen Arbeiten für die $\Delta^{(2)}$ -Triangulierungen vollständig vernachlässigt worden, vermutlich aufgrund der großen Träger der Funktionen, was die Konstruktion von biorthogonalen Funktionen recht schwierig macht. Außerdem wird unser Waveletraum W_0 im Unterschied zu üblichen Waveletkonstruktionen $2\mathbb{Z}^2$ -translationsinvariant sein.

Der Vollständigkeit halber sei erwähnt, dass es auch Waveletkonstruktionen höherer Regularität auf $\Delta^{(2)}$ -Triangulierungen gibt. Spline-Wavelets quadratischer Ordnung beruhen auf dem bekannten Zwart-Powell-Element [Zwa73, Pow74], Wavelets kubischer Ordnung wurden z. B. in [BDG03] untersucht. Aber auch in diesen Fällen wurde unseres Wissens bisher nicht die duale MRA betrachtet.

2.5.1 Konstruktion eines linearen Splinepaares

Die Gitterlinien des in (2.44) definierten Gitters Δ schneiden sich in den Knoten $\mathbb{Z}^2 \cup (\mathbb{Z}^2 + (\frac{1}{2}, \frac{1}{2}))$. Zu diesen Knoten definieren wir die stetigen und auf den Dreiecksflächen $T \in \Delta$ linearen Knotenfunktionen, die an einem Knoten den Wert 1 annehmen und an allen anderen Gitterpunkten gleich Null sind. Sei

$$\phi_1(x, y) := \begin{cases} 2x, & x \leq y \leq 1-x, \\ 2y, & y \leq x, y \leq 1-x, \\ 2-2x, & 1-x \leq y \leq x, \\ 2-2y, & x \leq y, 1-x \leq y, \end{cases}$$

für $(x, y) \in [0, 1]^2$ und außerhalb $[0, 1]^2$ gleich Null. Wegen $\phi_1(1/2, 1/2) = 1$ ist die Pyramidenfunktion ϕ_1 die Knotenfunktion zum Knoten $(1/2, 1/2)$. Ihre \mathbb{Z}^2 -Translationen sind die Knotenfunktionen an den Gitterpunkten $\mathbb{Z}^2 + (\frac{1}{2}, \frac{1}{2})$. Weiter sei

$$\phi_2 := \phi_1 \left(Q^{-1} \left(\cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right) \right)$$

mit der Quincunx-Matrix $Q = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$ die Knotenfunktion an der Stelle $(0, 0)$, d. h. es gilt $\phi_2(0, 0) = 1$. Ihre Translationen sind die Knotenfunktionen an den Gitterpunkten \mathbb{Z}^2 .

Wie in Abbildung 2.17 dargestellt gilt $\text{supp } \phi_1 = \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}$ und damit auch

$$\begin{aligned} \text{supp } \phi_2 &= Q \text{supp } \phi_1 + \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\ &= \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix} \right\} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \text{conv} \left\{ \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}. \end{aligned}$$

Wir zeigen nun, dass dieses Splinepaar mit seinen \mathbb{Z}^2 -Translationen eine Basis

$$\mathcal{B} := \left\{ \phi_1(\cdot - k), \phi_2(\cdot - k) : k \in \mathbb{Z}^2 \right\}$$

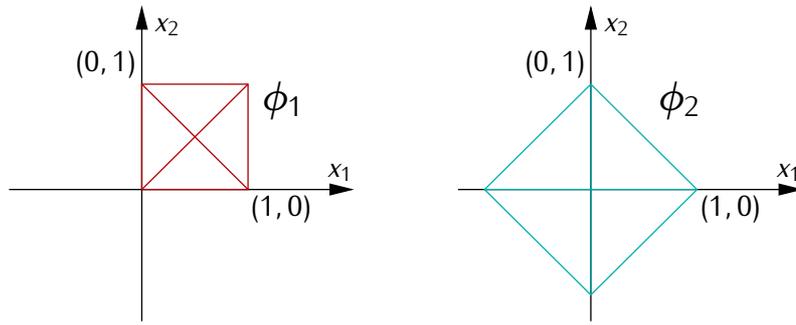


Abbildung 2.17: Träger der Spline-Skalierungsfunktionen ϕ_1 und ϕ_2 .

für den Raum $S_1^0(\Delta)$ erzeugt. Dazu beschränken wir uns zunächst auf die quadratischen Gebiete $[-1, 1]^2 + m$ für $m \in 2\mathbb{Z}^2$.

Satz 2.19. *Das System*

$$\mathcal{B}|_{[-1,1]^2+m} := \left\{ \phi|_{[-1,1]^2+m} : \phi \in \mathcal{B}, \text{supp } \phi \cap ((-1, 1)^2 + m) \neq \emptyset \right\}$$

ist eine Basis für $S_1^0(\Delta)|_{[-1,1]^2+m}$ für jedes $m \in 2\mathbb{Z}^2$.

Beweis. 1. Erzeugendensystem: Es ist klar, dass eine auf $T \in \Delta$ lineare Funktion als Linearkombination der drei Knotenfunktionen dargestellt werden kann, die an jeweils einem Eckpunkt von T gleich 1 sind und an den anderen beiden Eckpunkten verschwinden. Wie oben erwähnt, liegen die Eckpunkte der Dreiecke $T \in \Delta$ auf dem Knotengitter $\mathbb{Z}^2 \cup (\mathbb{Z}^2 + (\frac{1}{2}, \frac{1}{2}))$, dessen Knotenfunktionen nach Konstruktion die Translationen der Splinefunktionen ϕ_1 und ϕ_2 sind.

2. Lineare Unabhängigkeit: Sei o.B.d.A. $m = 0$. Es gilt die Dimensionsformel $\dim S_1^0(\Delta)|_{[-1,1]^2} = 13$ nach [Chu88]. Wir überlegen uns jetzt, welche Funktionen $\phi \in \mathcal{B}$ einen Träger haben, der eine nichtleere Schnittmenge mit $(-1, 1)^2$ besitzt. Das sind genau 13 Funktionen, und zwar $\phi_1(\cdot - k)$ für $k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}$, sowie $\phi_2(\cdot - k)$ für $k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}$. Daraus folgt die lineare Unabhängigkeit. \square

Nun verlassen wir das beschränkte Gebiet $[-1, 1]^2$ und betrachten die komplette Ebene des \mathbb{R}^2 .

Satz 2.20. *Das System $\mathcal{B} = \{ \phi_1(\cdot - k), \phi_2(\cdot - k) : k \in \mathbb{Z}^2 \}$ ist eine Rieszbasis des $S_1^0(\Delta)$.*

Beweis. Sei

$$f = \sum_{k \in \mathbb{Z}^2} \alpha_k^1 \phi_1(\cdot - k) + \alpha_k^2 \phi_2(\cdot - k)$$

mit $\|\alpha\|_2^2 := \sum_{k \in \mathbb{Z}^2, i=1,2} |\alpha_k^i|^2 < \infty$. Weil $\mathbb{R}^2 = \bigcup_{m \in 2\mathbb{Z}^2} [-1, 1]^2 + m$ eine disjunkte Zerlegung von \mathbb{R}^2 ist, gilt

$$\|f\|_2^2 = \int_{\mathbb{R}^2} |f|^2 = \sum_{m \in 2\mathbb{Z}^2} \int_{[-1,1]^2+m} |f|^2 = \sum_{m \in 2\mathbb{Z}^2} \int_{\mathbb{R}^2} |f|_{[-1,1]^2+m}|^2 = \sum_{m \in 2\mathbb{Z}^2} \|f|_{[-1,1]^2+m}\|_2^2. \quad (2.45)$$

Nach Satz 2.19 existiert eine Rieszbasis für $S_1^0(\Delta)|_{[-1,1]^2+m}$, $m \in 2\mathbb{Z}^2$, d. h.

$$C_1 \|\alpha_m\|_{l^2}^2 \leq \|f|_{[-1,1]^2+m}\|_2^2 \leq C_2 \|\alpha_m\|_{l^2}^2 \quad \forall f|_{[-1,1]^2+m} \in S_1^0(\Delta)|_{[-1,1]^2+m}, \quad (2.46)$$

wobei $f|_{[-1,1]^2+m} = \sum_{k \in \mathbb{Z}^2} \alpha_{m,k}^1 \phi_1(\cdot - k) + \alpha_{m,k}^2 \phi_2(\cdot - k)$ mit $\|\alpha_m\|_{l^2}^2 := \sum_{k \in \mathbb{Z}^2, i=1,2} |\alpha_{m,k}^i|^2 < \infty$. Außerdem sind die Konstanten C_1 und C_2 von m unabhängig. Wir summieren jetzt (2.46) über $m \in 2\mathbb{Z}^2$ und erhalten dann mit Hilfe von (2.45) für alle $f \in S_1^0(\Delta)$

$$C_1 \sum_{m \in 2\mathbb{Z}^2} \|\alpha_m\|_{l^2}^2 \leq \|f\|_2^2 \leq C_2 \sum_{m \in 2\mathbb{Z}^2} \|\alpha_m\|_{l^2}^2.$$

Weil die Translationen von ϕ_2 einen Träger besitzen, der eine nichtleere Schnittmenge mit höchstens vier Quadraten $[-1,1]^2 + m$ aufweist, gilt

$$C_1 \|\alpha\|_{l^2}^2 \leq \|f\|_2^2 \leq 4C_2 \|\alpha\|_{l^2}^2.$$

□

Die beiden Splinefunktionen sollen uns als Skalierungsfunktionen dienen, deshalb müssen wir ihre Verfeinerbarkeit sicherstellen. Wählen wir wieder die Dilatationsmatrix $A = 2I_2$, so finden wir

$$\phi_1 = \frac{1}{2} \left(\phi_1(2 \cdot) + \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \right) + \phi_2(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}). \quad (2.47)$$

und

$$\phi_2 = \frac{1}{2} \left(\phi_2(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) \right) + \phi_1(\cdot - \begin{pmatrix} -1/2 \\ -1/2 \end{pmatrix}).$$

Unter Verwendung von (2.47) können wir den letzten Teil der Verfeinerungsgleichung weiter umformen zu

$$\begin{aligned} \phi_2 &= \frac{1}{2} \left(\phi_2(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) \right) \\ &\quad + \frac{1}{2} \left(\phi_1(2 \cdot - \begin{pmatrix} -1 \\ -1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + \phi_1(2 \cdot) \right) + \phi_2(2 \cdot). \end{aligned} \quad (2.48)$$

Die skalierten Splinefunktionen $\phi_1(2 \cdot - k)$, $\phi_2(2 \cdot - k)$, $k \in \mathbb{Z}^2$, sind linear auf dem verfeinerten Gitter $\frac{1}{2}\Delta$, das durch

$$x = i/2, \quad y = i/2, \quad x + y = i/2, \quad x - y = i/2 \quad (i \in \mathbb{Z})$$

erzeugt wird. Dann ist nun $\{\phi_1(2 \cdot - k), \phi_2(2 \cdot - k) : k \in \mathbb{Z}^2\}$ eine Rieszbasis von $S_1^0(\frac{1}{2}\Delta)$. Wählen wir

$$V_j := S_1^0\left(\frac{1}{2^j}\Delta\right),$$

so ist $\{V_j\}_{j \in \mathbb{Z}}$ eine Riesz-MRA für den $L^2(\mathbb{R}^2)$.

Da \mathcal{B} eine Rieszbasis von $S_1^0(\Delta)$ ist, existiert eine Menge

$$\tilde{\mathcal{B}} = \left\{ \tilde{\phi}_1(\cdot - k), \tilde{\phi}_2(\cdot - k) : k \in \mathbb{Z}^2 \right\}$$

von Splinefunktionen, die zu \mathcal{B} dual sind, d. h. es gilt

$$\langle \phi_i(\cdot - k), \tilde{\phi}_{i'}(\cdot - k') \rangle = \delta_{i,i'} \delta_{k,k'} \|\phi_i\|_2^2. \quad (2.49)$$

Damit ist jedes $f \in S_1^0(\Delta)$ darstellbar als

$$f = \sum_{\substack{i=1,2 \\ k \in \mathbb{Z}^2}} \langle f, \tilde{\phi}_i(\cdot - k) \rangle \phi_i(\cdot - k).$$

Zur Bestimmung der dualen Funktionen treffen wir weitere Annahmen. Zunächst wählen wir

$$\tilde{\mathcal{B}} \subset S_1^0\left(\frac{1}{2}\Delta\right) = V_1. \quad (2.50)$$

Weil $S_1^0(\frac{1}{2}\Delta)$ sehr viel größer als $S_1^0(\Delta)$ ist, kann $\tilde{\mathcal{B}}$ nicht eindeutig bestimmt werden. Deshalb wählen wir als zusätzliche Voraussetzungen an die biorthogonalen Funktionen die Trägereigenschaft

$$\text{supp } \tilde{\phi}_i \subseteq \text{supp } \phi_i \quad i = 1, 2. \quad (2.51)$$

Dadurch erzwingen wir kleine Träger, was zu kleinen Filtermasken in der Zerlegungs-Filterbank führt. Mit den beiden Bedingungen (2.49) und (2.51) können wir die dualen Funktionen aus dem Ansatz

$$\tilde{\phi}_i = \sum_{k \in \mathbb{Z}^2} \tilde{a}_k^i \phi_1(2 \cdot - k) + \sum_{k \in \mathbb{Z}^2} \tilde{b}_k^i \phi_2(2 \cdot - k) \quad (2.52)$$

bestimmen, indem wir ein lineares Gleichungssystem für die Koeffizienten \tilde{a}_k^i und \tilde{b}_k^i aufstellen und lösen. Und zwar erhalten wir (siehe Anhang B.2) schließlich

$$\tilde{\phi}_1 = \frac{5}{2}\phi_2(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) - \frac{3}{4} \left(\phi_1(2 \cdot) + \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \right) \quad (2.53)$$

$$\begin{aligned} \tilde{\phi}_2 = & \frac{7}{2}\phi_2(2 \cdot) - \frac{1}{4} \left(\phi_2(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + \phi_2(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) \right) \\ & + \phi_1(2 \cdot - \begin{pmatrix} -1 \\ -1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) + \phi_1(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + \phi_1(2 \cdot). \end{aligned} \quad (2.54)$$

Die beiden dualen Funktionen sind in Abbildung 2.18 dargestellt. Es gilt $\text{supp } \tilde{\phi}_i = \text{supp } \phi_i$ für $i = 1, 2$. Vergleichen wir die beiden Formulierungen für $\tilde{\phi}_1$ und $\tilde{\phi}_2$ mit den Verfeinerungsgleichungen (2.47) und (2.48) von ϕ_1 und ϕ_2 , so erhalten wir sofort die kompakte Darstellung

$$\tilde{\phi}_1 = 4\phi_2(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) - \frac{3}{2}\phi_1, \quad (2.55)$$

$$\tilde{\phi}_2 = 4\phi_2(2 \cdot) - \frac{1}{2}\phi_2. \quad (2.56)$$

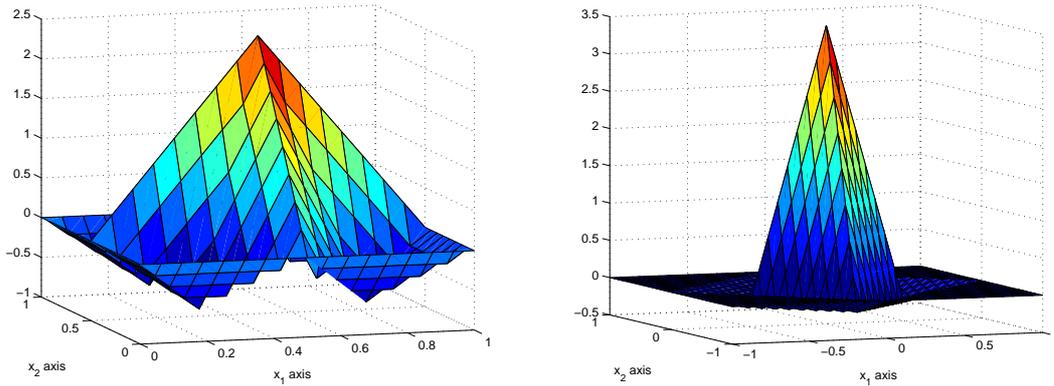


Abbildung 2.18: Die biorthogonalen Splinefunktionen $\tilde{\phi}_1$ und $\tilde{\phi}_2$.

Beachte, dass $\tilde{\phi}_i \notin V_0$ gilt. Die Problematik dieses Ansatzes besteht darin, dass diese dualen Funktionen keine biorthogonale MRA $\{\tilde{V}_j\}_{j \in \mathbb{Z}}$ generieren, denn die Approximationsräume

$$\tilde{V}_j := \text{clos}_{L^2(\mathbb{R}^2)} \text{span}\{\tilde{\phi}_i(\cdot - k) : i = 1, 2; k \in \mathbb{Z}^2\}$$

sind nicht ineinander geschachtelt, d. h. es gilt nicht $\tilde{V}_j \subset \tilde{V}_{j+1}$. Mit anderen Worten: Die beiden dualen Funktionen $\tilde{\phi}_1$ und $\tilde{\phi}_2$ sind nicht verfeinerbar! Dieses Problem ist auch in [BDG03] ungelöst geblieben. Trotzdem können wir uns der Konstruktion der Wavelets zuwenden.

2.5.2 Konstruktion der linearen Spline-Wavelets

Wir verfolgen den üblichen Weg, zunächst eine Waveletbasis für den Differenzenraum W_0 zu konstruieren. Danach lassen sich die weiteren Waveleträume W_j leicht durch eine Dilatation der erhaltenen Basisfunktionen ableiten.

Dazu suchen wir einen Raum W_0 , so dass die direkte (nicht orthogonale) Summe $V_0 + W_0 = V_1$ sowie die Bedingung $W_0 \perp \tilde{V}_0$ erfüllt ist. Zunächst betrachten wir die erste Forderung

$$W_0 = V_1 - V_0 = S_1^0\left(\frac{1}{2}\Delta\right) - S_1^0(\Delta).$$

Sei nun $f \in V_1$ gegeben, so kann f dargestellt werden als

$$f = \sum_{\substack{i=1,2 \\ k \in \mathbb{Z}^2}} \frac{\langle f, \tilde{\phi}_i(2 \cdot - k) \rangle}{\|\phi_i(2 \cdot)\|_2^2} \phi_i(2 \cdot - k). \quad (2.57)$$

Die Projektion $P_{V_0} f \in V_0$ ist dann

$$\begin{aligned} P_{V_0} f &= \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle f, \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k') \\ &= \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \sum_{\substack{i=1,2 \\ k \in \mathbb{Z}^2}} \frac{\langle f, \tilde{\phi}_i(2 \cdot - k) \rangle}{\|\phi_i(2 \cdot)\|_2^2} \frac{\langle \phi_i(2 \cdot - k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k'). \end{aligned} \quad (2.58)$$

Nun soll die Differenz $f - P_{V_0}f$ mit Wavelets dargestellt werden, wir subtrahieren also (2.58) von (2.57) und erhalten

$$f - P_{V_0}f = \sum_{\substack{i=1,2 \\ k \in \mathbb{Z}^2}} \frac{\langle f, \tilde{\phi}_i(2 \cdot -k) \rangle}{\|\phi_i(2 \cdot)\|_2^2} \left\{ \phi_i(2 \cdot -k) - \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k') \right\}.$$

Die Funktionen in dem Klammerausdruck sind also Erzeugendenfunktionen von W_0 . Wir definieren für $k \in \mathbb{Z}^2$ und $i = 1, 2$ die gesuchten Wavelets

$$\psi_{i,k} := \phi_i(2 \cdot -k) - \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k'), \quad (2.59)$$

wobei wir sehen werden, dass die Wavelets nicht wie gewöhnlich \mathbb{Z}^2 -translationsinvariant, sondern $2\mathbb{Z}^2$ -translationsinvariant sind. Die Wavelets erfüllen gemäß Konstruktion auch die zweite Bedingung $W_0 \perp \tilde{V}_0$, denn für $s = 1, 2$ und $l \in \mathbb{Z}^2$ gilt

$$\begin{aligned} \langle \tilde{\phi}_s(\cdot - l), \psi_{i,k} \rangle &= \langle \tilde{\phi}_s(\cdot - l), \phi_i(2 \cdot -k) \rangle - \langle \tilde{\phi}_s(\cdot - l), \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k') \rangle \\ &= \langle \tilde{\phi}_s(\cdot - l), \phi_i(2 \cdot -k) \rangle - \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \langle \tilde{\phi}_s(\cdot - l), \phi_{i'}(\cdot - k') \rangle \\ &= \langle \tilde{\phi}_s(\cdot - l), \phi_i(2 \cdot -k) \rangle - \langle \phi_i(2 \cdot -k), \tilde{\phi}_s(\cdot - l) \rangle = 0. \end{aligned}$$

Aus der Definition der Wavelets (2.59) folgen sofort die Rekonstruktionsformeln für $k \in \mathbb{Z}^2$ und $i = 1, 2$:

$$\phi_i(2 \cdot -k) = \psi_{i,k} + \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k').$$

Dabei ist die Summe endlich, sie besitzt maximal neun Nichtnull-Koeffizienten, wie wir im Folgenden sehen werden.

Die durch (2.59) definierten Wavelets haben für verschiedene Translationen unterschiedliche Darstellungen, weil das Skalarprodukt $\langle \phi_i(2 \cdot -k), \tilde{\phi}_{i'}(\cdot - k') \rangle$ in Abhängigkeit von der Translation $k \in \mathbb{Z}^2$ verschiedene Werte liefert und sogar häufig verschwindet. Konkret können wir zwischen folgenden Translationen unterscheiden (die Herleitung ist im Anhang B.3 zu finden):

$$(a) k = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + 2\mathbb{Z}^2$$

$$\begin{aligned} \psi_{1,k} = \phi_1(2 \cdot -k) + \frac{1}{32} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{2}\right) - \frac{1}{16} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ 0 \end{pmatrix}}{2}\right) + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{2}\right) \\ + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ -2 \end{pmatrix}}{2}\right) \end{aligned}$$

$$\begin{aligned} \psi_{2,k} = \phi_2(2 \cdot -k) - \frac{1}{16} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{2}\right) - \frac{1}{16} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 2 \end{pmatrix}}{2}\right) - \frac{1}{32} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{2}\right) \\ - \frac{1}{32} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ 0 \end{pmatrix}}{2}\right) \end{aligned}$$

$$(b) k = \begin{pmatrix} 0 \\ 1 \end{pmatrix} + 2\mathbb{Z}^2$$

$$\begin{aligned} \psi_{1,k} = \phi_1(2 \cdot -k) + \frac{1}{32} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 1 \end{pmatrix}}{2}\right) - \frac{1}{16} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ -1 \end{pmatrix}}{2}\right) + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 1 \end{pmatrix}}{2}\right) \\ + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -2 \\ -1 \end{pmatrix}}{2}\right) \end{aligned}$$

$$\begin{aligned} \psi_{2,k} = \phi_2(2 \cdot -k) - \frac{1}{16} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 1 \end{pmatrix}}{2}\right) - \frac{1}{16} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 2 \\ 1 \end{pmatrix}}{2}\right) - \frac{1}{32} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 1 \end{pmatrix}}{2}\right) \\ - \frac{1}{32} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ -1 \end{pmatrix}}{2}\right) \end{aligned}$$

$$(c) k = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 2\mathbb{Z}^2$$

$$\begin{aligned} \psi_{1,k} = \phi_1(2 \cdot -k) + \frac{1}{32} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{2}\right) - \frac{1}{16} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ 1 \end{pmatrix}}{2}\right) + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{2}\right) \\ + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ 1 \end{pmatrix}}{2}\right) \end{aligned}$$

$$\begin{aligned} \psi_{2,k} = \phi_2(2 \cdot -k) - \frac{17}{16} \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{2}\right) + \frac{3}{128} \left(\phi_2\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{2}\right) + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 1 \\ -1 \end{pmatrix}}{2}\right) \right. \\ \left. + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ 1 \end{pmatrix}}{2}\right) + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -1 \\ -1 \end{pmatrix}}{2}\right) \right) \end{aligned}$$

$$(d) k \in 2\mathbb{Z}^2$$

$$\begin{aligned} \psi_{1,k} = \phi_1(2 \cdot -k) + \frac{1}{32} \phi_1\left(\cdot - \frac{k}{2}\right) - \frac{1}{16} \phi_2\left(\cdot - \frac{k}{2}\right) + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -2 \\ 0 \end{pmatrix}}{2}\right) \\ + \frac{1}{128} \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ -2 \end{pmatrix}}{2}\right) \end{aligned}$$

$$\begin{aligned} \psi_{2,k} = \phi_2(2 \cdot -k) + \frac{3}{64} \left(\phi_1\left(\cdot - \frac{k}{2}\right) + \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 2 \\ 0 \end{pmatrix}}{2}\right) + \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 2 \end{pmatrix}}{2}\right) + \phi_1\left(\cdot - \frac{k - \begin{pmatrix} 2 \\ 2 \end{pmatrix}}{2}\right) \right) \\ + \frac{1}{128} \left(\phi_2\left(\cdot - \frac{k - \begin{pmatrix} 2 \\ 0 \end{pmatrix}}{2}\right) + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ 2 \end{pmatrix}}{2}\right) + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} -2 \\ 0 \end{pmatrix}}{2}\right) + \phi_2\left(\cdot - \frac{k - \begin{pmatrix} 0 \\ -2 \end{pmatrix}}{2}\right) \right) - \frac{13}{16} \phi_2\left(\cdot - \frac{k}{2}\right) \end{aligned}$$

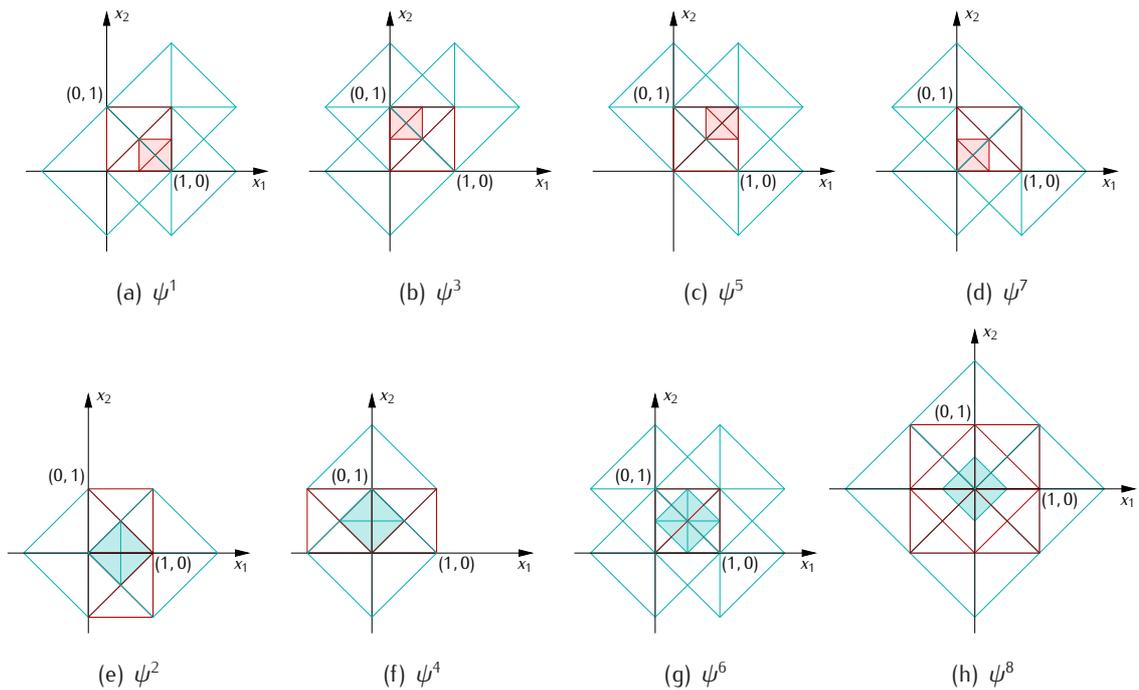


Abbildung 2.19: Die Träger der Mutterwavelets.

Diese acht Waveletfunktionen lassen sich als Mutterwavelets auffassen. Bezüglich der Translationen $k \in 2\mathbb{Z}^2$ können wir diese Funktionen wie folgt umschreiben:

$$\begin{aligned} \psi^1 &:= \psi_{1, \begin{pmatrix} 1 \\ 0 \end{pmatrix}}, & \psi^2 &:= \psi_{2, \begin{pmatrix} 1 \\ 0 \end{pmatrix}}, \\ \psi^3 &:= \psi_{1, \begin{pmatrix} 0 \\ 1 \end{pmatrix}}, & \psi^4 &:= \psi_{2, \begin{pmatrix} 0 \\ 1 \end{pmatrix}}, \\ \psi^5 &:= \psi_{1, \begin{pmatrix} 1 \\ 1 \end{pmatrix}}, & \psi^6 &:= \psi_{2, \begin{pmatrix} 1 \\ 1 \end{pmatrix}}, \\ \psi^7 &:= \psi_{1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}}, & \psi^8 &:= \psi_{2, \begin{pmatrix} 0 \\ 0 \end{pmatrix}}. \end{aligned}$$

Die Träger von ψ^l , $l = 1, \dots, 8$, sind in Abbildung 2.19 dargestellt. Dass

$$\mathcal{D}_0^* := \{\psi^l(\cdot - k) : l = 1, \dots, 8; k \in 2\mathbb{Z}^2\}$$

ein Erzeugendensystem des Differenzraums W_0 bildet, ist klar, weil wir die Wavelets in (2.59) genau so definiert hatten. Wir werden aber feststellen, dass \mathcal{D}_0^* ein redundantes System ist. Um eine Basis von W_0 zu erhalten, muss \mathcal{D}_0^* passend verkleinert werden.

Wir beschränken uns zunächst wieder auf das Gebiet $[-1, 1]^2$, indem wir

$$W_0|_{[-1, 1]^2} := V_1|_{[-1, 1]^2} - V_0|_{[-1, 1]^2} = S_1^0\left(\frac{1}{2}\Delta\right)|_{[-1, 1]^2} - S_1^0(\Delta)|_{[-1, 1]^2}$$

betrachten. Dabei ist $W_0|_{[-1, 1]^2} + V_0|_{[-1, 1]^2} = V_0|_{[-1, 1]^2}$ eine direkte (aber nicht orthogonale) Summe.

Satz 2.21. *Das System*

$$\mathcal{B}^1|_{[-1,1]^2} = \{\phi_i(2 \cdot -k)|_{[-1,1]^2} : i = 1, 2; k \in \mathbb{Z}^2, \text{supp } \phi_i(2 \cdot -k) \cap (-1, 1)^2 \neq \emptyset\}$$

ist eine Rieszbasis des $S_1^0(\frac{1}{2}\Delta)|_{[-1,1]^2}$.

Beweis. Der Beweis verläuft analog zu Satz 2.19. Die Eigenschaft des Erzeugendensystems ist offensichtlich. Wir zählen die Funktionen mit einem nicht-verschwindenden Träger im Gebiet $[-1, 1]^2$, dies sind genau 41 Stück. Weil nach Chuis Dimensionsformel $\dim S_1^0(\frac{1}{2}\Delta)|_{[-1,1]^2} = 41$ gilt (siehe [Chu88]), müssen diese 41 Funktionen linear unabhängig sein. \square

Wegen $\dim W_0|_{[-1,1]^2} = \dim V_1|_{[-1,1]^2} - \dim V_0|_{[-1,1]^2} = 41 - 13 = 28$ suchen wir ein Waveletsystem $\mathcal{D}_0|_{[-1,1]^2}$, das 28 Funktionen enthält und eine Rieszbasis von $W_0|_{[-1,1]^2}$ bildet. Wenn wir \mathcal{D}_0^* auf das Gebiet $[-1, 1]^2$ einschränken, stellen wir fest, dass

$$\mathcal{D}_0^*|_{[-1,1]^2} := \{\psi^l(\cdot - k)|_{[-1,1]^2} : l = 1, \dots, 8; k \in 2\mathbb{Z}^2, \text{supp } \phi_{l \bmod 2}(2 \cdot -k) \cap (-1, 1)^2 \neq \emptyset\} \quad (2.60)$$

41 Funktionen enthält. Beachte dabei die Trägerbedingung für die Funktionen in $\mathcal{D}_0^*|_{[-1,1]^2}$: Es genügt nicht, dass $\text{supp } \psi^l(\cdot - k) \cap (-1, 1)^2 \neq \emptyset$ gilt, denn jedes $\psi^l(\cdot - k)$ besteht nach Definition aus einer Linearkombination von genau einer Funktion $\phi_{l \bmod 2}(2 \cdot -k) \in V_1$ und mehreren Funktionen $\phi_i(\cdot - k) \in V_0$, $i = 1, 2$. Wenn nun zwar $\text{supp } \psi^l(\cdot - k) \cap (-1, 1)^2 \neq \emptyset$ gelten würde, aber nicht $\text{supp } \phi_{l \bmod 2}(2 \cdot -k) \cap (-1, 1)^2 \neq \emptyset$, so würde das abgeschnittene Wavelet auf dem Gebiet $[-1, 1]^2$ in $V_0|_{[-1,1]^2}$ und nicht in $W_0|_{[-1,1]^2}$ enthalten sein.

Wegen $|\mathcal{D}_0^*|_{[-1,1]^2}| = 41$ ist das System also redundant. Deshalb entfernen wir 13 Funktionen, so dass eine Basis übrig bleibt. Dazu verwerfen wir ψ^7 und ψ^8 mit ihren Translationen. Wir erhalten schließlich folgenden Satz.

Satz 2.22. *Das Waveletsystem*

$$\mathcal{D}_0|_{[-1,1]^2} := \{\psi^l(\cdot - k)|_{[-1,1]^2} : l = 1, \dots, 6; k \in 2\mathbb{Z}^2, \text{supp } \phi_{l \bmod 2}(2 \cdot -k) \cap (-1, 1)^2 \neq \emptyset\}$$

ist eine Rieszbasis für $W_0|_{[-1,1]^2}$.

Beweis. Die beiden Mutterwavelets $\psi^7(\cdot - k)$ und $\psi^8(\cdot - k)$ sind für $k \in 2\mathbb{Z}^2$ linear abhängig von den übrigen Wavelets. Es gilt nämlich

$$\psi^7 = -2\psi^6 - \{\psi^1 + \psi^3 + \psi^5\} \quad (2.61)$$

für $k \in 2\mathbb{Z}^2$ sowie

$$\begin{aligned} \psi^8 = \psi^6 - \frac{1}{2} \left\{ \psi^5(\cdot - \begin{pmatrix} -2 \\ -2 \end{pmatrix}) + \psi^3(\cdot - \begin{pmatrix} 0 \\ -2 \end{pmatrix}) + \psi^1(\cdot - \begin{pmatrix} -2 \\ 0 \end{pmatrix}) - \psi^1 \right. \\ \left. - \psi^3 - \psi^5 + \psi^2 + \psi^4 + \psi^2(\cdot - \begin{pmatrix} -2 \\ 0 \end{pmatrix}) + \psi^4(\cdot - \begin{pmatrix} 0 \\ -2 \end{pmatrix}) \right\}. \end{aligned} \quad (2.62)$$

Entfernen wir also diese Funktionen, so ist das verbleibende Waveletsystem $\mathcal{D}_0|_{[-1,1]^2}$ immer noch ein Erzeugendensystem von $W_0|_{[-1,1]^2}$. Es verbleibt zu zeigen, dass keine Redundanzen

mehr vorhanden sind. Das folgt wieder aus dem Dimensionsargument. Denn wir haben 13 Wavelets aus $\mathcal{D}_0^*|_{[-1,1]^2}$ entfernt, und zwar $\psi^7(\cdot - k)$ mit $k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \begin{pmatrix} -2 \\ -2 \end{pmatrix}$, sowie $\psi^8(\cdot - k)$ mit $k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \begin{pmatrix} -2 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ -2 \end{pmatrix}, \begin{pmatrix} -2 \\ -2 \end{pmatrix}$. Das überlegt man sich leicht, wenn man die beiden entsprechenden Träger-Figuren in der letzten Spalte von Abbildung 2.19 betrachtet. Die übrigen $41 - 13 = 28$ Funktionen bilden dann wegen $\dim W_0|_{[-1,1]^2} = 28$ ein minimales Erzeugendensystem. \square

In Analogie zu Satz 2.19 und 2.20 kann auch hier wieder eine Verallgemeinerung von dem Gebiet $[-1, 1]^2$ auf die gesamte Ebene \mathbb{R}^2 vollzogen werden.

Satz 2.23. *Das Waveletsystem*

$$\mathcal{D}_0 := \{\psi^l(\cdot - k) : l = 1, \dots, 6; k \in 2\mathbb{Z}^2\}$$

ist eine Rieszbasis für W_0 .

Bemerkung 2.24. *Statt die Wavelets ψ^7 und ψ^8 , $k \in 2\mathbb{Z}^2$, zu entfernen, kann man auch andere Wavelets in geeigneter Weise aus $\mathcal{D}_0^*|_{[-1,1]^2}$ entfernen. Es ist z. B. sinnvoll sich auf Wavelets zu beschränken, die möglichst kleine und gerichtete Träger haben. Die letzten beiden Figuren in Abbildung 2.19 machen klar, dass ein Entfernen der Wavelets $\psi^6(\cdot - k)$ und $\psi^8(\cdot - k)$, $k \in 2\mathbb{Z}^2$, sinnvoll wäre. Anders ausgedrückt: Statt auf ψ^7 zu verzichten, wollen wir lieber die Funktion ψ^6 samt ihren Translationen verwerfen. Die lineare Abhängigkeit dieser Wavelets folgt sofort aus der Umstellung von (2.61) zu*

$$\psi^6 = -\frac{1}{2} \{\psi^1 + \psi^3 + \psi^5 + \psi^7\}$$

für $k \in 2\mathbb{Z}^2$. Damit nimmt (2.62) jetzt die vereinfachte Gestalt an:

$$\begin{aligned} \psi^8 = -\frac{1}{2} \left\{ \psi^7 + \psi^5(\cdot - \begin{pmatrix} -2 \\ -2 \end{pmatrix}) + \psi^3(\cdot - \begin{pmatrix} 0 \\ -2 \end{pmatrix}) + \psi^1(\cdot - \begin{pmatrix} -2 \\ 0 \end{pmatrix}) \right. \\ \left. + \psi^2 + \psi^4 + \psi^2(\cdot - \begin{pmatrix} -2 \\ 0 \end{pmatrix}) + \psi^4(\cdot - \begin{pmatrix} 0 \\ -2 \end{pmatrix}) \right\}. \end{aligned}$$

Wir haben also mittels einer geeigneten Basistransformation eine alternative Rieszbasis

$$\{\psi^l(\cdot - k) : l = 1, 2, 3, 4, 5, 7; k \in 2\mathbb{Z}^2, \text{supp } \phi_{l \bmod 2}(2 \cdot -k) \cap (-1, 1)^2 \neq \emptyset\}$$

für $W_0|_{[-1,1]^2}$ gefunden.

In einem zweiten Schritt können wir diese Basis weiter modifizieren, indem wir Differenzen und Summen der vier Mutterwavelets ψ^1, ψ^3, ψ^5 und ψ^7 für $k \in 2\mathbb{Z}^2$ betrachten. Dadurch erhalten wir symmetrische Träger. Wir definieren

$$\psi^{(1)} := \psi^1 - \psi^3 = \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) - \phi_1(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) - \frac{1}{16} \phi_2(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + \frac{1}{16} \phi_2(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}),$$

für $k \in 2\mathbb{Z}^2$. Der entsprechende Träger ist in Abbildung 2.20(a) zu sehen. (Vergleiche mit den Trägern in Abbildung 2.19(a) und (b).) Analog erhalten wir die weiteren Wavelets

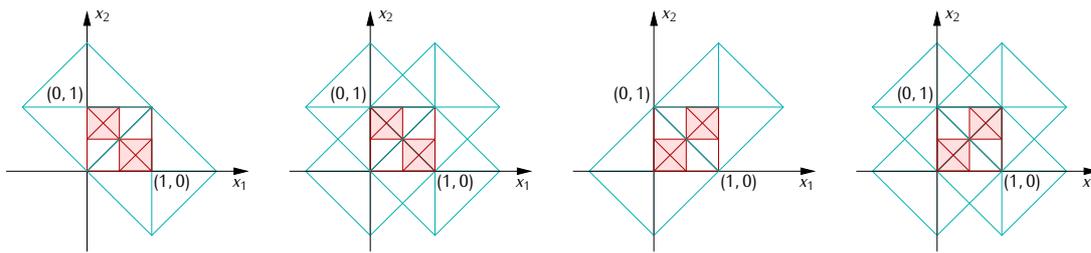


Abbildung 2.20: Die Träger von $\psi^{(i)}$ für $i = 1, \dots, 4$.

$$\psi^{(2)} := \psi^1 + \psi^3, \quad \psi^{(3)} := \psi^5 + \psi^7, \quad \psi^{(4)} := \psi^5 - \psi^7,$$

siehe Abbildung 2.20(b)-(d). Die beiden Mutterwavelets ψ^2 und ψ^4 lassen wir unverändert:

$$\psi^{(5)} := \psi^2, \quad \psi^{(6)} := \psi^4.$$

Dann bildet auch

$$\{\psi^{(i)}(\cdot - k) : i = 1, \dots, 6; k \in 2\mathbb{Z}^2\}$$

eine Rieszbasis für W_0 .

Kapitel 3

Adaptive Tetrolet-Transformation

Die Tetrolet-Transformation wurde in dem aktuellen Paper [Kro10] zur Approximation von Bildern eingeführt. Im Rahmen der vorliegenden Arbeit werden die dort dargestellten Ergebnisse in diesem Kapitel noch einmal ausführlich erläutert.

Die adaptive Tetrolet-Transformation ist eine Haarwavelet-Transformation (HWT) auf geometrischen Formen, auf sogenannten *Tetrominoes*, die die Bildgeometrie berücksichtigen. Auf diese Weise verbindet sie die Effizienz einer Haarwavelet-Filterbank mit der Approximationsqualität der adaptiven Methoden. Ihr großer Vorteil ist, dass sie durch einen einfachen Zerlegungsalgorithmus unmittelbar im Diskreten definiert werden kann. Dadurch entfällt der in der Bildverarbeitung notwendige aber oftmals sehr lästige Schritt der Projektion in den diskreten Bildraum. In Kapitel 2.1.2 und 2.2.3 haben wir diese Problematik bei der Anwendung der gerichteten Wavelets auf Dreiecken untersucht: Wenn man die volle Richtungssensitivität der kontinuierlichen Wavelets in den diskreten Fall übertragen möchte, handelt man sich bei der Projektion und der damit häufig verbundenen Interpolation Redundanzen ein, die bei der Approximation bzw. Kompression von Bildern hinderlich sind.

Die der Tetrolet-Transformation zugrunde liegende Idee ist sehr ähnlich zu der Konstruktion der Wedgelets [Don99]. Dies sind charakteristische Funktionen auf keilförmigen Figuren (wedge = Keil). Bei der Wedgelet-Transformation wird das (kontinuierliche) Bildgebiet zunächst in dyadische Quadrate zerlegt, die dann entsprechend der Bildgeometrie durch einen Schnitt weiter unterteilt werden, so dass Keilformen entstehen, auf denen die Wedgelets als charakteristische Funktionen definiert werden. Bei der Tetrolet-Transformation zerlegen wir das diskrete Bildgebiet in 4×4 -Blöcke und berechnen in jedem Block eine Tetromino-Zerlegung, die der lokalen Bildgeometrie angepasst ist. Tetrominoes sind Figuren, die aus vier Einheitsquadraten bestehen, die durch mindestens eine gemeinsame Kante miteinander verbunden sind. Auf diesen Tetromino-Trägern werden Haarwavelets definiert, die wir *Tetrolets* nennen, und die eine lokale ONB bilden. Der dazugehörige, schnelle Filterbank-Algorithmus zerlegt ein Bild in eine dünne Darstellung mit möglichst wenig Tetrolet-Koeffizienten. Die

kleinen Träger der Tetrolets bewirken neben einer schnellen Zerlegung auch eine Bilddarstellung, die nahezu frei von Pseudo-Gibbs-Artefakten ist.

Zu Beginn des Kapitels legen wir eine Notation fest und erläutern dann im zweiten Abschnitt die Idee der Tetrolets ausführlich. Anschließend geben wir eine detaillierte Beschreibung des Filterbank-Algorithmus an. Die Adaptivität der Methode bringt zusätzliche Speicherkosten mit sich, die wir im Unterkapitel 3.6 untersuchen. Danach beschreiben wir die numerische Komplexität und illustrieren die Effizienz der Tetrolet-Transformation anhand von verschiedenen Testbildern. Schließlich widmen wir uns zwei wichtigen theoretischen Resultaten, indem wir eine adaptive MRA-Struktur sowie das asymptotische Abklingverhalten des Approximationsfehlers der Tetrolet-Approximation beweisen.

3.1 Notation

Aus Gründen der Übersichtlichkeit beschränken wir uns wieder auf quadratische Bilder. Sei $I = \{(m_1, m_2) : m_1, m_2 = 0, \dots, N-1\} \subset \mathbb{Z}^2$ die *Indexmenge* eines digitalen Bildes $\mathbf{a} = (a[m_1, m_2])_{(m_1, m_2) \in I}$ mit $N = 2^J, J \in \mathbb{N}$. Die *4er-Nachbarschaft* eines Index $(m_1, m_2) \in I$ ist die Menge

$$N_4(m_1, m_2) := \{(m_1 - 1, m_2), (m_1 + 1, m_2), (m_1, m_2 - 1), (m_1, m_2 + 1)\}.$$

Ein Index, der am Rand des Bildes liegt, hat drei Nachbarn; die vier Indizes in den Ecken haben nur zwei Nachbarn. Im Verlauf des Kapitels wird es für manche Untersuchungen von Vorteil sein, eine eindimensionale Indizierung der Bildpixel vorzunehmen. Dazu verwenden wir die Menge $G(I)$, mit der bijektiven Abbildung

$$G : \begin{cases} I & \rightarrow \{0, 1, \dots, N^2 - 1\} \\ (m_1, m_2) & \mapsto m_2 N + m_1 \end{cases}.$$

Wir numerieren also die Pixel des Bildes spaltenweise durch.

Eine Menge $E = \{I_0, \dots, I_r\}, r \in \mathbb{N}$, von Teilmengen $I_\nu \subset I$ ist eine *disjunkte Überdeckung* von I wenn $I_\nu \cap I_\mu = \emptyset$ für $\nu \neq \mu$ gilt sowie $\bigcup_{\nu=0}^r I_\nu = I$. Wir interessieren uns im Folgenden für disjunkte Überdeckungen E der Indexmenge I , die zwei Bedingungen erfüllen:

1. Jede Teilmenge I_ν enthält vier Indizes, d. h. $\#I_\nu = 4$.
2. Jedes Indexelement von I_ν besitzt mindestens einen Nachbarn in I_ν , d. h.

$$\forall (m_1, m_2) \in I_\nu \exists (m'_1, m'_2) \in I_\nu : (m'_1, m'_2) \in N_4(m_1, m_2).$$

Solche Teilmengen I_ν nennen wir *Tetromino*, weil das Überdeckungsproblem der diskreten Indexmenge $I = \{0, 1, \dots, N-1\}^2$ sehr eng mit dem bekannten Überdeckungsproblem des



Abbildung 3.1: Die fünf freien Tetrominoes, auch O-, I-, T-, S-, und L-Tetromino genannt.

Quadrates $[0, N]^2$ mittels Tetrominoes verbunden ist. Diese Tetromino-Überdeckungen werden wir im nächsten Unterkapitel kurz erläutern.

Den vier Elementen eines jeden Tetrominos $I_v = \{(m_1^1, m_2^1), (m_1^2, m_2^2), (m_1^3, m_2^3), (m_1^4, m_2^4)\}$ möchten wir nun eine Ordnungszahl zwischen 0 und 3 zuordnen. Dies geschieht mittels einer Bijektion $L : I_v \rightarrow \{0, 1, 2, 3\}$, die die vier eindimensionalen Werte $G(m_1^1, m_2^1), \dots, G(m_1^4, m_2^4)$ der Größe nach ordnet. Dem kleinsten Index wird die 0 zugeordnet, dem größten die 3.

3.2 Tetromino-Überdeckungen

Polyominoes sind geometrische Formen, die aus zusammengesetzten Einheitsquadraten bestehen, die durch mindestens eine gemeinsame Kante miteinander verbunden sind. Betrachtet man Figuren aus vier Quadraten, so spricht man von Tetrominoes. Die Frage, mit welchen Polyominoes man welche Flächen überdecken kann, beschäftigt die Mathematiker schon seit langer Zeit. Eine grundlegende Arbeit ist das Buch von Golomb [Gol94], vertiefende Studien über Tetrominoes können z. B. in der Dissertation von Korn [Kor04] gefunden werden. Das Überdeckungsproblem einer Rechteckfläche mittels Tetrominoes ist der breiten Öffentlichkeit durch den Computerspiel-Klassiker „Tetris“ bekannt geworden.

Lässt man Rotationen und Spiegelungen außer acht, gibt es fünf verschiedene Figuren, die sogenannten *freien Tetrominoes*, siehe Abbildung 3.1. Manchmal bezeichnet man diese Tetrominoes entsprechend ihrer Ähnlichkeit mit Buchstaben als O-, I-, T-, S-, oder L-Tetromino. Berücksichtigen wir die Isometrien, so erhalten wir 19 verschiedene Tetrominoes (auch *feste Tetrominoes* genannt).

Man macht sich schnell klar, dass ein Quadrat $[0, N]^2$ genau dann disjunkt von Tetrominoes überdeckt werden kann, wenn $N \in \mathbb{N}$ gerade ist. Bereits 1937 zeigte Larsson in [Lar37], dass es 117 Möglichkeiten gibt, ein 4×4 -Quadrat mit jeweils vier Tetrominoes zu überdecken. Wie in Abbildung 3.2 dargestellt, gibt es 22 verschiedene Grundformen, um ein solches Quadrat zu überdecken. Die Überdeckung mit vier O-Tetrominoes (erste Zeile) ist invariant gegenüber den Isometrien. Die vier Lösungen in der zweiten Zeile liefern durch Anwendung der Isometrien jeweils eine weitere Überdeckung, während die sieben Konfigurationen der dritten Zeile jeweils drei weitere Versionen liefern. Die zehn Fundamental-Konfigurationen der vierten Zeile sind so unsymmetrisch, dass jede Spiegelung oder Rotation eine neue Überdeckung ergibt, insgesamt also acht Lösungen pro Grundform. Zusammen existieren damit in der Tat $1 \cdot 1 + 4 \cdot 2 + 7 \cdot 4 + 10 \cdot 8 = 117$ Überdeckungsmöglichkeiten.

Mit größer werdendem N steigt die Anzahl der Überdeckungsmöglichkeiten für das Qua-

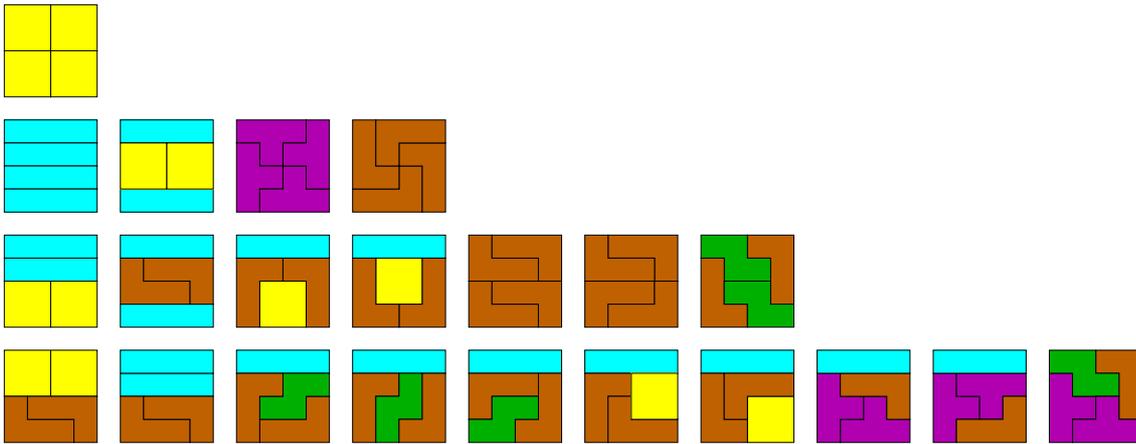


Abbildung 3.2: Die 22 Fundamental-Konfigurationen, die ein 4×4 -Quadrat überdecken. Unter zusätzlicher Berücksichtigung von Rotationen und Spiegelungen erhält man insgesamt $1 \cdot 1 + 4 \cdot 2 + 7 \cdot 4 + 10 \cdot 8 = 117$ Konfigurationen.

drat $[0, N]^2$ rapide. Schon für ein Quadrat der Größe 8×8 ist $117^4 > 10^8$ eine grobe untere Schranke für die Anzahl der möglichen Tetromino-Überdeckungen. Um in der Praxis die Überdeckungsmöglichkeiten noch handhaben zu können, ist es vernünftig, wenn wir uns im Weiteren bei den gesuchten Partitionen der Bildindexmenge auf Zerlegungen in 4×4 -Blöcke beschränken.

3.3 Die Idee der Tetrolets

Die klassische zweidimensionale Haarwavelet-Zerlegung eines Bildes kann als eine spezielle Tetromino-Partition angesehen werden. Bevor wir nun die diskrete Tetrolet-Transformation einführen, schreiben wir die übliche HWT noch einmal in einer Notation auf, die konsistent ist zur dann folgenden Idee der Tetrolets.

In der Haarwavelet-Filterbank werden der Tiefpassanteil und die Hochpassanteile durch gewichtete Mittelwertbildungen von jeweils vier Pixelwerten berechnet, wobei die vier Pixel in einem 2×2 -Bildblock angeordnet sind. Genauer gesagt: Mit $I_{m_1, m_2} = \{(2m_1, 2m_2), (2m_1 + 1, 2m_2), (2m_1, 2m_2 + 1), (2m_1 + 1, 2m_2 + 1)\}$ für $m_1, m_2 = 0, 1, \dots, \frac{N}{2} - 1$, erhalten wir eine dyadische Partition $E = \{I_{0,0}, \dots, I_{\frac{N}{2}-1, \frac{N}{2}-1}\}$ der Indexmenge I . Sei nun L die oben eingeführte bijektive Abbildung, die die vier Pixel in I_{m_1, m_2} in eine eindeutige Reihenfolge bringt, indem sie ihnen eine Zahl aus $\{0, 1, 2, 3\}$ zuordnet.

Dann berechnen wir den Tiefpassanteil des Bildes \mathbf{a} im ersten Level durch

$$\mathbf{a}^1 = (\mathbf{a}^1[m_1, m_2])_{m_1, m_2=0}^{\frac{N}{2}-1} \quad \text{mit} \quad \mathbf{a}^1[m_1, m_2] = \sum_{(m'_1, m'_2) \in I_{m_1, m_2}} \epsilon[0, L(m'_1, m'_2)] a[m'_1, m'_2] \quad (3.1)$$

ebenso wie die drei Hochpassanteile für $l = 1, 2, 3$

$$\mathbf{w}_l^1 = (w_l^1[m_1, m_2])_{m_1, m_2=0}^{\frac{N}{2}-1} \quad \text{mit} \quad w_l^1[m_1, m_2] = \sum_{(m'_1, m'_2) \in I_{m_1, m_2}} \epsilon[l, L(m'_1, m'_2)] a[m'_1, m'_2], \quad (3.2)$$

wobei die Gewichte $\epsilon[l, m]$, $l, m = 0, \dots, 3$, Einträge der Haarwavelet-Transformationsmatrix

$$W := (\epsilon[l, m])_{l, m=0}^3 = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}. \quad (3.3)$$

sind. Das Zerlegen der Indexmenge in dyadische Quadrate I_{m_1, m_2} ist offensichtlich sehr ineffizient, weil die lokalen Strukturen eines Bildes nicht berücksichtigt werden. Unsere Idee ist nun, allgemeinere Partitionen in Abhängigkeit von der lokalen Bildgeometrie zuzulassen. Dazu verwenden wir adaptive Tetromino-Zerlegungen. Wie schon im vorhergehenden Unterkapitel erwähnt, zerlegen wir die Bildindexmenge zunächst in 4×4 -Blöcke. In jedem dieser Blöcke suchen wir dann eine optimale Tetromino-Überdeckung entsprechend der lokalen Bildstruktur.

Aufgrund dieser Überlegungen fügen wir den beiden in Abschnitt 3.1 geforderten Bedingung an die gewünschte Zerlegung E der Indexmenge I eine dritte Bedingung hinzu:

3. Jeder 4×4 Block $Q_{m_1, m_2} := \{4m_1, \dots, 4m_1 + 3\} \times \{4m_2, \dots, 4m_2 + 3\}$, $m_1, m_2 = 0, 1, \dots, \frac{N}{4} - 1$, wird durch genau vier Tetrominoes disjunkt überdeckt.

Wir erkennen jetzt, dass die klassische HWT ein Spezialfall der Tetrolet-Transformation ist, wenn wir die Adaptivität aufgeben und a priori in jedem 4×4 -Block die Überdeckung mittels der vier O-Tetrominoes wählen würden (erste Figur in Abbildung 3.2).

In einer detaillierten Beschreibung des Zerlegungsalgorithmus erklären wir genauer, was eine „optimale“ Tetromino-Überdeckung ist, nämlich, dass die sogenannten Tetrolet-Koeffizienten minimale l^1 -Norm aufweisen. Dafür definieren wir im folgenden Abschnitt Tetrolets als Haarwavelets mit Tetromino-Trägern.

3.4 Eine Orthonormalbasis der Tetrolets

Wir beschreiben die diskreten Basisfunktionen, die dem unten stehenden Algorithmus zugrunde liegen werden. Wir erinnern uns, dass das digitale Bild $\mathbf{a} = (a[m_1, m_2])_{(m_1, m_2) \in I}$ eine Teilmenge des $l_2(\mathbb{Z}^2)$ ist. Für jedes Tetromino I_v aus I definieren wir mithilfe der Haarwavelet-Transformationsmatrix in (3.3) die diskreten Funktionen

$$\phi_{I_v}[m_1, m_2] := \begin{cases} 1/2, & (m_1, m_2) \in I_v, \\ 0, & \text{sonst,} \end{cases}$$

und

$$\psi_{l_v}^l[m_1, m_2] := \begin{cases} \epsilon[l, L(m_1, m_2)], & (m_1, m_2) \in l_v, \\ 0, & \text{sonst.} \end{cases}$$

Wegen der tetrominoförmigen Träger nennen wir $\psi_{l_v}^l$ *Tetrolets*, und ϕ_{l_v} ist die zugehörige Skalierungsfunktion. Als eine unmittelbare Konsequenz aus der Orthonormalität der üblichen zweidimensionalen Haar-Basisfunktionen und der disjunkten Zerlegung des diskreten Raumes mittels der Tetromino-Träger, erhalten wir die folgende wesentliche Tetrolet-Eigenschaft.

Satz 3.1. Für jede zulässige Überdeckung $\{l_0, l_1, l_2, l_3\}$ eines 4×4 -Quadrats $Q \subset \mathbb{Z}^2$ ist das Tetroletsystem

$$\{\phi_{l_v} : v = 0, 1, 2, 3\} \cup \{\psi_{l_v}^l : v = 0, 1, 2, 3; l = 1, 2, 3\}$$

eine ONB des $l^2(Q)$.

Damit können wir jedes Bild $\mathbf{a} = (a[m_1, m_2])_{(m_1, m_2) \in l}$ darstellen mit

$$a[m_1, m_2] = \sum_{v=0}^3 c_v[m_1, m_2] \phi_{l_v}[m_1, m_2] + \sum_{l=1}^3 \sum_{v=0}^3 d_v^l[m_1, m_2] \psi_{l_v}^l[m_1, m_2],$$

wobei $\{l_0, l_1, l_2, l_3\}$ die Tetromino-Überdeckung des 4×4 -Blockes $Q \subset l$ ist, der das Pixel (m_1, m_2) enthält. Die Koeffizienten haben die Gestalt

$$c_v[m_1, m_2] = \langle a[\cdot, \cdot], \phi_{l_v} \rangle = \sum_{(m', n') \in l_v} \frac{1}{2} a[m', n'],$$

$$d_v^l[m_1, m_2] = \langle a[\cdot, \cdot], \psi_{l_v}^l \rangle = \sum_{(m', n') \in l_v} \epsilon[l, L(m', n')] a[m', n'].$$

falls $(m_1, m_2) \in l_v$.

3.5 Detaillierte Beschreibung des Tetrolet-Filterbank-Algorithmus

Der Analyse-Teil des Tetrolet-Filterbank-Algorithmus besitzt die folgende grobe Struktur.

Algorithmus: Adaptive Tetrolet-Zerlegung

Input: Bild $\mathbf{a} = (a[m_1, m_2])_{m_1, m_2=0}^{N-1}$ mit $N = 2^J, J \in \mathbb{N}$.

1. Zerlege das Bild in 4×4 -Blöcke.
2. Suche die „dünnste“ Tetrolet-Darstellung in jedem Bildblock.
3. Ordne die Tief- und Hochpasskoeffizienten eines jeden Blockes wieder in einem 2×2 -Block an.
4. Speichere die Tetrolet-Koeffizienten (Hochpassanteile).
5. Wende Schritte 1 bis 4 iterativ auf das Tiefpassbild an.

Output: Zerlegtes Bild $\tilde{\mathbf{a}}$.

Wir untersuchen nun die einzelnen Schritte des Zerlegungsalgorithmus. Dabei richten wir unser Hauptaugenmerk auf den zweiten Schritt, in dem die Adaptivität ins Spiel kommt.

Das Eingangsbild ist gegeben durch $\mathbf{a}^0 = (a[m_1, m_2])_{m_1, m_2=0}^{N-1}$ mit $N = 2^J, J \in \mathbb{N}$. Wir sind also in der Lage $J - 1$ Level der Tetrolet-Transformation durchzuführen. Im j ten Level, $j = 1, \dots, J - 1$, werden folgende Berechnungen durchgeführt.

1. Zerlege das Tiefpassbild \mathbf{a}^{j-1} in Blöcke Q_{m_1, m_2} der Größe 4×4 , $m_1, m_2 = 0, \dots, \frac{N}{2^{j+1}} - 1$.
2. In jedem Block Q_{m_1, m_2} betrachten wir die 117 zulässigen Tetromino-Überdeckungen $c = 1, \dots, 117$. Für jede Überdeckung c wenden wir die HWT auf die vier adaptiven Tetromino-Teilmengen $I_s^{(c)}, s = 0, 1, 2, 3$, an. Auf diese Weise erhalten wir für jede Überdeckung c vier Tiefpass-Koeffizienten und 12 Tetrolet-Koeffizienten. Wir berechnen dazu in Q_{m_1, m_2} analog zu (3.1) und (3.2) die gewichteten Pixel-Mittelwerte für jede zulässige Konfiguration $c = 1, \dots, 117$,

$$\mathbf{a}^{j,(c)} = (a^{j,(c)}[s])_{s=0}^3 \quad \text{mit} \quad a^{j,(c)}[s] = \sum_{(m_1, m_2) \in I_s^{(c)}} \epsilon[0, L(m_1, m_2)] a^{j-1}[m_1, m_2], \quad (3.4)$$

ebenso wie die drei Hochpassanteile für $l = 1, 2, 3$

$$\mathbf{w}_l^{j,(c)} = (w_l^{j,(c)}[s])_{s=0}^3 \quad \text{mit} \quad w_l^{j,(c)}[s] = \sum_{(m_1, m_2) \in I_s^{(c)}} \epsilon[l, L(m_1, m_2)] a^{j-1}[m_1, m_2]. \quad (3.5)$$

Hierbei kommen die Gewichte/Koeffizienten $\epsilon[l, L(m_1, m_2)]$ wieder aus (3.3), und L ist die bijektive Abbildung, die die vier Indexpaare (m_1, m_2) in $I_s^{(c)}$ mit den Werten 0, 1, 2, und 3 in absteigender Reihenfolge verbindet. Das bedeutet, dass mithilfe der eindimensionalen Indizierung $G(m_1, m_2)$ der kleinste Index mit dem Wert 0 identifiziert wird und der größte mit dem Wert 3.

Jetzt wird die Überdeckung c^* so gewählt, dass die 12 Tetrolet-Koeffizienten die kleinste l^1 -Norm aufweisen

$$c^* = \arg \min_c \sum_{l=1}^3 \|\mathbf{w}_l^{j,(c)}\|_1 = \arg \min_c \sum_{l=1}^3 \sum_{s=0}^3 |w_l^{j,(c)}[s]|. \quad (3.6)$$

Auf diese Weise bekommt man für jeden Bildblock Q_{m_1, m_2} eine optimale Tetroletzerlegung $[\mathbf{a}^{j,(c^*)}, \mathbf{w}_1^{j,(c^*)}, \mathbf{w}_2^{j,(c^*)}, \mathbf{w}_3^{j,(c^*)}]$, die die lokalen Strukturen des Bildblocks berücksichtigt. Die beste Überdeckung c^* ist nämlich eine solche, deren Tetrominoes keine wichtige Bildkante im Bild \mathbf{a}^{j-1} schneiden. Weil die Tetrolet-Koeffizienten so klein wie möglich werden, bekommen wir schließlich eine dünne Bilddarstellung. Für jeden Block Q_{m_1, m_2} müssen wir uns zusätzlich merken, welche Überdeckung c^* gewählt wurde, denn diese Information ist notwendig für eine perfekte Rekonstruktion. Wenn die optimale Überdeckung nicht eindeutig ist, wählen wir das c^* , das bereits in den vorigen Bildblöcken möglichst häufig gewählt wurde. Auf diese Weise reduzieren wir

die Kodierungskosten für die Adaptivitätswerte. Wir schreiben den Wert c^* für jeden der $\frac{N^2}{4^{j+1}}$ Bildblöcke Q_{m_1, m_2} , $m_1, m_2 = 0, \dots, \frac{N}{2^{j+1}} - 1$, in einen Vektor

$$s^j = (s^j[t])_{t=0}^{\frac{N^2}{4^{j+1}}-1} \quad \text{mit} \quad s^j[t] = c^* \text{ im Bildblock } Q_{m_1, m_2} \text{ mit } t = G(m_1, m_2). \quad (3.7)$$

3. Nun ordnen wir die Einträge der 4er-Vektoren $\mathbf{a}^{j, (c^*)}$ und $\mathbf{w}_l^{j, (c^*)}$ jeweils in einer 2×2 -Matrix an, um Tiefpass- und Hochpassbilder zu bekommen. Dazu verwenden wir eine Reshape-Funktion R ,

$$\mathbf{a}_{|Q_{m_1, m_2}}^j = R(\mathbf{a}^{j, (c^*)}) = \begin{pmatrix} \mathbf{a}^{j, (c^*)}[0] & \mathbf{a}^{j, (c^*)}[2] \\ \mathbf{a}^{j, (c^*)}[1] & \mathbf{a}^{j, (c^*)}[3] \end{pmatrix}, \quad (3.8)$$

und analog $\mathbf{w}_{l|Q_{m_1, m_2}}^j = R(\mathbf{w}_l^{j, (c^*)})$ für $l = 1, 2, 3$. Für eine günstige Darstellung des Tiefpassbildes (in Hinblick auf die Tetromino-Überdeckung im nächsten Zerlegungsschritt), ist eine „vernünftige“ Anordnung der Tiefpass-Werte wesentlich. Das bedeutet, dass die Nummerierung der vier Tetrominoes der optimalen Überdeckung c^* in jedem Block Q_{m_1, m_2} sehr wichtig ist. Diese Nummerierung mittels der Werte 0, 1, 2 und 3 muss so geschehen, dass die Geometrie der Überdeckung möglichst gut der Anordnung $\begin{pmatrix} 0 & 2 \\ 1 & 3 \end{pmatrix}$ entspricht, weil die Tiefpass-Koeffizienten gemäß (3.8) in dieser Form angeordnet werden. Wir wählen die Quadrate-Überdeckung als Referenzüberdeckung, weil sie eine optimale Anordnung darstellt. Hier wird nämlich der Koeffizient, der aus den vier Pixeln links oben berechnet wird, in der Tiefpass-Matrix wieder in den Eintrag links oben geschrieben. Der Koeffizient, der aus den vier Pixeln links unten berechnet wird, ist der Matrixeintrag links unten usw., siehe Abbildung 3.3(a). Also nummerieren wir die vier Tetrominoes der gewählten, optimalen Überdeckung c^* , indem wir sie mit dem Quadrate-Fall vergleichen. Unter den 24 Möglichkeiten, die vier Tetrominoes durchnummerieren, wählen wir die Variante, die die höchste Übereinstimmung mit der Quadrate-Partition aufweist. Das Beispiel in Abbildung 3.3 macht das Vorgehen sofort klar: Für den Vergleich der verschiedenen Nummerierungen (und der daraus resultierenden Anordnungen der Koeffizienten) berechnen wir die Abweichung von dem Quadrate-Fall, der in 3.3(a) dargestellt ist. Dazu zählen wir die Anzahl der Pixel in Q_{m_1, m_2} , für die die Nummerierung von derjenigen der Quadrate-Überdeckung abweicht (diese sind in 3.3(b) und (c) mit einem Punkt markiert). Wir wählen dann die Nummerierung mit der minimalen Abweichung. Diese optimale Konfiguration muss nicht eindeutig sein. Die Anordnung in Abbildung 3.3(b) würde zu einem verzerrten Tiefpassbild führen, weil der Eintrag oben rechts, der Koeffizient $\mathbf{a}^{j, (c^*)}[2]$, von vier Pixeln herrührt, die im unteren Bereich des Bildblocks Q_{m_1, m_2} lagen. Abbildung 3.3(c) zeigt eine optimale Anordnung.

4. Nachdem wir in jedem Bildblock Q_{m_1, m_2} für $m_1, m_2 = 0, \dots, \frac{N}{2^{j+1}} - 1$ eine günstige Darstellung gefunden haben, speichern wir (wie üblich) die Matrizen des Tiefpassbildes

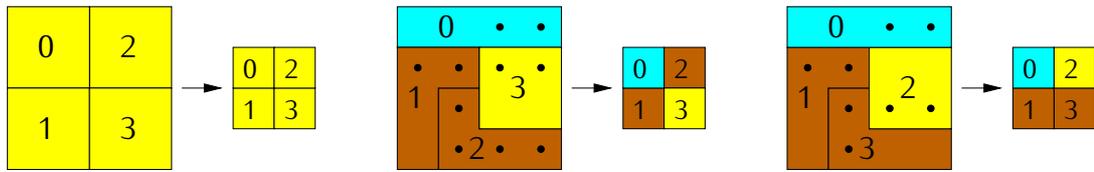


Abbildung 3.3: Beispiel für zwei verschiedene Möglichkeiten der Nummerierung der Tetrominoes, die zu zwei verschiedenen Tiefpassbildern führen. (a) Referenzanordnung, (b) Schlechte Anordnung, (c) Optimale Anordnung.

$\mathbf{a}^j = \left(\mathbf{a}_{Q_{m_1, m_2}}^j \right)_{m_1, m_2=0}^{\frac{N}{2^{j+1}}-1}$ und der Hochpassbilder $\mathbf{w}_l^j = \left(\mathbf{w}_{l|Q_{m_1, m_2}}^j \right)_{m_1, m_2=0}^{\frac{N}{2^{j+1}}-1}$, $l = 1, 2, 3$, ab, indem wir das alte Tiefpassbild \mathbf{a}^{j-1} durch die Matrix

$$\begin{pmatrix} \mathbf{a}^j & \mathbf{w}_2^j \\ \mathbf{w}_1^j & \mathbf{w}_3^j \end{pmatrix}$$

ersetzen.

Nach einer geeigneten Anzahl von Zerlegungsschritten, wenden wir ein *Wavelet-Shrinkage* auf die Tetrolet-Koeffizienten an, um eine dünne Bilddarstellung zu erhalten. Bei den numerischen Experimenten im Abschnitt 3.8, werden wir stets die globale Hard-Thresholding-Funktion

$$S_\lambda(x) = \begin{cases} x, & |x| \geq \lambda, \\ 0, & |x| < \lambda, \end{cases}$$

benutzen.

Zur Rekonstruktion des Bildes wird der Algorithmus invers ausgeführt. Man benötigt – wie gewöhnlich – die Tiefpass-Koeffizienten aus dem größten Level $J - 1$ und alle Tetrolet-Koeffizienten aus den Leveln $J - 1, \dots, 1$. Zusätzlich benötigt man den Vektor $\mathbf{s} := (\mathbf{s}^{J-1}, \dots, \mathbf{s}^1)$ der Adaptivitätswerte, der uns die notwendige Information liefert, welche der zulässigen Tetromino-Konfigurationen im jeweiligen Bildblock gewählt wurde. Für $j = J - 1, \dots, 1$ hat der Teilvektor \mathbf{s}^j gemäß (3.7) die Länge $\frac{N^2}{4^{j+1}}$, weil es ebenso viele Bildblöcke im j -ten Level gibt. Nach $J - 1$ Zerlegungsebenen haben wir somit in \mathbf{s}

$$\sum_{j=1}^{J-1} \frac{N^2}{4^{j+1}} = \frac{N^2}{4} \sum_{j=1}^{J-1} \frac{1}{4^j} = \frac{N^2}{4} \left(\frac{1 - \frac{1}{4^J}}{1 - \frac{1}{4}} - 1 \right) = \frac{N^2}{12} \left(1 - \frac{1}{4^{J-1}} \right) \quad (3.9)$$

Werte zusätzlich zu speichern. Bei vollständiger Zerlegung des Bildes, d. h. für $J = \log_2(N)$, enthält der Vektor \mathbf{s} also $(N^2 - 4)/12$ Adaptivitätswerte.

Bemerkung 3.2. Man kann die adaptive Tetrolet-Filterbank als eine Haarwavelet-Filterbank mit lokal permutierten Pixelwerten interpretieren. Diese Auffassung führt zu einer effizienten Implementierung der Tetrolet-Zerlegung. Und zwar schreiben wir die 16 Pixelwerte eines jeden 4×4 -Bildblocks Q_{m_1, m_2} in einen Vektor $\mathbf{q} = (q_1, \dots, q_{16})^T$ indem wir die Spalten der

Bildmatrix aneinander hängen. Nach der Anwendung einer Permutationsmatrix $P_c \in \mathbb{R}^{16 \times 16}$, die die Bildwerte entsprechend der optimalen Tetromino-Überdeckung $c \in \{1, \dots, 117\}$ umordnet, berechnen wir die Tiefpass- und Hochpass-Koeffizienten gemäß (3.4) und (3.5) durch

$$\tilde{\mathbf{q}} = (I_4 \otimes W) P_c \mathbf{q}. \quad (3.10)$$

Hier ist W wieder die HWT-Matrix aus (3.3), I_4 ist die Einheitsmatrix der Größe 4×4 , und \otimes bezeichnet das Kronecker-Produkt, d. h. $(I_4 \otimes W) = \text{blockdiag}(W, W, W, W)$. Der Ergebnisvektor $\tilde{\mathbf{q}} \in \mathbb{R}^{16}$ hat die Gestalt

$$\tilde{\mathbf{q}} = (a^{j,(c)}[0], w_1^{j,(c)}[0], w_2^{j,(c)}[0], w_3^{j,(c)}[0], \dots, a^{j,(c)}[3], w_1^{j,(c)}[3], w_2^{j,(c)}[3], w_3^{j,(c)}[3])^T.$$

Beispiel 3.3. Betrachte den 4×4 -Block eines synthetischen Bildes

$$\begin{bmatrix} 20 & 20 & 20 & 20 \\ 20 & 160 & 160 & 20 \\ 20 & 160 & 160 & 20 \\ 20 & 20 & 20 & 20 \end{bmatrix}, \quad (3.11)$$

siehe Abbildung 3.4(a). Die konventionelle HWT der Bildfunktion hat die maximale Anzahl von nicht-verschwindenden Waveletkoeffizienten. Nach (3.1) und (3.2) erhält man nämlich nach einem Zerlegungsschritt

$$\begin{bmatrix} \mathbf{a}^1 & \mathbf{w}_2^1 \\ \mathbf{w}_1^1 & \mathbf{w}_3^1 \end{bmatrix} = \begin{bmatrix} 110 & 110 & -70 & -70 \\ 110 & 110 & 70 & 70 \\ -70 & 70 & 70 & -70 \\ -70 & 70 & -70 & 70 \end{bmatrix}.$$

Alle 12 Waveletkoeffizienten sind von Null verschieden! Im Gegensatz dazu verschwinden bei der Tetrolet-Transformation alle Pixeldifferenzen, weil die Tetromino-Überdeckung entsprechend der Bildgeometrie adaptiv gewählt wird, siehe Abbildung 3.4(c). Beachte, dass es wegen der Rotationsinvarianz vier optimale Tetromino-Überdeckungen gibt.

Wir schreiben die Spalten von (3.11) in einen Vektor

$$\mathbf{q} = (20, 20, 20, 20, 20, 160, 160, 20, 20, 160, 160, 20, 20, 20, 20, 20)^T$$

und wenden eine Permutationsmatrix P_c an, die der entsprechenden Tetromino-Partition in Abbildung 3.4(c) entspricht. In diesem Fall bedeutet das, dass P_c bei eindimensionaler Indizierung der Pixel durch

$$P_c (1, \dots, 16)^T = (1, 5, 9, 13, 2, 3, 4, 8, 6, 7, 10, 11, 12, 14, 15, 16)^T$$

gegeben ist. Wir erhalten also mittels (3.10) in der Tat eine dünne Darstellung

$$\tilde{\mathbf{q}} = (40, 0, 0, 0, 40, 0, 0, 0, 320, 0, 0, 0, 40, 0, 0, 0)^T,$$

weil alle Tetrolet-Koeffizienten verschwinden.

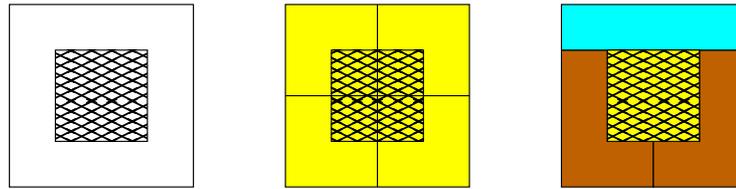


Abbildung 3.4: Beispiel einer adaptiven Tetromino-Überdeckung eines Bildblocks. (a) Bildfunktion, (b) Quadratische Träger der klassischen Haarwavelets, (c) Träger der adaptiven Tetrolets.

3.6 Adaptivitätskosten: Modifikationen der Tetrolet-Transformation

Die im Vorhergehenden beschriebene Tetrolet-Transformation reduziert die Anzahl der Waveletkoeffizienten im Vergleich zur klassischen Tensorprodukt-Wavelettransformation. Diese Verbesserung muss mit zusätzlichem Speicheraufwand bezahlt werden. Denn neben den Waveletkoeffizienten müssen wir uns noch die zusätzliche Information merken, welche Überdeckung in welchem Bildblock verwendet wurde. Der Aufwand zum Abspeichern dieser für die Rekonstruktion notwendigen Information kann nicht vernachlässigt werden. In diesem Abschnitt untersuchen wir die Adaptivitätskosten genauer und schlagen gewisse Modifikationen der Tetrolet-Transformation vor, um die Kosten zu reduzieren.

Wir hatten bereits in (3.9) gesehen, dass bei vollständiger Zerlegung eines $N \times N$ -Bildes zusätzlich $(N^2 - 4)/12$ Adaptivitätswerte im Vektor s zu speichern sind.

Es ist bekannt, dass ein Vektor der Länge N , der die Entropie E besitzt, mit $N \cdot E$ Bits abgespeichert werden kann. Das heißt, die Entropie

$$E = - \sum_{i=1}^n p(x_i) \log_2(p(x_i)) \quad (3.12)$$

beschreibt die benötigten Bits pro Pixel (bpp) und ist somit ein geeignetes Maß für die Qualität der Kompression. Die Entropie (3.12) kann als die erwartete Länge eines binären Codes über alle Symbole aus dem gegebenen Alphabet $A = \{x_1, \dots, x_n\}$ aufgefasst werden. Dabei beschreibt $p(x_i)$ die relative Häufigkeit, mit der das Symbol x_i in der Datenmenge auftritt.

Wir schlagen im Folgenden drei Methoden der Entropiereduktion vor, um die Kodierungskosten von s zu reduzieren. Eine Anwendung der modifizierten Versionen der Tetrolet-Transformation sowie eine Kombination von ihnen werden wir im nächsten Unterkapitel vornehmen.

- (a) **TT16.** Der einfachste Ansatz zur Entropiereduktion ist die Reduzierung des Alphabets A . Die ursprüngliche Tetrolet-Transformation wählt für jeden Bildblock die optimale Überdeckung aus dem Alphabet $\{1, \dots, 117\}$. Die starke Ähnlichkeit mancher

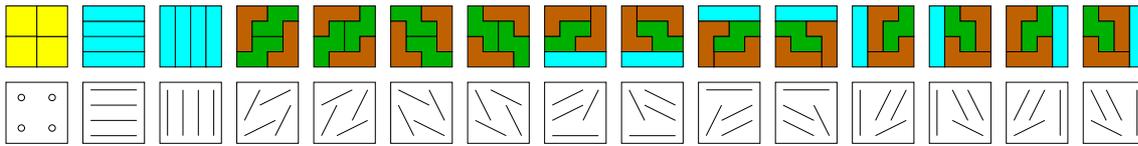


Abbildung 3.5: Die a priori ausgewählten 16 Konfigurationen mit verschiedenen Richtungen.

Tetromino-Konfigurationen (siehe Abbildung 3.2) legt es nahe, einige Überdeckungen außer acht zu lassen. Wir beschränken uns auf 16 geeignete Konfigurationen. Natürlich hängt die optimale Wahl dieser 16 Konfigurationen vom vorliegenden Bild ab, aber es ist möglich, auch a priori 16 wichtige Überdeckungsvarianten auszuwählen. Und zwar entscheiden wir uns für solche Konfigurationen, die verschiedene Richtungen aufweisen, siehe Abbildung 3.5. Auf diese Weise sind wir in der Lage, gerichtete Kanten im Bild gut zu adaptieren. Außerdem führt die Beschränkung auf nur 16 Überdeckungen zu einer erheblichen Beschleunigung der Berechnungszeit der Tetrolet-Zerlegung. Diese Variante der Tetrolet-Transformation nennen wir *TT16*.

- (b) **Relaxierte TT.** Ein zweiter Ansatz der Entropiereduktion ist die Manipulation der Häufigkeiten $p(x_i)$ in (3.12). Wir verändern die Verteilung der Adaptivitätswerte dahingehend, dass zwar alle 117 Überdeckungen zugelassen werden, aber nur sehr wenige Konfigurationen bevorzugt werden. Dies kann durch eine Relaxation der Tetrolet-Transformation erreicht werden, indem wir uns mit einer *fast* optimalen Überdeckung c^* in (3.6) begnügen, um dafür aber eine Überdeckung wählen zu können, die bereits möglichst häufig verwendet wurde. Wir ersetzen also (3.6) durch die zwei Schritte:

1. Suche die Menge $A' \subset A$ von fast optimalen Konfigurationen c , die die Bedingung

$$\sum_{l=1}^3 \|w_l^{j,(c)}\|_1 \leq \min_{c \in A} \sum_{l=1}^3 \|w_l^{j,(c)}\|_1 + \theta$$

mit einem vorgegebenem Toleranzparameter θ erfüllen.

2. Unter diesen Überdeckungen wähle schließlich die Konfiguration $c^* \in A'$, die in den vorherigen Bildblöcken und Leveln bereits möglichst häufig verwendet wurde.

Mithilfe eines geeigneten Parameters θ erreicht man eine zufriedenstellende Balance zwischen einer niedrigen Entropie (d.h. niedrige Adaptivitätskosten) und minimalen Tetrolet-Koeffizienten. Diese Modifikation bezeichnen wir als *Relaxierte TT*.

- (c) **Haar-TT.** Auch die dritte Methode, die wir *Haar-TT* nennen, reduziert die Entropie durch eine Optimierung der Verteilung der Adaptivitätswerte. Mithilfe eines Kantendetektors können wir die verschiedenen Bildblöcke in zwei Klassen einteilen. In Bildblöcken, die Kanten enthalten, wenden wir die Standard-Tetrolet-Transformation mit

ihrer vollen Adaptivität an. In den übrigen Bildblöcken, die innerhalb flacher Bildgebiete liegen, verwenden wir die klassische HWT. Auf diese Weise erreichen wir eine große Anzahl von Überdeckungen mit vier O-Tetrominoes, die der HWT entsprechen.

3.7 Arithmetische Komplexität

Eine zweidimensionale HWT aus vier Pixeln kann mit 8 Additionen durchgeführt werden (bei geeigneter Faktorisierung von W) und mit 4-facher Multiplikation mit dem Faktor $1/2$.

Für ein quadratisches $N \times N$ -Bild gilt: Im j -ten Level wird das Tiefpassbild \mathbf{a}^{j-1} in $\frac{N^2}{4^{j+1}}$ Blöcke der Größe 4×4 zerlegt. Wenn wir in jedem dieser Blöcke p Tetromino-Überdeckungen zulassen (d. h. für die klassische Tetrolet-Transformation gilt $p = 117$, für TT16 gilt $p = 16$) sind folgende Operationen durchzuführen:

1. Berechne für $c = 1, \dots, p$ die Koeffizienten $\mathbf{a}^{j,(c)}$ und $\mathbf{w}_l^{j,(c)}$ aus (3.4) bzw. (3.5)
 $\rightarrow p \cdot 4 \cdot 8$ Additionen.
2. Berechne für $c = 1, \dots, p$ den Ausdruck $\sum_{l=1}^3 \sum_{s=0}^3 |w_l^{j,(c)}[s]|$ aus (3.6)
 $\rightarrow p \cdot 11$ Additionen.
3. Suche die günstigste Überdeckung c^* über alle $c = 1, \dots, p$ gemäß (3.6)
 $\rightarrow p - 1$ Vergleiche.
4. Multipliziere die 16 optimalen Koeffizienten mit dem Faktor $1/2$
 $\rightarrow 16$ Multiplikationen.

Also müssen im j -ten Level insgesamt

$$\frac{N^2}{4^{j+1}}(43p \text{ Additionen} + p - 1 \text{ Vergleiche} + 16 \text{ Multiplikationen})$$

durchgeführt werden. Für die klassische HWT hat man dagegen nur

$$\frac{N^2}{4^{j+1}}(32 \text{ Additionen} + 16 \text{ Multiplikationen}).$$

Die ersten beiden Schritte bewirken die recht hohe Anzahl an arithmetischen Additionen, weil die permutierte HWT für jede der p Tetromino-Überdeckungen durchgeführt werden muss. Im Vergleich zur klassischen HWT besitzen die Kosten einen Faktor der Größenordnung $43p/32$. Allerdings sind diese Berechnungen *unabhängig* voneinander durchführbar, so dass auf einem Parallelrechner in der Praxis so gut wie überhaupt keine verlängerte Rechenzeit bemerkbar ist, wenn nur entsprechend viele Multi-Kerne vorhanden sind.

Wir haben für ein konkretes Beispiel die Berechnungszeit der Tetrolet-Transformation mit anderen Methoden verglichen. Dabei haben wir unsere MATLAB-Codes (ohne C-Routinen)

Bildgröße	Zerlegung		Rekonstruktion	
	256 × 256	512 × 512	256 × 256	512 × 512
Tetrolet	1.0212	4.0259	0.4451	2.0324
TT16	0.6592	2.6735	0.4125	1.7821
Relaxierte TT	1.1419	4.5773	0.4302	1.8668
Haar-TT	1.8096	5.5622	0.4150	1.8373
Relaxierte Haar-TT16	0.6548	2.8035	0.4188	1.8457
Contourlets	0.1536	0.4689	0.1539	0.4782
Curvelets	0.3490	1.9876	0.3741	1.5079

Tabelle 3.1: Vergleich der modifizierten Tetrolet-Transformationen mit gerichteten Waveletmethoden bezüglich der Berechnungszeit (in sec).

auf einem MacBook getestet, das mit einem 2 GHz Intel Core 2 Duo Prozessor und mit 4 GB RAM ausgestattet ist. Als Referenzbild diente das „pepper“-Bild in den Größen 256 × 256 und 512 × 512. Die gemessenen CPU-Zeiten sind in Tabelle 3.1 aufgelistet.

Natürlich besitzen die Contourlet- und Curvelettransformationen eine schnellere Zerlegung als die Tetrolet-Transformation. Das liegt einerseits an der Verwendung von C-Routinen in den beiden MATLAB-Toolboxen, andererseits aber auch an der Tatsache, dass beide Methoden nicht-adaptiv sind. Für eine adaptive Methode erreicht der Tetrolet-Filterbank-Algorithmus eine hervorragende Zerlegungszeit, für ein Bild der Größe 256 × 256 benötigt er nur eine einzige Sekunde. Andere adaptive Waveletsysteme wie die Wedgelets oder Bandelets sind in der Praxis wegen ihrer langsamen Zerlegungen kaum anwendbar, siehe [FDFW07].

Beachte auch folgende Punkte in Tabelle 3.1: Der Zerlegungsschritt bei der Tetrolet-Transformation benötigt länger als der entsprechende Rekonstruktionschritt, weil hier die Adaptivität berücksichtigt werden muss. Die modifizierten Versionen der Tetrolet-Transformation unterscheiden sich nur in der Zerlegung, die inverse Transformation ist dieselbe. Deshalb sind die Zeiten für die Rekonstruktion auch nahezu gleich. Wie bereits weiter oben erwähnt, bringt die Reduzierung der zulässigen Tetromino-Konfigurationen auf nur 16 Überdeckungen ('TT16') einen deutlichen Gewinn in der Berechnungszeit.

3.8 Numerische Resultate

Auch wenn die Tetrolet-Transformation ein effizientes Werkzeug zur Kompression von reellwertigen Datenmenge ist, beschränken wir uns im Folgenden auf digitale Bilder.

Während redundante Waveletframes sehr nützlich sind für die Entstörung von Bildern (weil redundante Information bessere Rekonstruktion der ursprünglichen Daten gewährleistet), ist für die Bildkompression eine Waveletbasis zu bevorzugen. Die Tetrolets



Abbildung 3.6: Testbilder.

bilden eine Basis eines Unterraumes von $l^2(\mathbb{Z}^2)$ und liefern deshalb eine dünne Bilddarstellung nach Anwendung des oben beschriebenen Filterbank-Algorithmus. Die in diesem Kapitel benutzten MATLAB-Programme sind auf unserer Internetseite www.uni-due.de/mathematik/krommweh/ bereitgestellt.

Standard-Tetrolet-Transformation Für unsere umfangreichen numerischen Tests benutzen wir verschiedene Arten von Testbildern, von stückweise konstanten Bildern bis hin zu Bildern mit feinen Texturen, siehe Abbildung 3.6. Wir zerlegen die Bilder vollständig in ihre Tetrolet-Koeffizienten und unterwerfen diese einem globalen Hard-Shrinkage, wobei der Schwellwert λ so gewählt wird, dass eine vorgegebene Anzahl an Koeffizienten erhalten bleibt.

Die Ergebnisse der verschiedenen Testbilder in den Abbildungen 3.7 bis 3.12 sind in Tabelle 3.2 zusammengefasst.

Das synthetische 256×256 -Bild in Abbildung 3.7 zeigt, dass die Tetrolet-Transformation exzellente Ergebnisse für stückweise konstante Bilder liefert. Mit nur 512 Koeffizienten nach dem Shrinkage weist das rekonstruierte Bild einen beachtlichen PSNR-Wert von 38.47 dB auf, weil die gerichteten Kanten gut adaptiert werden.

Obwohl die Haar-ähnlichen Tetrolets nicht stetig sind, illustrieren die weiteren Abbildungen 3.8 bis 3.12, dass die Tetrolet-Transformation sogar für natürliche Bilder die Tensorprodukt-Wavelets in der biorthogonalen 9-7 Filterbank aussticht. Das bestätigt die Feststellung, die im Hinblick auf Wedgelets [Don99] und Grouplets [Mal09] getroffen wur-

	synthetic (Abb. 3.7)	cameraman (Abb. 3.8)	pepper (Abb. 3.9)	Lena (Abb. 3.10)	monarch (Abb. 3.11)	Barbara (Abb. 3.12)
Koeffizienten	512	2048	2048	2048	256	512
Tensor Haar	28.13	25.47	26.11	25.58	18.98	19.69
Tensor 9-7	30.23	27.26	28.96	28.26	21.78	20.49
Tetrolets	37.70	29.29	29.32	28.66	24.19	21.09
Tetrolets BF	38.47	29.17	30.00	28.98	24.43	21.05
Contourlets	30.21	26.07	27.70	27.30	21.00	23.16
Curvelets	26.02	21.91	22.77	22.77	12.85	18.37

Tabelle 3.2: Vergleich der PSNR-Werte bei Approximation.

de, dass auch Haarwavelets in Verbindung mit gut konstruierten adaptiven Methoden sehr gute Resultate liefern können. Das heißt, dass adaptive Methoden im Gegensatz zu nicht-adaptiven Verfahren nicht notwendigerweise glatte Wavelets benötigen. Besonders auffallend ist die Tatsache, dass nahezu keine Gibbs-Artefakte auftreten. Dieser Effekt beruht auf den kleinen Trägern der Tetrolets. Für visuelle Zwecke kann man die Bilder in einem anschließenden Postprocessing-Schritt leicht mit einem bilateralen Filter glätten ('Tetrolets BF'), was bei fast allen Testbildern auch eine leichte Verbesserung des PSNR-Wertes bewirkt. Als Parameter für die Gewichte des bilateralen Filters [TM98] haben wir $\sigma_s = 2$ und $\sigma_t = 35$ mit einer 8er-Nachbarschaft gewählt (siehe die vertiefende Analyse der Filtergewichte in (4.13) im nächsten Kapitel).

Insbesondere anhand des „monarch“-Bildes in Abbildung 3.11 kann man gut beobachten, wie aufgrund der hohen Adaptivität die verschiedenen, gerichteten Kanten vorzüglich approximiert werden. Der Ausschnitt des „Barbara“-Bildes in Abbildung 3.12 zeigt, dass die Tetrolet-Approximation auch für feine Texturen akzeptabel ist, aber deutlich von der Contourlettransformation übertroffen wird. Das liegt an dem Umstand, dass diese Art von gerichteten Streifen-Texturen ideal für Contourlets ist.

Modifizierte Tetrolet-Transformation Wir vergleichen die ursprüngliche Tetrolet-Transformation mit ihren kostenminimierenden Varianten aus dem vorigen Kapitel.

In Abbildung 3.13 sehen wir am Beispiel des „pepper“-Bildes, dass wir mithilfe der TT16 die Entropie fast halbieren können, wenn wir kleine Qualitätseinbußen in der Größenordnung von ungefähr 0.7 dB beim PSNR-Wert in Kauf nehmen. Die Abbildungen 3.14 und 3.15 erläutern die Funktionsweise der Relaxierten TT. Eine vollständige Tetrolet-Zerlegung des 256×256 -„cameraman“-Bildes ergibt 5461 Adaptivitätswerte $c \in \{1, \dots, 117\}$. Die Verteilung dieser Werte im Adaptivitätsvektor s ist in Abbildung 3.14(a) zu sehen, die Entropie beträgt 0.56 bpp. Die relaxierte Version mit dem Toleranzparameter $\theta = 25$ bewirkt eine verbesserte Häufigkeitsverteilung, wie das Histogramm in Abbildung 3.14(b) verdeutlicht.

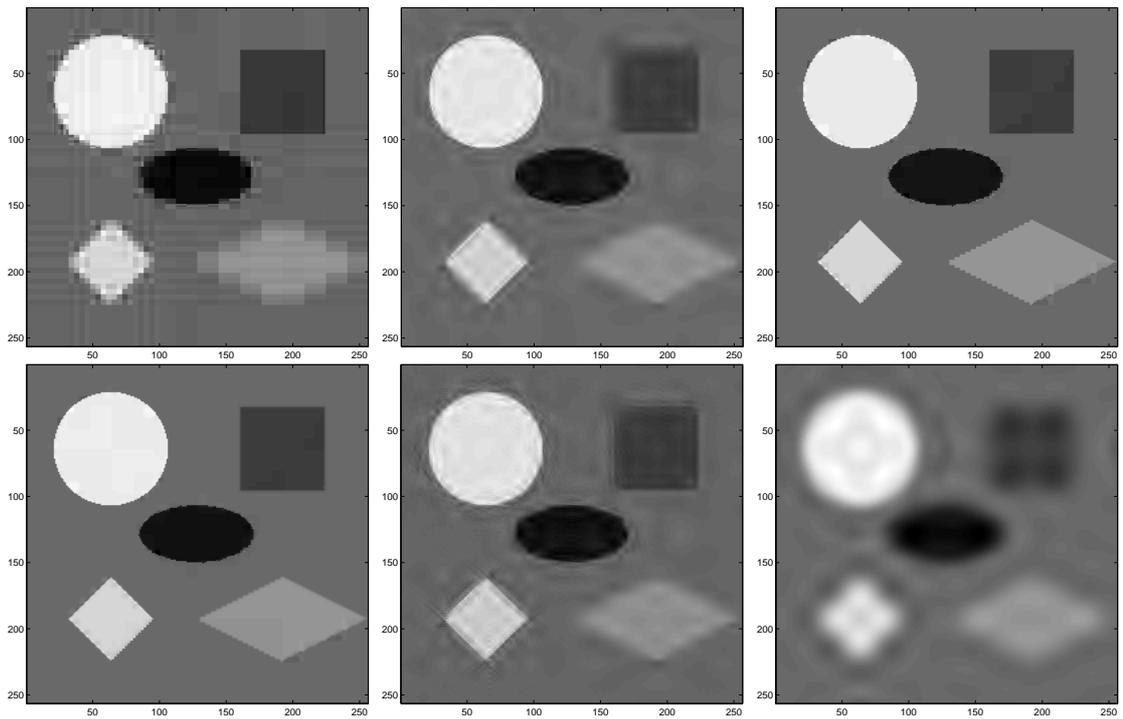


Abbildung 3.7: Approximation des synthetischen Bildes mit 512 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.



Abbildung 3.8: Approximation des „cameraman“-Bildes mit 2048 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.

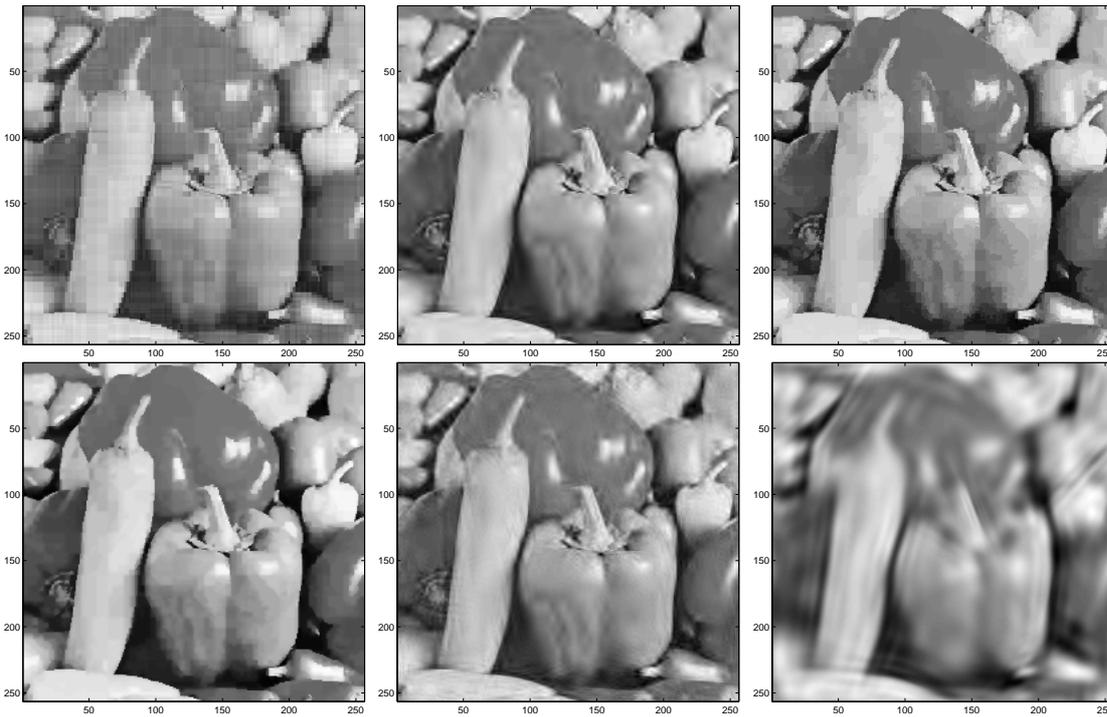


Abbildung 3.9: Approximation des „pepper“-Bildes mit 2048 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.



Abbildung 3.10: Approximation des „Lena“-Bildes mit 2048 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.

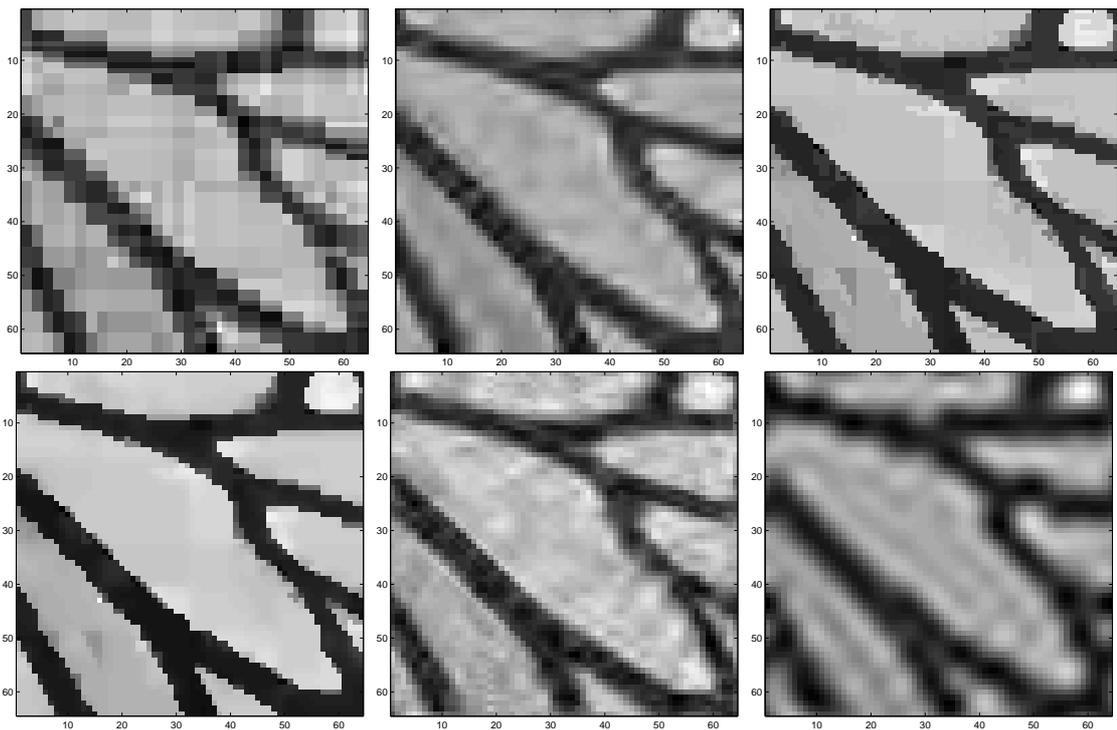


Abbildung 3.11: Approximation des 64×64 -Detailbildes von „monarch“ mit 256 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.

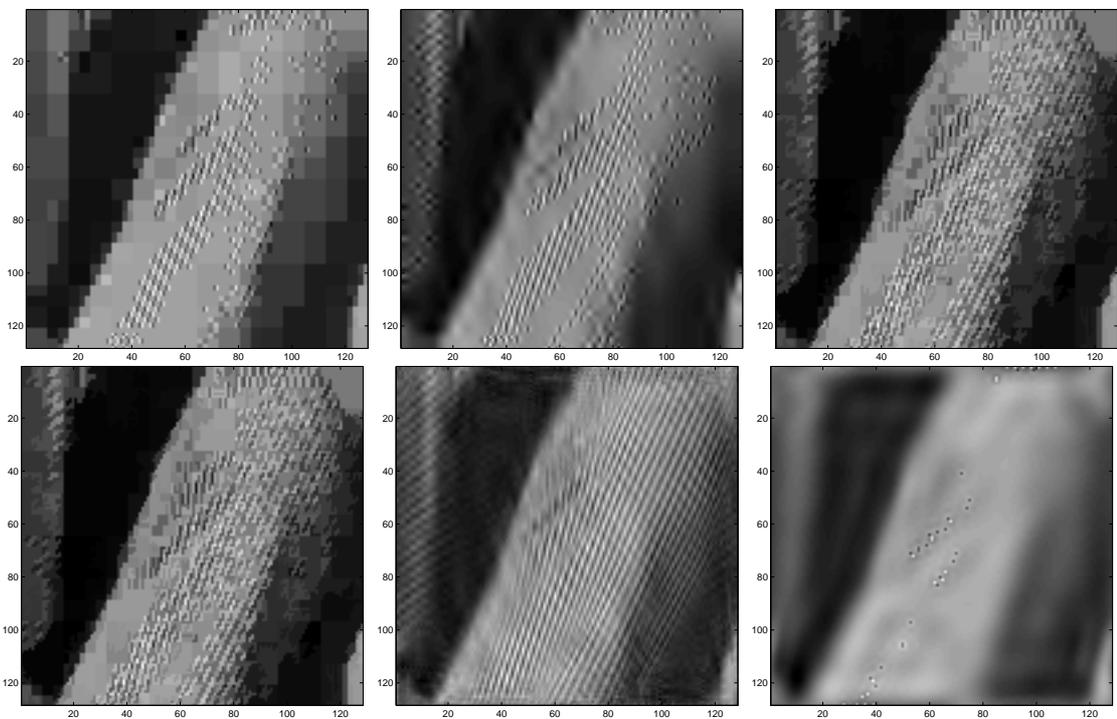


Abbildung 3.12: Approximation des 128×128 -Detailbildes von „Barbara“ mit 512 Koeffizienten. (a) Klassische HWT, (b) Biorthogonal 9-7, (c) Tetrolets, (d) Tetrolets BF, (e) Contourlets, (f) Curvelets.

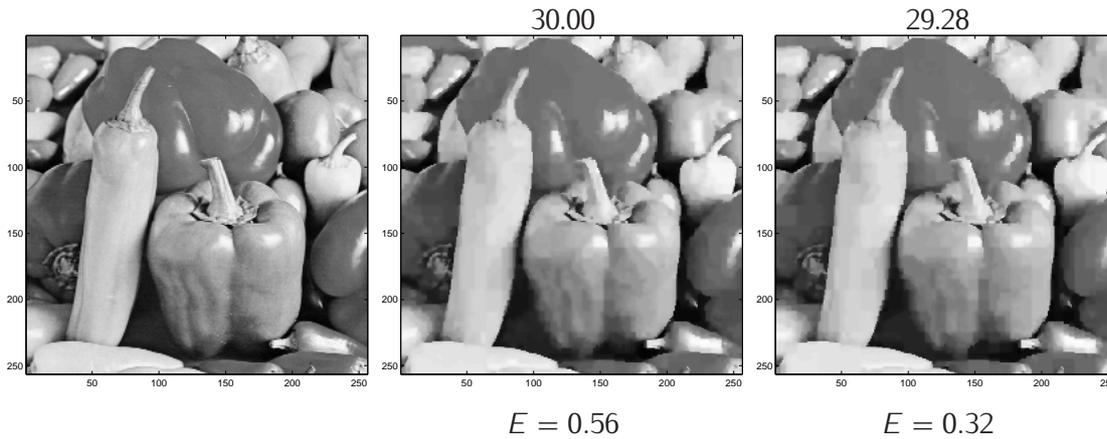


Abbildung 3.13: Approximation des „pepper“-Bildes mit 2048 Koeffizienten. (a) Original, (b) Standard-Tetrolet-Transformation, (c) TT16.

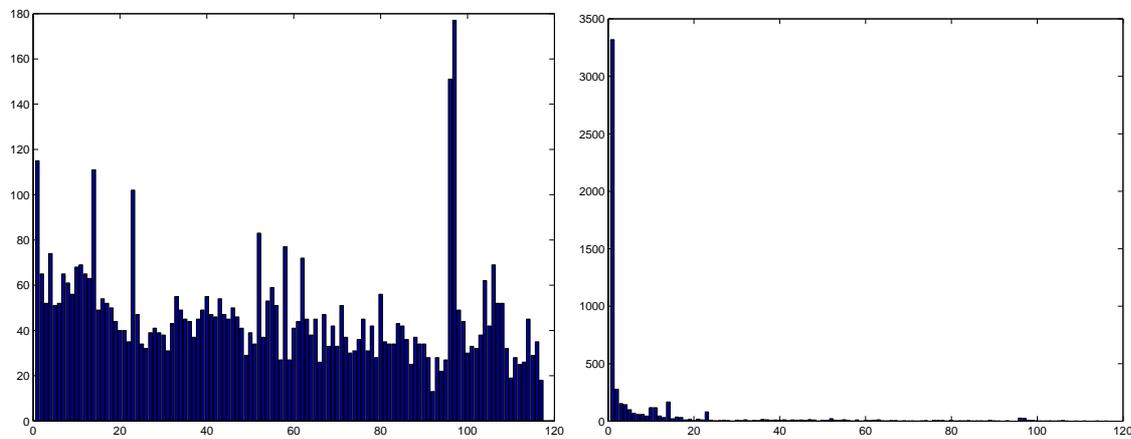


Abbildung 3.14: Verteilung der 5461 Adaptivitätswerte beim „cameraman“-Bild. (a) Standard TT, $E = 0.56$ bpp, (b) Relaxierte TT mit $\theta = 25$, $E = 0.25$ bpp.

Die dazugehörige Entropie wird auf 0.25 bpp reduziert – eine beträchtliche Verbesserung angesichts eines kleinen Verlustes der Qualität (statt 29.17 dB haben wir lediglich einen PSNR-Wert von 28.91 dB).

Abbildung 3.15 illustriert anhand des „cameraman“-Bildes, dass wir durch die Wahl des Kontrollparameters θ die Balance zwischen geringer Entropie und hohem PSNR-Wert steuern können. Für den sehr hohen Wert $\theta = 100$ reduziert sich die Entropie um ca. 80 %, während die Qualität um ca. 1.00 dB gegenüber der Standard-Tetrolet-Transformation ($\theta = 0$) sinkt.

Die dritte Variante zur Minimierung der Adaptivitätskosten, die Haar-TT, wird in Abbildung 3.16 dargestellt, die Änderungsrate zwischen Entropie und PSNR im Vergleich zur Standard-Tetrolet-Transformation ist in der gleichen Größenordnung wie bei der ersten Modifikation (vgl. Abbildung 3.13).

Zusammenfassend präsentieren wir in der Übersicht von Tabelle 3.3 einige numeri-

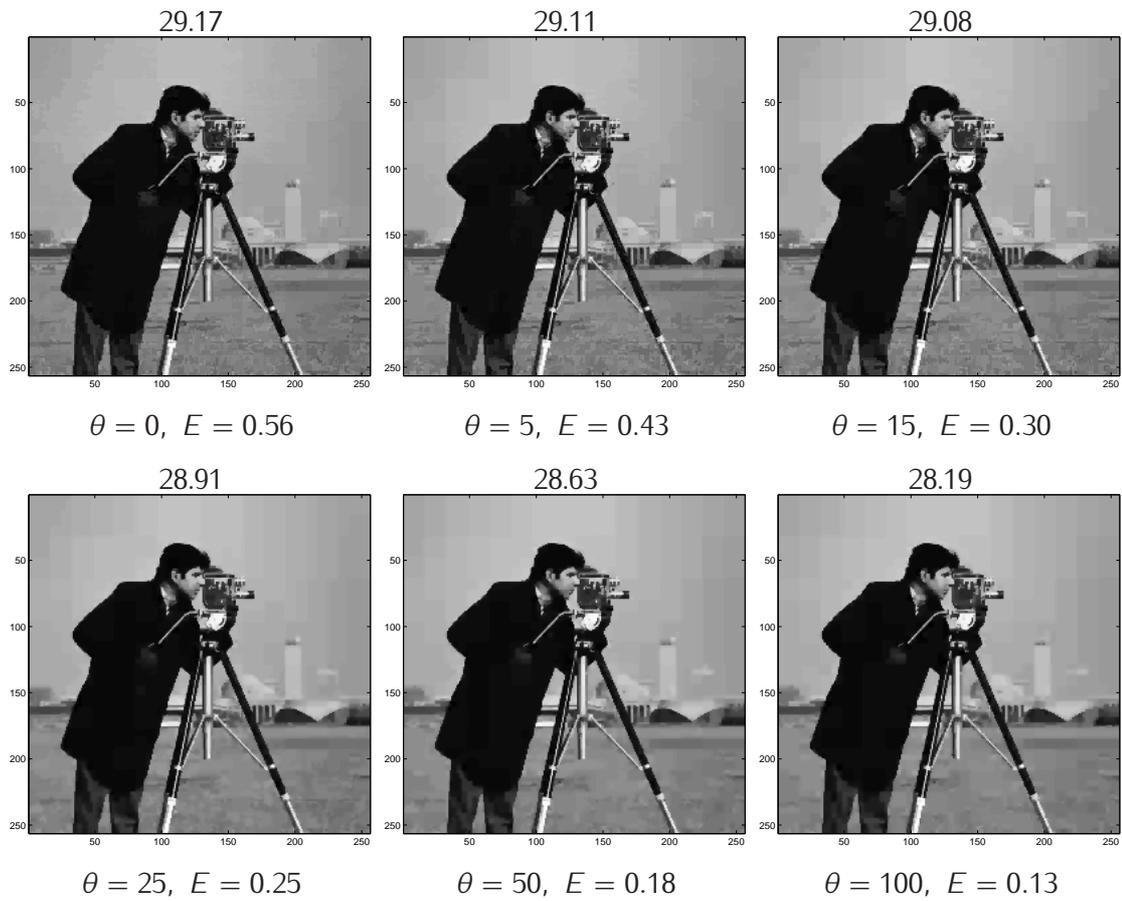


Abbildung 3.15: Approximation des „cameraman“-Bildes mit 2048 Koeffizienten für die Relativierte TT mit verschiedenen Toleranzparametern θ .

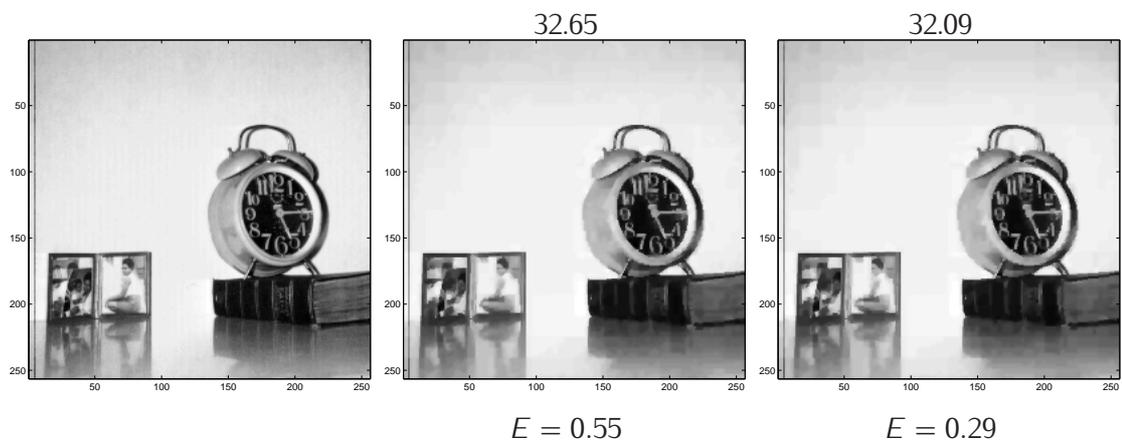


Abbildung 3.16: Approximation des „clock“-Bildes mit 2048 Koeffizienten für die Haar-TT. (a) Original, (b) Standard-Tetrolet-Transformation, (c) Haar-TT.

sche Resultate für das „cameraman“-Bild und das „monarch“-Detailbild. Wir vergleichen die Standard-Tetrolet-Transformation mit den klassischen Tensormethoden (Tensor Haar, Tensor 9-7) sowie mit den modifizierten Versionen (TT16, Relaxierte TT, Haar-TT) und schließlich auch mit einer Kombination der drei Modifikationen (Relaxierte Haar-TT16). Es wird ein Vergleich bzgl. Approximationsqualität und Speicherkosten angestellt. Die Speicherkosten beinhalten die Kodierung der Waveletkoeffizienten und der Adaptivitätswerte (wenn vorhanden). Eine grobe Schätzung für die vollständigen Speicherkosten eines komprimierten Bildes mit N^2 Pixeln kann man mit einem vereinfachten Schema

$$cost = cost_W + cost_P + cost_A,$$

erhalten. Hierbei sind $cost_W = 16 \cdot M/N^2$ die Kosten in bpp, die zum Abspeichern von M Nichtnull-Waveletkoeffizienten mit 16 Bits benötigt werden. Der Ausdruck $cost_P$ gibt die Kodierungskosten für die Position dieser M Koeffizienten an, die vereinfacht mittels $-\frac{M}{N^2} \log_2(\frac{M}{N^2}) - \frac{N^2-M}{N^2} \log_2(\frac{N^2-M}{N^2})$ berechnet werden können. Dies ist nämlich die Entropie der sogenannten *Signifikanz-Karte* [Mal99, S.551]. Die Signifikanz-Karte ist nichts anderes als ein binäres Signal der Länge N^2 , in dem Einsen an den Stellen stehen, wo die M Nichtnull-Koeffizienten sind. Die dritte Kostenkomponente, die Adaptivitätskosten $cost_A = E \cdot R/N^2$, treten nur bei der Tetrolet-Transformation auf. Dabei steht R für die Anzahl der Adaptivitätswerte, und E ist wie bisher die Entropie.

Es ist offensichtlich, dass dieses Schema nur eine sehr grobe obere Schranke für die Speicherkosten liefert. In der Praxis kann die Kodierung der Tetrolet-Koeffizienten wesentlich günstiger durchgeführt werden, indem man z. B. die Korrelation benachbarter Bildblöcke berücksichtigt oder die Korrelation zwischen den Tetrolet-Koeffizienten in aufeinander folgenden Zerlegungsleveln.

Wir haben in Tabelle 3.3 versucht, die Kombination der modifizierten Tetrolet-Transformation so auszubalancieren, dass die vollen Speicherkosten $cost$ in der gleichen Größenordnung sind wie bei dem 9-7 Filter. Jetzt kann man in diesem nunmehr fairen Vergleich feststellen, dass die Tetrolets einen leicht besseren PSNR-Wert aufweisen als der 9-7 Filter. Für die relaxierte Version haben wir den globalen Toleranzparameter $\theta = 25$ verwendet.

3.9 Multiresolutionsanalyse und Approximation

In diesem Abschnitt versuchen wir, eine adaptive MRA für die Tetrolets zu formulieren. In einem zweiten Schritt zeigen wir, dass die beste M -Term-Tetrolet-Approximation f_M unter gewissen Voraussetzungen die optimale asymptotische Abschätzung

$$\epsilon_M = \|f - f_M\|_2^2 \leq C'M^{-\alpha}$$

	monarch				cameraman			
	M	PSNR	E	$cost$	M	PSNR	E	$cost$
Tensor Haar	300	19.58	-	1.55	2500	26.29	-	0.84
Tensor 9-7	300	22.62	-	1.55	2500	28.14	-	0.84
Tetrolet	256	24.43	0.53	1.86	2048	29.17	0.56	1.26
TT16	256	23.56	0.30	1.64	2048	28.44	0.32	1.02
Relaxierte TT	256	24.51	0.32	1.66	2048	28.91	0.25	0.95
Haar-TT	256	24.24	0.43	1.77	2048	28.94	0.32	1.02
Relaxierte Haar-TT16	256	23.48	0.21	1.55	2048	28.24	0.14	0.84

Tabelle 3.3: Vergleich zwischen Tensor-Wavelettransformationen und verschiedenen Versionen der Tetrolet-Transformation bezüglich Approximationsqualität (PSNR in dB) und Speicherkosten ($cost$ in bpp).

erfüllt. Hierbei ist C' eine Konstante, die unabhängig vom Zerlegungslevel j ist, und $\alpha \in (0, 1]$ gibt die Ordnung der Hölder-Stetigkeit innerhalb der Bildregionen an.

Damit reiht sich unsere Methode in eine Anzahl von Wavelet-Techniken ein, die in letzter Zeit entwickelt wurden und ebenfalls optimale Approximationseigenschaften aufweisen. Wir hatten bereits in der Einleitung einen Überblick über repräsentative Verfahren mit ihrer jeweiligen Approximationsqualität angegeben. An dieser Stelle erinnern wir nur noch einmal an die verwandte Methode der EPWT, weil wir feststellen werden, dass man mit ähnlichen Argumenten wie in [PTI09] das Approximationsverhalten der Tetrolet-Transformation für $\alpha \in (0, 1]$ beweisen kann.

Um das asymptotische Abfallen des Approximationsfehlers ϵ_M für Tetrolets theoretisch beschreiben zu können, weichen wir das Konzept der Tetrolet-Filterbank etwas auf, indem wir auf die feste 4×4 -Blockbildung verzichten. Wir lassen also allgemeinere Tetromino-Partitionen der Bildindexmenge I zu, die nicht mehr unbedingt die dritte Bedingung im Abschnitt 3.3 erfüllen müssen. Außerdem werden wir *stetige* Tetrolets auf einem kontinuierlichen Bildgebiet $[0, 1]^2$ betrachten, die dort einer MRA-Struktur genügen.

3.9.1 Notation

Eine $L^2([0, 1]^2)$ -Version eines quadratischen, dyadischen digitalen Bildes $\mathbf{a} = (a[s])_{s \in I}$ mit der Indexmenge $I = \{0, \dots, N-1\}^2$ mit $N = 2^J$, $J \in \mathbb{N}$, kann durch

$$f(x) = \sum_{s \in I} a[s] \chi_{[0,1]^2}(2^J x - s), \quad x \in [0, 1]^2, \quad (3.13)$$

angegeben werden.

Wir setzen voraus, dass der Definitionsbereich $[0, 1]^2$ der Funktion f disjunkt in Bildgebiete Ω_i , $i = 1, \dots, K$, zerlegt werden kann, d. h. $[0, 1]^2 = \bigcup_{i=1}^K \Omega_i$. Innerhalb eines jeden

Gebietes sei f Hölder-stetig, d. h. für $\alpha \in (0, 1]$ gilt

$$|f(x) - f(y)| \leq C \|x - y\|_2^\alpha, \quad x, y \in \Omega_i \quad (3.14)$$

mit einer von x und y unabhängigen Konstante $C > 0$. Für die Bildwerte des digitalen Bildes \mathbf{a} bedeutet das wegen (3.13)

$$|a[s_1] - a[s_2]| \leq C 2^{-J\alpha} \|s_1 - s_2\|_2^\alpha, \quad s_1/2^J, s_2/2^J \in \Omega_i. \quad (3.15)$$

Bevor wir die MRA-Struktur der Tetrolets untersuchen, betrachten wir die Multiskalen-Struktur der Tetrominoes, die den Tetrolets als Träger dienen sollen. Dazu überdecken wir zunächst das Gebiet $[0, 1]^2$ disjunkt mit 2^{2J-2} Tetromino-Teilmenge \tilde{T}_n^{J-1} . Die Gebiete \tilde{T}_n^{J-1} bestehen also aus genau vier Quadraten der Größe $1/2^J \times 1/2^J$, die jeweils durch mindestens eine Kante miteinander verbunden sind. Es gilt somit

$$[0, 1]^2 = \bigcup_{n \in G(I_{J-1})} \tilde{T}_n^{J-1}, \quad (3.16)$$

dabei ist $G(I_{J-1})$ die eindimensionale Indexmenge zu $I_{J-1} = \{0, \dots, N/2 - 1\}^2$, also $G(I_{J-1}) = \{0, 1, \dots, 2^{2J-2} - 1\}$. Unter den vielen Möglichkeiten, das Bildgebiet $[0, 1]^2$ disjunkt mit Tetrominoes zu überdecken, wählen wir die Überdeckung aus, die am besten die Bildstrukturen berücksichtigt. Eine solche Überdeckung hat eine minimale Anzahl von Tetrominoes, die nicht vollständig innerhalb eines Bildgebietes Ω_i , $i = 1, \dots, K$, liegen. Das diskrete Analogon von (3.16) ist die disjunkte Überdeckung der Indexmenge I durch Teilmengen T_n^{J-1} , die die beiden Bedingungen aus Abschnitt 3.1 erfüllen:

$$I = \bigcup_{n \in G(I_{J-1})} T_n^{J-1}$$

Die diskreten Indexmengen T_n^{J-1} enthalten die vier Gitterpunkte, die in dem Gebiet \tilde{T}_n^{J-1} liegen. Wir nennen sie auch Tetromino, der Kontext wird deutlich machen ob wir das stetige Gebiet \tilde{T}_n^{J-1} oder die entsprechende diskrete Menge T_n^{J-1} meinen.

Im $(J-2)$ -ten Level fassen wir vier „feine“ Tetrominoes $T_{n_1}^{J-1}, \dots, T_{n_4}^{J-1}$, $n_i \in G(I_{J-1})$, zu einer „gröberen“ Menge T_n^{J-2} zusammen, $n \in G(I_{J-2}) = \{0, \dots, 2^{2J-4} - 1\}$, die wir wieder Tetromino nennen (auch wenn diese Teilmenge von $[0, 1]^2$ nicht mehr notwendigerweise eine Tetromino-Form hat, wie Abbildung 3.17 auf Seite 121 anschaulich zeigen wird). Jede Menge T_n^{J-2} enthält dann 16 Indizes. Dieses Vergrößern soll so geschehen, dass möglichst nur Mengen innerhalb eines Bildgebietes zusammengefasst werden. Im j -ten Level ($0 \leq j < J$) gilt also für $n \in G(I_j) = \{0, \dots, 2^{2j} - 1\}$: $\#T_n^j = 4^{J-j}$ und $T_n^j \subset I$ sowie

$$T_n^j = \bigcup_{i=1}^4 T_{n_i}^{j+1} = \bigcup_{i=1}^4 \bigcup_{s=1}^4 T_{n_{is}}^{j+2} = \bigcup_{k=1}^{16} T_k^{j+2} = \dots = \bigcup_{k=1}^{4^{J-j-1}} T_k^{j-1} \quad (3.17)$$

für passend gewählte Reihenfolgen von Indizes k .

3.9.2 Adaptive Multiresolutionsanalyse der Tetrolets

Nachdem wir die Multiskalen-Struktur der Tetrominoes für das Bildgebiet $[0, 1]^2$ untersucht haben, betrachten wir jetzt die dazugehörigen Tetrolets.

Wir wählen für $0 \leq j \leq J$ die Approximationsräume $V_j \subset L^2([0, 1]^2)$, die durch Skalierungsfunktionen erzeugt werden sollen. Im feinsten Level $J > 0$ ist

$$V_J = \text{clos}_{L^2([0,1]^2)} \text{span}\{\phi_{J,n} : n \in G(J)\},$$

dabei sind die Skalierungsfunktionen

$$\phi_{J,n} := 2^J \chi_{[0,1]^2}(2^J \cdot -G^{-1}(n)) \quad (3.18)$$

charakteristische Funktionen auf den kleinen Quadraten der Größe $1/2^J \times 1/2^J$.

Es ist also $f = f_J$ die Projektion in den feinsten Approximationsraum V_J :

$$f_J = P_{V_J} f = \sum_{n \in G(J)} c_{J,n} \phi_{J,n},$$

mit $l_j = l$ und

$$\begin{aligned} c_{J,n} = \langle f, \phi_{J,n} \rangle &= \sum_{m \in l} a[m] 2^J \langle \chi_{[0,1]^2}(2^J \cdot -m), \chi_{[0,1]^2}(2^J \cdot -G^{-1}(n)) \rangle \\ &= \sum_{m \in l} a[m] 2^J 2^{-2J} \delta_{m, G^{-1}(n)} = 2^{-J} a[G^{-1}(n)]. \end{aligned} \quad (3.19)$$

Im nächsten Level sind die „größeren“ Skalierungsfunktionen $\phi_{J-1,n}$ für $n \in G(l_{J-1})$ über die Verfeinerungsgleichung definiert

$$\phi_{J-1,n} := \frac{1}{2} \sum_{k \in G(T_n^{J-1})} \phi_{J,k}.$$

Wegen (3.18) ist dies nichts anderes als $\phi_{J-1,n} = 2^{J-1} \chi_{\tilde{T}_n^{J-1}}$. Beachte dabei, dass im Gegensatz zur klassischen MRA-Struktur der Index n hier nicht eine zweidimensionale Translation angibt, sondern die skalare Tetromino-Ordnungsnummer. Ebenso gilt für die Tetrolets mit $l = 1, 2, 3$

$$\psi_{J-1,n}^l := \sum_{k \in G(T_n^{J-1})} \epsilon[l, L(G^{-1}(k))] \phi_{J,k},$$

wobei T_n^{J-1} die n -te Tetromino-Teilmenge von l_j ist, die von dem Träger-Tetromino von $\phi_{J-1,n}$ bzw. $\psi_{J-1,n}^l$ überdeckt wird. Insbesondere ist also $\#T_n^{J-1} = 4$ für alle $n \in l_{J-1}$. Wir erinnern daran, dass die Koeffizienten $\epsilon[l, m]$, $l, m = 0, \dots, 3$, Einträge aus der orthogonalen HWT-Matrix (3.3) sind, und L die bijektive Abbildung ist, die den vier Pixeln aus T_n^{J-1} einen Wert zwischen 0 und 3 zuordnet.

Damit können wir jetzt f_j in den Tiefpassanteil f_{j-1} und den Hochpassanteil g_{j-1} zerlegen,

$$\begin{aligned} f_{j-1}(x) &= \sum_{n \in G(l_{j-1})} \langle f, \phi_{j-1,n} \rangle \phi_{j-1,n}(x), \\ g_{j-1}(x) &= \sum_{n \in G(l_{j-1})} \sum_{l=1,2,3} \langle f, \psi_{j-1,n}^l \rangle \psi_{j-1,n}^l(x). \end{aligned}$$

In einem weiteren Schritt gehen wir zu beliebigen Zerlegungsleveln über. Für $0 \leq j < J$ und $n \in G(l_j) = \{0, 1, \dots, 2^j - 1\}$ sind die Skalierungsfunktionen durch

$$\phi_{j,n} := 2^j \chi_{\tilde{T}_n^j} \quad (3.20)$$

definiert, wobei das Träger-Tetromino \tilde{T}_n^j aus der Vereinigung von vier feineren Tetromino-Mengen entstanden ist: $\tilde{T}_n^j = \bigcup_{i=1}^4 \tilde{T}_{n_i}^{j+1}$. Das bedeutet

$$2^j \chi_{\tilde{T}_n^j} = 2^j \left(\chi_{\tilde{T}_{n_1}^{j+1}} + \chi_{\tilde{T}_{n_2}^{j+1}} + \chi_{\tilde{T}_{n_3}^{j+1}} + \chi_{\tilde{T}_{n_4}^{j+1}} \right).$$

Damit bekommen wir also eine Verfeinerungsgleichung für die adaptiven Tetrolets:

$$\phi_{j,n} = \frac{1}{2} \sum_{k \in \{n_1, n_2, n_3, n_4\}} \phi_{j+1,k}.$$

Wir können (3.20) auch in der folgenden Form schreiben:

$$\phi_{j,n} = \frac{1}{2^{J-j}} \sum_{k \in G(T_n^j)} \phi_{J,k}. \quad (3.21)$$

Wegen $\#T_n^j = 4^{J-j}$ ist die Skalierungsfunktion $\phi_{j,n}$, $n \in G(l_j)$, also die charakteristische Funktion auf 4^{J-j} Quadraten der Größe $1/2^j \times 1/2^j$ in $[0, 1)^2$. Insbesondere ist $\phi_{0,0}$ die charakteristische Funktion auf ganz $[0, 1)^2$ wegen $4^J = \mathbb{N}^2$.

Die Tetrolets im j -ten Level, $0 \leq j < J$, werden über die Verfeinerungsgleichung konstruiert:

$$\psi_{j,n}^l = \sum_{k \in \{n_1, n_2, n_3, n_4\}} \epsilon[l, L(G^{-1}(k))] \phi_{j+1,k}, \quad l = 1, 2, 3,$$

wobei wieder $\tilde{T}_n^j = \bigcup_{i=1}^4 \tilde{T}_{n_i}^{j+1}$ für $n \in G(l_j) = \{0, 1, \dots, 2^j - 1\}$ zugrunde gelegt wird. Wegen der Orthogonalität der HWT-Matrix W in (3.3) gilt $\langle \psi_{j',n'}^l, \phi_{j,n} \rangle = 0$ für $0 \leq j, j' < J; n, n' \in G(l_j)$. Damit existiert eine orthogonale Zerlegung

$$V_j(f) \oplus W_j(f) = V_{j+1}(f)$$

mit den adaptiven Approximationsräumen

$$V_j(f) = \text{clos}_{L^2([0,1]^2)} \text{span}\{\phi_{j,n} : n \in G(l_j)\}$$

und den Waveleträumen

$$W_j(f) = \text{clos}_{L^2([0,1]^2)} \text{span}\{\psi_{j,n}^l : l = 1, 2, 3; n \in G(l_j)\},$$

die von der jeweiligen Funktion f (bzw. dem Bild α) abhängig sind.

Die orthogonale Zerlegung in Tief- und Hochpassanteile kann nun durch $f_{j+1} = f_j + g_j$ berechnet werden, mit

$$f_j(x) = \sum_{n \in G(l_j)} c_{j,n} \phi_{j,n}(x),$$

$$g_j(x) = \sum_{n \in G(l_j)} \sum_{l=1,2,3} d_{j,n}^l \psi_{j,n}^l(x).$$

Hierbei sind die Koeffizienten gegeben durch

$$c_{j,n} = \langle f, \phi_{j,n} \rangle = \frac{1}{2} \sum_{k \in \{n_1, n_2, n_3, n_4\}} \langle f, \phi_{j+1,k} \rangle = \frac{1}{2} \sum_{k \in \{n_1, n_2, n_3, n_4\}} c_{j+1,k},$$

$$d_{j,n}^l = \langle f, \psi_{j,n}^l \rangle = \sum_{k \in \{n_1, n_2, n_3, n_4\}} \epsilon[l, L(G^{-1}(k))] \langle f, \phi_{j+1,k} \rangle = \sum_{k \in \{n_1, n_2, n_3, n_4\}} \epsilon[l, L(G^{-1}(k))] c_{j+1,k},$$

für $l = 1, 2, 3$ und jedes Tetromino $\tilde{T}_n^j = \bigcup_{i=1}^4 \tilde{T}_{n_i}^{j+1}$, $n \in G(l_j)$. Mithilfe von (3.21) erhält man für die Tiefpass-Koeffizienten die Darstellung

$$c_{j,n} = \langle f, \phi_{j,n} \rangle = \frac{1}{2^{J-j}} \sum_{k \in G(T_n^j)} \langle f, \phi_{J,k} \rangle = \frac{1}{2^{J-j}} \sum_{k \in G(T_n^j)} c_{J,k}. \quad (3.22)$$

3.9.3 Approximationseigenschaften

Die betragsmäßige Größe der Tetrolet-Koeffizienten hängt im Wesentlichen von der Wahl der Träger-Tetrominoes ab. Um möglichst viele verschwindende Koeffizienten zu bekommen, müssen wir eine adaptive Tetromino-Partition des Bildes finden, die die lokalen Bildstrukturen berücksichtigt. Ein Tetromino, das vollständig innerhalb einer regulären Bildregion liegt, erzeugt einen betragsmäßig kleinen Tetrolet-Koeffizient, während Tetrominoes, die Bildkanten enthalten, im Allgemeinen einen betragsmäßig großen Koeffizient liefern, weil hier die Hölder-Bedingung (3.14) nicht gilt.

Analog zu [PTI09] stellen wir deshalb folgende Bedingungen an unsere Tetromino-Partitionen in den verschiedenen Leveln der Transformation:

- (a) **Gebietsbedingung:** Im j -ten Zerlegungslevel, $j = J - 1, \dots, 0$, sollen höchstens $C_1 K$ Tetrominoes existieren, die nicht vollständig in einem Bildgebiet Ω_i , $i = 1, \dots, K$, liegen. Dabei ist C_1 unabhängig von j und J .
- (b) **Durchmesserbedingung:** Im j -ten Zerlegungslevel, $j = J - 1, \dots, 0$, sollen unter den Tetrominoes, die vollständig in einem Bildgebiet liegen, höchstens C_2 Tetrominoes sein, die nicht die Bedingung

$$\text{diam } T_n^j := \max_{s_1, s_2 \in T_n^j} \|s_1 - s_2\|_2 \leq D \cdot 2^{J-j} \quad (3.23)$$

für eine feste Konstante $D < \infty$ erfüllen. Dabei sind C_2 und D unabhängig von j und J .

Tetrolets, deren Tetromino-Träger innerhalb eines Bildgebietes liegen und gleichzeitig (3.23) erfüllen, nennen wir *Typ-1-Tetrolets*. Die übrigen Tetrolets heißen *Typ-2-Tetrolets*. Die Anzahl der Typ-2-Tetrolets in jedem Level ist durch $C_1K + C_2$ beschränkt.

Bemerkung 3.4. Die bestmögliche Konstante D in (3.23) liefert die Verwendung von *O-Tetrominoes*, denn in diesem Fall ist $\text{diam } T_n^{j-1} = \sqrt{2}$, also $D = 1/\sqrt{2}$. Der ungünstigste Fall tritt bei der Verwendung von *I-Tetrominoes* auf, denn es gilt $\text{diam } T_n^{j-1} = 3$, also $D = 3/2$.

Anhand eines ausführlichen Beispiels wollen wir uns vergegenwärtigen, was die beiden Bedingungen für ein Bild bedeuten.

Beispiel 3.5. Betrachte das Bild der Größe 16×16 , das in Abbildung 3.17(oben) zu sehen ist. Es besteht aus $K = 3$ Bildregionen: Kreis, schräger Balken und Hintergrund. Innerhalb der Regionen soll die Bildfunktion Hölder-stetig gemäß (3.15) sein. Wir zeigen, dass es Tetromino-Partitionen für die $J = 4$ Zerlegungslevel gibt, die die beiden oben genannten Bedingungen mit möglichst kleinen Konstanten C_1, C_2, D erfüllen.

Im Level $j = J - 1 = 3$ überdecken wir das 16×16 -Bild adaptiv mit 64 Tetrominoes T_n^3 , $n = 0, \dots, 63$, wie in Abbildung 3.17(mitte links) dargestellt. Es gibt nur zwei Tetrominoes, nämlich Nr. 27 und 53, die nicht vollständig in einem Bildgebiet liegen. Also gilt $C_1 = 2/3$. Damit die Durchmesserbedingung (3.23) auch für das I-Tetromino Nr. 35 erfüllt wird, wählen wir $D = 3/2$. Wir haben damit dann $C_2 = 0$.

Aus jeweils vier feinen Tetrominoes werden gröbere Tetromino-Mengen T_n^2 , $n = 0, \dots, 15$, gebildet, so dass die Tetrolet-Approximation f_2 entsteht, bei der wir folgendes beobachten: Die Mengen T_n^2 haben im Allgemeinen keine Tetrominoform mehr. Außerdem müssen sie nicht mehr zusammenhängend sein, wie T_9^2 deutlich macht. Darüber hinaus stellen wir fest, dass es drei Tetrominoes (Nr. 9, 10, 14) gibt, die nicht der Gebietsbedingung entsprechen, also $C_1 = 1$. Wählen wir $D = \sqrt{13}/2 \approx 1,8027$, so genügen alle Tetrominoes bis auf T_3^2 der Durchmesserbedingung, und wir erhalten $C_2 = 1$. Für T_3^2 gilt $\text{diam } T_3^2 = \sqrt{68}$, d. h. die Durchmesserbedingung wäre nur für $D \geq \sqrt{17}/2$ erfüllt.

Gemäß der Bildgeometrie werden wieder jeweils vier Tetromino-Mengen vereinigt, so dass im nächsten Level $j = 1$ die Tetrolet-Approximation f_1 nur noch aus vier charakteristischen Funktionen auf den Tetrominoes T_n^1 , $n = 0, \dots, 3$, besteht. Es gilt offensichtlich $C_1 = 2/3$. Wir berechnen leicht $\text{diam } T_0^1 = \sqrt{274}$, $\text{diam } T_1^1 = \sqrt{160}$, $\text{diam } T_2^1 = \sqrt{170}$, $\text{diam } T_3^1 = \sqrt{100}$. Wählen wir also $D = \sqrt{170}/8 \approx 1,6298$ erfüllt nur T_0^1 die Durchmesserbedingung nicht, wir haben also wieder $C_2 = 1$.

Schließlich erhalten wir die konstante Funktion f_0 , die den Mittelwert des Bildes repräsentiert. Es gilt $C_1 = 1$ und $\text{diam } T_0^0 = \sqrt{450}$. Damit haben wir für $D = \sqrt{450}/16 \approx 1.3258$ sogar wieder $C_2 = 0$.

Als levelunabhängige Konstanten können wir dann $C_1 = C_2 = 1$ und $D = \sqrt{13}/2$ wählen.

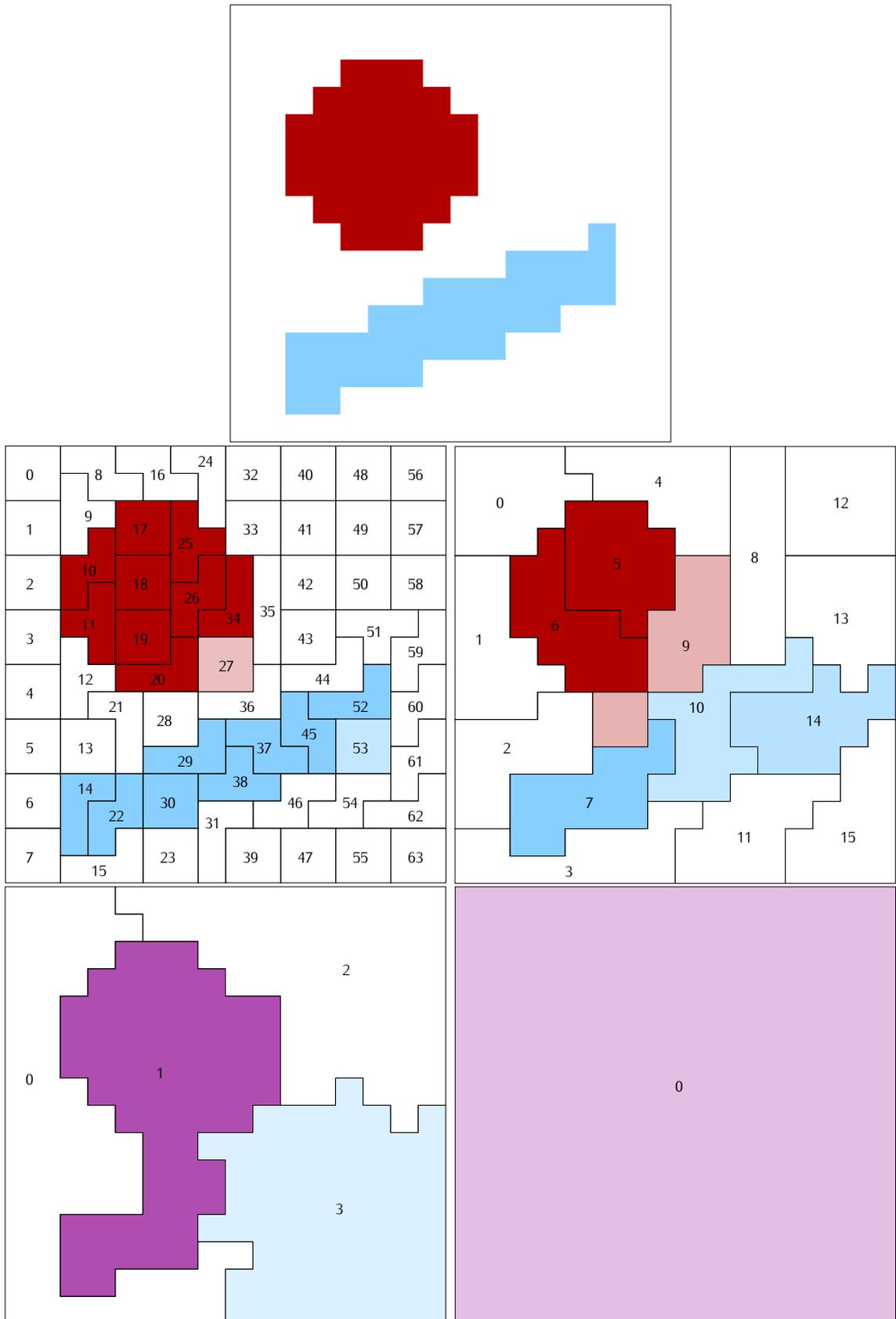


Abbildung 3.17: Bildfunktion f eines 16×16 -Bildes (oben) und ihre Tetrolet-Approximationen f_3, f_2, f_1 und f_0 .

Im zweiten Teil des Beispiels illustrieren wir anhand von Abbildung 3.18 die Vorgehensweise des Zerlegungs-Algorithmus, der die Tetrolet-Approximationen aus Abbildung 3.17 lieferte. In Abbildung 3.18(links) sehen wir die Tetromino-Partition für das Eingangsbild. Aus diesen 64 Tetrominoes erhalten wir ein neues Tiefpassbild, indem wir die Tiefpassanteile (Mittelwerte) der Tetromino-Gebiete spaltenweise in eine 8×8 -Matrix schreiben. Dies überdecken wir wieder mit Tetrominoes unter Berücksichtigung der Bildgeometrie. Wie bereits oben erwähnt rechtfertigt diese Vorgehensweise die Bezeichnung „Tetromino“ für die Mengen T_n^j mit $0 \leq j < J - 1$, auch wenn sie keine Tetrominoform mehr besitzen (siehe Abbildung 3.17). Auch hier erkennen wir wieder gut, dass die Tetrominoes Nr. 9, 10 und 14 nicht der Gebietsbedingung entsprechen und die entsprechenden Wavelets $\psi_{2,9}^l, \psi_{2,10}^l$ und $\psi_{2,14}^l$ mit $l = 1, 2, 3$ Typ-2-Tetrolets sind.

Die Tiefpasswerte der 16 Tetrominoes werden im Level $j = 1$ in eine 4×4 -Matrix eingetragen. Die Tetromino-Partition führt hier zu einem 2×2 -Tiefpassbild im Level $j = 0$, das schließlich nur noch mit einem einzigen O-Tetromino überdeckt werden kann.

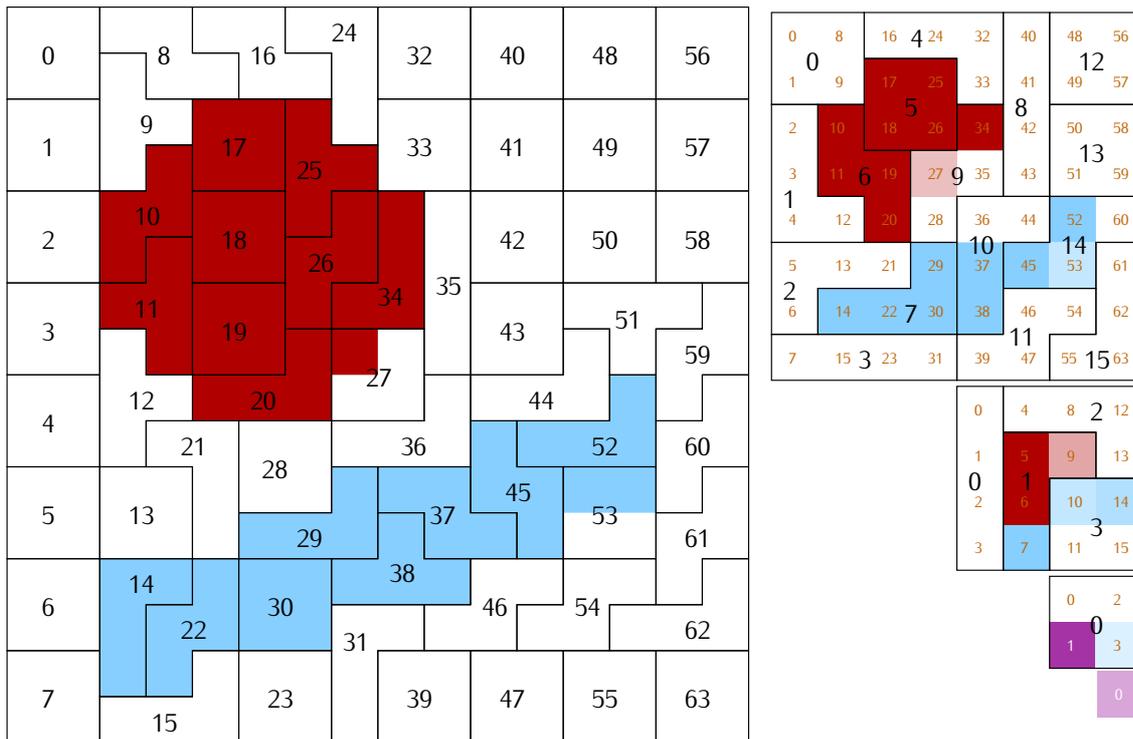


Abbildung 3.18: Adaptive Tetromino-Überdeckung eines 16×16 -Bildes und seiner Tiefpassanteile.

Lemma 3.6. Für die Tetrolet-Koeffizienten im j -ten Level, $0 \leq j < J$, gelten folgende Abschätzungen:

1. $|d_{j,n}^l| \leq CD^\alpha \cdot 2^{-j(\alpha+1)-1}$ für Typ-1-Tetrolets.
2. $|d_{j,n}^l| \leq \tilde{C} \cdot 2^{-(j+1)}$ für Typ-2-Tetrolets.

Dabei ist C die Hölder-Konstante aus (3.14), D die Durchmesser-Konstante aus (3.23) und \tilde{C} eine positive, endliche Konstante.

Beweis. Es gilt für $0 \leq j < J$ o. B. d. A.

$$|d_{j,n}^l| = \left| \sum_{k \in \{n_1, n_2, n_3, n_4\}} \epsilon[l, L(G^{-1}(k))] c_{j+1,k} \right| \leq \frac{1}{2} (|c_{j+1,n_1} - c_{j+1,n_2}| + |c_{j+1,n_3} - c_{j+1,n_4}|),$$

basierend auf der Notation $\tilde{T}_n^j = \bigcup_{i=1}^4 \tilde{T}_{n_i}^{j+1}$ und mit passend gewählter Reihenfolge n_1, n_2, n_3, n_4 . Wir wenden (3.22) an und erhalten

$$|d_{j,n}^l| \leq \frac{1}{2^{2^{j+1}}} \left(\left| \sum_{k \in G(T_{n_1}^{j+1})} c_{J,k} - \sum_{k \in G(T_{n_2}^{j+1})} c_{J,k} \right| + \left| \sum_{k \in G(T_{n_3}^{j+1})} c_{J,k} - \sum_{k \in G(T_{n_4}^{j+1})} c_{J,k} \right| \right).$$

Wegen $\#T_n^{j+1} = 4^{J-j-1}$ besteht jede Summe aus 4^{J-j-1} Summanden. Mit $m_i \in G(T_{n_i}^{j+1})$, $i = 1, 2, 3, 4$, können wir also o. B. d. A. abschätzen

$$|d_{j,n}^l| \leq \frac{1}{2^{J-j}} 4^{J-j-1} \left(\max_{m_1, m_2} |c_{J, m_1} - c_{J, m_2}| + \max_{m_3, m_4} |c_{J, m_3} - c_{J, m_4}| \right).$$

Nun sei $s_i = G^{-1}(m_i)$ die zweidimensionale Ortskoordinate des Bildpixels $a[s_i]$, $i = 1, 2, 3, 4$. Dann gilt mit (3.19)

$$|d_{j,n}^l| \leq 2^{-j-2} \left(\max_{s_1, s_2} |a[s_1] - a[s_2]| + \max_{s_3, s_4} |a[s_3] - a[s_4]| \right). \quad (3.24)$$

Zu 1. Für Typ-1-Tetrolets nutzen wir nun die Hölder-Stetigkeit (3.15) und erhalten

$$\begin{aligned} |d_{j,n}^l| &\leq 2^{-j-2} \left(\max_{s_1, s_2} (C 2^{-j\alpha} \|s_1 - s_2\|_2^\alpha) + \max_{s_3, s_4} (C 2^{-j\alpha} \|s_3 - s_4\|_2^\alpha) \right) \\ &= 2^{-j\alpha-j-2} C \left(\max_{s_1, s_2} \|s_1 - s_2\|_2^\alpha + \max_{s_3, s_4} \|s_3 - s_4\|_2^\alpha \right). \end{aligned}$$

Die Durchmesserbedingung (3.23) liefert schließlich die gewünschte Ungleichung

$$|d_{j,n}^l| \leq 2^{-j\alpha-j-2} C \cdot 2(D 2^{j-1})^\alpha = C D^\alpha 2^{-j(\alpha+1)-1}. \quad (3.25)$$

Zu 2. Weil s_1 und s_2 bei Typ-2-Tetrolets ebenso wie s_3 und s_4 i. a. aus verschiedenen Bildregionen kommen, können wir in (3.24) weder die Hölder-Stetigkeit noch die Durchmesserbedingung verwenden. Da die Bildwerte aber durch \tilde{C} beschränkt sind, erhalten wir zumindest die Satzaussage. \square

Bemerkung 3.7. Wir können die Konstante \tilde{C} in der zweiten Abschätzung von Lemma 3.6 verbessern, wenn wir bei den Typ-2-Tetrolets noch weitere Unterscheidungen vornehmen würden, weil viele Typ-2-Tetrolets einen Null-Koeffizient liefern, obwohl ihr Tetromino-Träger nicht vollständig in einem Bildgebiet liegt. Zu jedem Tetromino T_n^j , das nicht in einem Bildgebiet liegt, gibt es drei Typ-2-Tetrolet-Koeffizienten $d_{j,n}^l$ für $l = 1, 2, 3$. Wenn aber nun der Träger

derart in zwei verschiedenen Bildgebieten liegt, dass die zwei Pixel s_1 und s_2 in einem Gebiet liegen, und s_3 und s_4 in dem anderen, so können wir in (3.24) trotzdem die Hölder-Stetigkeit und die Durchmesserbedingung anwenden, und erhalten auch in diesem Fall für zwei der drei Typ-2-Koeffizienten die günstigere Typ-1-Abschätzung (3.25). Ein solches Beispiel finden wir in Abbildung 3.18 für das Träger-Tetromino Nr. 53 im feinsten Level.

Analog zu [PTI09] und in Anlehnung an [Mal89, Proposition 9.4] können wir nun die optimale M-Term-Approximation durch Tetrolets unter den oben genannten Voraussetzungen beweisen.

Satz 3.8. Sei f eine Bildfunktion wie in (3.13) beschrieben, die innerhalb der Bildregionen Hölder-stetig von der Ordnung $\alpha \in (0, 1]$ ist. Wir wenden die adaptive Tetrolet-Transformation an, so dass die zugrunde liegenden Tetromino-Partitionen in den Zerlegungsleveln die Gebiets- und Durchmesserbedingung erfüllen. Die so erhaltene beste M-Term-Approximation f_M von $f \in L^2([0, 1]^2)$ liefert für den Approximationsfehler die Abschätzung

$$\epsilon_M = \|f - f_M\|_2^2 \leq C' M^{-\alpha}.$$

Beweis. Ordne alle Tetrolet-Koeffizienten $d_{j,n}^l = \langle f, \psi_{j,n}^l \rangle$ von f in absteigender Reihenfolge in einen Vektor, d. h. mit dem Tripel-Index $\mu = (j, n, l)$ erhalten wir $(d_\mu)_{\mu=0}^{2^{2j}-1}$ mit $|d_\mu| \geq |d_{\mu+1}|$. Dabei ist $d_0 = c_{0,0}$ der einzig verbleibende Tiefpass-Koeffizient.

1. Wir zeigen zunächst mithilfe von Lemma 3.6, dass die Reihe $\sum_{\mu=0}^{2^{2j}-1} |d_\mu|^p$ beschränkt ist für alle p mit $\frac{1}{2} < \frac{1}{p} < \frac{\alpha+1}{2}$.

In jedem Level $0 \leq j < J$ gibt es maximal $3 \cdot 2^{2j}$ Typ-1-Tetrolets, so dass gilt

$$\sum_{d_\mu \text{ vom Typ 1}} |d_\mu|^p \leq \sum_{j=0}^{J-1} 3 \cdot 2^{2j} \cdot |CD^\alpha 2^{-j(\alpha+1)-1}|^p = (CD^\alpha/2)^p \cdot 3 \sum_{j=0}^{J-1} 2^{2j} \cdot 2^{-j(\alpha+1)p}.$$

Für $J \rightarrow \infty$ ist die Reihe endlich, wenn $p\alpha + p - 2 > 0$, d. h. wenn $\frac{1}{p} < \frac{\alpha+1}{2}$.

Nach Voraussetzung gibt es in jedem Level höchstens $C_1K + C_2$ Typ-2-Tetrolets, so dass für alle p gilt

$$\begin{aligned} |d_0|^p + \sum_{d_\mu \text{ vom Typ 2}} |d_\mu|^p &\leq |d_0|^p + (C_1K + C_2) \cdot \sum_{j=0}^{J-1} |\tilde{C} 2^{-(j+1)}|^p \\ &= |d_0|^p + (C_1K + C_2) (\tilde{C}/2)^p \sum_{j=0}^{J-1} 2^{-jp} \\ &\leq |d_0|^p + (C_1K + C_2) (\tilde{C}/2)^p \frac{1}{1 - 2^{-p}} < \infty. \end{aligned}$$

2. Wir betrachten die aus k Summanden bestehenden Partialsummen

$$S_k := \sum_{\mu=k}^{2k-1} |d_\mu|^p$$

für $0 < k \leq 2^{2J-1}$. Diese Partialsummen sind für beliebiges $J > 1$ beschränkt, weil wir oben gezeigt haben, dass $\sum_{\mu=0}^{2^{2J}-1} |d_{\mu}|^p$ beschränkt ist. Weil die Koeffizienten nach ihrer betragsmäßigen Größe geordnet sind, gilt für $k \leq \mu \leq 2k-1$: $|d_{2k}| \leq |d_{2k-1}| \leq |d_{\mu}|$. Damit können wir die Partialsummen S_k für $1 < p < 2$ nach unten abschätzen:

$$S_k \geq k|d_{2k-1}|^p \geq k|d_{2k}|^p.$$

Das bedeutet $|d_{2k}| \leq |d_{2k-1}| \leq \left(\frac{S_k}{k}\right)^{1/p}$, bzw.

$$|d_k| \leq \left(\frac{S_{\lceil k/2 \rceil}}{k/2}\right)^{1/p} = \left(\frac{2}{k} S_{\lceil k/2 \rceil}\right)^{1/p}.$$

Nun sind wir in der Lage, den Approximationsfehler abzuschätzen; es gilt für $1 < M < 2^{2J}-1$:

$$\epsilon_M = \sum_{\mu=M+1}^{2^{2J}-1} |d_{\mu}|^2 \leq \sum_{\mu=M+1}^{2^{2J}-1} \left(\frac{2}{\mu}\right)^{2/p} (S_{\lceil \mu/2 \rceil})^{2/p} \leq 2^{2/p} \left(\sup_{\mu>M} |S_{\lceil \mu/2 \rceil}|\right)^{2/p} \sum_{\mu=M+1}^{\infty} \mu^{-2/p}.$$

Die letzte Summe ist die Untersumme des Integrals $\int_M^{\infty} x^{-2/p} dx$, denn die Funktion $x^{-2/p}$ ist monoton fallend für $x > 0$. Damit können wir die Summe nach oben hin abschätzen durch

$$\sum_{\mu=M+1}^{\infty} \mu^{-2/p} \leq \int_M^{\infty} x^{-2/p} dx = \frac{1}{2/p-1} M^{1-2/p},$$

so dass wir insgesamt für den Approximationsfehler die Ungleichung

$$\epsilon_M \leq 2^{2/p} \sup_{\mu>M} |S_{\lceil \mu/2 \rceil}|^{2/p} \frac{1}{2/p-1} M^{1-2/p}$$

erhalten, was für $p \rightarrow 2/(\alpha+1)$ mit $C' = \frac{2^{\alpha+1}}{\alpha} \sup_{\mu>M} |S_{\lceil \mu/2 \rceil}|^{\alpha+1}$ die Behauptung liefert. \square

Kapitel 4

Postprocessing-Methode für gerichtete Haarwavelets

Die in dieser Arbeit vorgestellten, gerichteten Wavelets sind durchweg Haarwavelets (mit Ausnahme der stetigen Wavelets im Abschnitt 2.5). Ihr kleiner, kompakter Träger führt in der Anwendung nur zu sehr geringen Gibbs-Phänomenen, die vernachlässigbar sind. Die bisher erzielten Ergebnisse in der Bildverarbeitung leiden vielmehr durch die fehlende Regularität der Basisfunktionen. Deshalb schlagen wir in diesem Kapitel eine Postprocessing-Methode vor, die dieses Problem behebt. Dadurch lässt sich die Approximationsqualität für glatte Bilder wesentlich verbessern.

Die in unserer Arbeit [KM10] vorgestellte Hybridmethode kombiniert die adaptive Tetrolet-Transformation mit einem Variationsansatz. Das dabei verwendete iterative Minimierungsschema minimiert die *anisotrope* totale Variation (ATV), um die stückweise konstante Tetrolet-Approximation zu glätten unter Erhalt der wichtigen Bildkanten. Auf diese Weise werden auch die Blocking-Effekte der Tetrolet-Transformation beseitigt.

Hybridmethoden Um in der Anwendung exzellente Ergebnisse zu erzielen, bedient man sich in der Praxis häufig Hybridmethoden, die die Vorteile verschiedener Verfahren verbinden. In der Bildverarbeitung sind in den letzten Jahren Kombinationen von einer Wavelettransformation mit Diffusions- und Variationstechniken untersucht worden [SWB⁺04, WWS05]. Diese Verfahren sind bisher insbesondere zur Bildentstörung eingesetzt worden und basieren auf einer gerichteten Wavelettransformation wie der Curvelet- oder Shearlettransformation [MP07, ELC09]. Einer der populärsten Ansätze unter der großen Anzahl der Variations-techniken ist die Minimierung von Funktionalen, die die totale Variation (TV) des Bildes als Glättungsterm verwenden.

TV-Minimierung in der Bildverarbeitung Bekanntlich lässt sich ein gegebenes, verrauschtes Bild $f : \Omega \rightarrow \mathbb{R}$ in dem rechteckigen Bildgebiet Ω entstören, indem man das konvexe

Funktional

$$\inf_{u \in L^2(\Omega)} \left\{ \frac{1}{2} \int_{\Omega} (u - f)^2 dx + \lambda \int_{\Omega} \Phi(|\nabla u|^2) dx \right\}$$

mit dem Regularisierungsparameter $\lambda > 0$ minimiert. Der erste Term bewirkt, dass das gesuchte Bild u ähnlich zu dem gegebenen f ist, während der zweite Term des Funktionals bei geeigneter Wahl von $\Phi : [0, \infty) \rightarrow \mathbb{R}$ für eine kantenerhaltende Glättung sorgt. Dabei ist der Gradient im distributionellen Sinne zu verstehen. In ihrer wegbereitenden Arbeit [ROF92] stellten Rudin, Osher und Fatemi 1992 das sogenannte ROF-Modell zur Entstörung von Bildern vor, indem sie im Regularisierungsterm die Funktion $\Phi(s^2) = |s|$ verwendeten:

$$\inf_{u \in L^2(\Omega)} \left\{ \frac{1}{2} \int_{\Omega} (u - f)^2 dx + \lambda \int_{\Omega} |\nabla u| dx \right\}.$$

Die Verwendung dieser TV-Seminorm

$$TV(f) = \int_{\Omega} |\nabla f(x)| dx \quad (4.1)$$

lässt Unstetigkeiten zu und bestraft trotzdem kleinere Oszillationen. Deshalb hat sich dieses Modell als äußerst effektiv für die Bildentstörung erwiesen: Störungen werden entfernt, aber – im Gegensatz zu einer Regularisierung mit einer L^2 -Norm – die Bildkanten in ihrer vollen Schärfe erhalten.

Die Variationsformulierung des Entstörungsproblems führt sofort zu einem Gradientenabstiegs-Verfahren. Wegen der Nicht-Differenzierbarkeit von $TV(f)$ muss man entweder mit einem Subgradienten arbeiten oder mit einer regularisierten Version der TV-Seminorm. Dazu ersetzt man häufig (4.1) durch das regularisierte Funktional

$$TV(f) = \int_{\Omega} \sqrt{|\nabla f(x)|^2 + \beta^2} dx, \quad (4.2)$$

mit einem kleinen Parameter $\beta > 0$. Wir werden beide Möglichkeiten im Verlauf dieses Kapitels erläutern. Auch in der Literatur lassen sich beide Varianten finden. Dazu verweisen wir auf die gute Überblicksarbeit von Chan et al. [CEPY06] mit den dort enthaltenen Referenzen, und erwähnen hier lediglich die Arbeiten, die mit unseren Ergebnissen in unmittelbarem Zusammenhang stehen.

Esedoglu und Osher [EO04] haben das ROF-Modell verallgemeinert, indem sie anisotrope Energien zugelassen haben. Bei der Bildrekonstruktion werden nun geometrische Bildeigenschaften berücksichtigt.

Basierend auf dieser Arbeit haben Berkels et al. [BBD⁺06] eine Variationsmethode mit speziellen Richtungsenergien zur Bildklassifikation untersucht, die auf der Berechnung der Rotationswinkel der rechteckigen Bildfiguren beruht. Steidl und Teuber haben zur kantenerhaltenden Entstörung von gedrehten und gescherten Rechtecken adaptive Diffusionstensoren und anisotrope Regularisierungsmethoden verwendet [ST09], die insbesondere die scharfen Eckpunkte der Figuren erhalten und nicht abrunden, wie es z. B. im isotropen ROF-Modell

geschieht. In ihrer aktuellen Arbeit [ST10] bedienen sie sich einer anisotropen TV-Seminorm im Regularisierungsterm, um multiplikative Störungen zu entfernen. In [CSS08] wurden anisotrope Regularisations- und Diffusionsmethoden für ein Inpainting-Verfahren benutzt, das auf einem Haarwavelet-Shrinkage basiert.

Der digitale TV-Filter von Chan, Osher und Shen [COS01] ist eine geeignete Diskretisierung des ROF-Modells und liefert aufgrund seiner Nicht-Linearität sehr gute Ergebnisse in der Bildentstörung.

Auch Ta et al. betrachten in [TBEL07] ein diskretes ATV-Funktional zur Filterung von Bildern und verwenden verschiedene Regularisierungsvarianten für die TV-Seminorm.

Chan und Zhou haben in [CZ00] einen iterativen TV-Minimierungs-Algorithmus zur Bildkompression vorgestellt, um die aus dem Wavelet-Shrinkage resultierenden Pseudo-Gibbs-Phänomene zu eliminieren.

In diesem Kapitel verallgemeinern wir eine sehr ähnliche TV-Minimierungs-Technik, die von Durand und Froment [DF03] im Rahmen der Bildentstörung vorgeschlagen wurde. Ihr Verfahren kann man als *projizierten Subgradienten-Abstieg* bezeichnen. Diese Methode ist bereits erfolgreich zur Bildentstörung angewandt worden, und zwar in Kombination mit komplexen Wavelets [Ma05], komplexen Ridgelets [MF06] und Wave-Atomen [Ma07]. Wir kombinieren dieses Postprocessing-Verfahren mit der Tetrolet-Transformation und wenden es erstmals für die Approximation von Bildern an.

4.1 Methode des projizierten Subgradienten-Abstiegs

Wir stellen zunächst den eindimensionalen, diskreten Ansatz von [DF03] dar und lassen die ausführlichen Beweise zugunsten einer besseren Lesbarkeit aus. Dann erweitern wir die Methode auf den zweidimensionalen Fall mit anisotroper totaler Variation.

Für ein gegebenes Signal $f = (f_0, \dots, f_{N-1}) \in \mathbb{R}^N$ ist die *diskrete totale Variation von f* durch

$$TV(f) = \sum_{k=0}^{N-2} |f_{k+1} - f_k|, \quad (4.3)$$

gegeben, siehe z. B. [Mal99, Kapitel 2]. Dies ist eine Diskretisierung des stetigen TV-Funktionals in (4.1), indem man den Gradienten durch Vorwärtsdifferenzen ersetzt. Wir setzen wieder $N = 2^{J'}$ mit $J' \in \mathbb{N}$ voraus und zerlegen das Signal f mit der DWT in einen Tief- und Hochpassanteil. Dazu sei f^c die kontinuierliche Funktion in dem Approximationsraum V_0 , die mit dem gegebenen diskreten Signal f verbunden ist durch

$$f^c = \sum_{k=0}^{N-1} f_k \phi_{0,k}, \quad (4.4)$$

wobei $f_k = \langle f^c, \phi_{0,k} \rangle$ ist (im Falle einer ONB-MRA). Nach J Leveln, $0 < J < J'$, erhalten wir

die Darstellung

$$f^c = \sum_{k=0}^{2^J N-1} c_{-J,k}(f^c) \phi_{-J,k} + \sum_{j=-1}^{-J} \sum_{k=0}^{2^j N-1} d_{j,k}(f^c) \psi_{j,k}$$

mit den Waveletkoeffizienten $d_{j,k}(f^c) := \langle f^c, \psi_{j,k} \rangle$ und den Approximationskoeffizienten $c_{-J,k}(f^c) := \langle f^c, \phi_{-J,k} \rangle$. Nach einem Wavelet-Shrinkage hat die rekonstruierte Funktion \tilde{f}^c die Gestalt

$$\tilde{f}^c = \sum_{k=0}^{2^J N-1} c_{-J,k}(f^c) \phi_{-J,k} + \sum_{j=-1}^{-J} \sum_{k=0}^{2^j N-1} S_\lambda(d_{j,k}(f^c)) \psi_{j,k} \quad (4.5)$$

mit dem Shrinkage-Operator S_λ . Üblicherweise wählt man auch hier für S_λ den globalen Hard-thresholding-Operator

$$S_\lambda(x) = \begin{cases} x, & |x| \geq \lambda, \\ 0, & |x| < \lambda, \end{cases}$$

mit einem geeigneten Schwellwert λ . Die Funktion in (4.5) kann schließlich wieder in V_0 dargestellt werden durch

$$\tilde{f}^c = \sum_{k=0}^{N-1} \tilde{f}_k \phi_{0,k}$$

wobei die Koeffizienten $\tilde{f}_k = \langle \tilde{f}^c, \phi_{0,k} \rangle$ das approximierte Signal $\tilde{f} = (\tilde{f}_0, \dots, \tilde{f}_{N-1}) \in \mathbb{R}^N$ von f liefern. Sei nun

$$M_{f^c, \lambda} := \{(j, k) : |d_{j,k}(f^c)| \geq \lambda\}$$

die Indexmenge der Koeffizienten, die nach der Shrinkage-Prozedur erhalten bleiben, und sei $X := \mathbb{R}^N$ die Menge der Signale der Länge N . Diese Signale haben stets eine endliche TV-Seminorm. Dann können wir einen Unterraum $U = U_{M_{f^c, \lambda}} \subset X$ definieren, der die Signale enthält, die im Wesentlichen mit \tilde{f} übereinstimmen. Genauer gesagt meint „im Wesentlichen“, dass die betragsmäßig großen Waveletkoeffizienten identisch sind. Wir definieren also:

$$U := \{u \in X : d_{j,k}(u^c) = d_{j,k}(f^c) \forall (j, k) \in M_{f^c, \lambda}, c_{-J,k}(u^c) = c_{-J,k}(f^c) \forall k\},$$

wobei u^c die kontinuierliche Funktion zum diskreten Signal u ist im Sinne von (4.4). Die Signale $u \in U$ unterscheiden sich also von dem gegebenen \tilde{f} höchstens in den betragsmäßig kleinen Koeffizienten, die eine geringe Information von f beinhalten. Es gilt also

$$U = \{\tilde{f}\} + V,$$

mit dem Komplementärraum

$$V = \{v \in X : d_{j,k}(v^c) = 0 \forall (j, k) \in M_{f^c, \lambda}, c_{-J,k}(v^c) = 0 \forall k\}. \quad (4.6)$$

Die Idee ist nun, unter den Signalen $u \in U$ das Signal zu finden, das minimale TV-Seminorm besitzt. Dazu lösen wir das Variationsproblem

Problem 4.1. Finde $u^* \in U$ derart, dass $TV(u^*) = \min_{u \in U} TV(u)$.

Wegen der Konvexität des TV-Funktionalen besitzt das Problem in dem Untervektorraum U eine Lösung. Das führt uns schließlich zu der Hauptaussage von Durand und Froment, die besagt, dass eine Lösung mit Hilfe eines *Subgradienten-Abstiegs-Verfahren* und einer Projektion auf den linearen Raum V ermittelt werden kann. Diese Aussage lässt sich direkt auf den zweidimensionalen Fall übertragen.

Satz 4.2. *Eine Approximation der Lösung u^* des Problems 4.1 erhält man mittels des Algorithmus*

$$u_{k+1} = u_k - t_k P_V(g_{TV}(u_k)), \quad (4.7)$$

mit der Orthogonalprojektion P_V auf den Raum V und einem Subgradienten $g_{TV}(u_k)$ von TV an der Stelle u_k . Der Zeitschritt t_k genügt dabei den Bedingungen

$$t_k > 0, \quad \lim_{k \rightarrow \infty} t_k = 0, \quad \sum_{k \in \mathbb{N}} t_k = \infty, \quad (4.8)$$

um Konvergenz zu gewährleisten.

4.2 ATV-Minimierung

Die Idee von Durand und Froment haben wir in [KM10] verallgemeinert, indem wir zum einen eine anisotrope totale Variation verwendet und zum anderen den zweidimensionalen Fall betrachtet haben.

Sei nun $f := (f_p)_{p \in I}$ eine diskrete, zweidimensionale Datenmenge. Dann ersetzen wir die eindimensionale TV in (4.3) durch die *diskrete zweidimensionale TV*

$$TV(f) = \sum_{p \in I} \sum_{p' \in N(p)} |f_{p'} - f_p|. \quad (4.9)$$

für eine symmetrische Nachbarschaft $N(p)$ des Pixels p . Typische Nachbarschaften sind die 4er-Nachbarschaft $N_4(p) = \{p' \in I : \|p - p'\|_2^2 = 1\}$ oder die 8er-Nachbarschaft $N_8(p) = \{p' \in I : \|p - p'\|_2^2 \leq 2\}$. Die *diskrete ATV* ist die gewichtete Summe

$$ATV(f) = \sum_{p \in I} \sum_{p' \in N(p)} \sqrt{w_{p,p'}} |f_{p'} - f_p|, \quad (4.10)$$

vgl. [TBEL07], wobei die Gewichte $w_{p,p'} \geq 0$ eine anisotrope Glättung bewirken, indem die lokale Bildgeometrie beachtet wird. Weiter unten erläutern wir die Wahl geeigneter Gewichte.

Nun berechnen wir mit der Kettenregel den benötigten Subgradienten g_{ATV} des aniso-

tropen Funktionals $ATV(f)$ an der Stelle $p \in I$,

$$\begin{aligned}
g_{ATV}(f)|_p &= \frac{\partial ATV(f)}{\partial f_p} \\
&= \sum_{p' \in N(p)} \sqrt{w_{p,p'}} \operatorname{sgn}(f_p - f_{p'}) + \sum_{p' \in N(p) \setminus p} \sqrt{w_{p',p}} \operatorname{sgn}(f_p - f_{p'}) \\
&= 2 \sum_{p' \in N(p)} \sqrt{w_{p,p'}} \operatorname{sgn}(f_p - f_{p'}), \tag{4.11}
\end{aligned}$$

mit symmetrischen Gewichten $w_{p',p} = w_{p,p'}$. Um schließlich Satz 4.2 anwenden zu können, müssen wir zeigen, dass die hergeleitete Formel tatsächlich ein Subgradient von $ATV(f)$ ist.

Lemma 4.3. *Der Ausdruck in (4.11) ist ein Subgradient von ATV , d. h. es gilt für alle $N \times N$ -Bilder $h = (h_p)_{p \in I}$, $f = (f_p)_{p \in I}$*

$$ATV(h) \geq ATV(f) + \langle g_{ATV}(f), h - f \rangle.$$

Beweis. Es gilt $|y| \geq |x| + \operatorname{sgn}(x)(y - x)$ für alle $x, y \in \mathbb{R}$. Setze $y := \sqrt{w_{p,p'}}(h_{p'} - h_p)$ und $x := \sqrt{w_{p,p'}}(f_{p'} - f_p)$. Dann bekommen wir

$$\sqrt{w_{p,p'}}|h_{p'} - h_p| \geq \sqrt{w_{p,p'}}|f_{p'} - f_p| + \operatorname{sgn}(f_{p'} - f_p) \cdot \sqrt{w_{p,p'}}(h_{p'} - h_p - f_{p'} + f_p).$$

Wir summieren jetzt über $p \in I$ und $p' \in N(p)$ und erhalten

$$\begin{aligned}
ATV(h) &\geq ATV(f) + \sum_{p \in I} \sum_{p' \in N(p)} \operatorname{sgn}(f_{p'} - f_p) \cdot \sqrt{w_{p,p'}}((h_{p'} - f_{p'}) - (h_p - f_p)) \\
&= ATV(f) + \sum_{p \in I} 2 \sum_{p' \in N(p)} \sqrt{w_{p,p'}} \operatorname{sgn}(f_{p'} - f_p) \cdot (h_p - f_p) \\
&= ATV(f) + \sum_{p \in I} g_{ATV}(f)|_p \cdot (h - f)|_p \\
&= ATV(f) + \langle g_{ATV}(f), h - f \rangle.
\end{aligned}$$

□

Zusammenfassend erhalten wir also aus (4.7) das verallgemeinerte, anisotrope Iterationsverfahren

$$f^{(k+1)} = f^{(k)} - t_k P_V(g_{ATV}(f^{(k)})),$$

wobei die Folge der Schrittweiten $(t_k)_{k \in \mathbb{N}}$ wieder den Bedingungen (4.8) genügen muss. Wegen (4.6) kann die Projektion P_V in den Unterraum V für ein gegebenes Signal $g \in X$ mittels der inversen Wavelettransformation berechnet werden. Sei g^c die zu g gehörende kontinuierliche Funktion im Sinne von (4.4) und $M_{g^c, \lambda}$ die gegebene Indexmenge der signifikanten Koeffizienten. Dann erhält man aus

$$\sum_{(j,k) \in M_{g^c, \lambda}} d_{j,k}(g^c) \psi_{j,k}$$

eine Funktion, die aus den insignifikanten Koeffizienten rekonstruiert wird, also aus

$$c_{-j,k}(g^c) := 0, \quad d_{j,k}(g^c) = \begin{cases} 0, & (j,k) \in M_{g^c,\lambda}, \\ d_{j,k}(g^c), & \text{sonst.} \end{cases} \quad (4.12)$$

Stellen wir diese Funktion dann bezüglich der Basisfunktionen $\phi_{0,k}$ im Startraum V_0 dar, so liefern uns die Koeffizienten die diskreten Werte der Projektion $P_V(g)$.

Als Wavelettransformation wählen wir die adaptive Tetrolet-Transformation aus dem vorigen Kapitel. Insgesamt ergibt sich dann für die Hybridmethode folgender Algorithmus.

Algorithmus: ATV-Minimierung der Tetrolet-Approximation

Input: Bild $f = (f_p)_{p \in I}$ mit Indexmenge $I = \{0, \dots, N-1\}^2$, $N = 2^J$, $J \in \mathbb{N}$.

1. Berechne die diskreten Tetrolet-Koeffizienten.
2. Wende Shrinkage an und markiere die Indizes der verbliebenden Koeffizienten in $M_{f,\lambda}$.
3. Rekonstruiere \tilde{f} aus den verbliebenen Tetrolet-Koeffizienten mittels der inversen Tetrolet-Transformation.
4. Setze $f^{(0)} = \tilde{f}$. Für $k = 0, 1, \dots$ minimiere die ATV-Seminorm der $f^{(k)}$ durch das folgende iterative Schema
 - (i) Berechne den Subgradienten $g_{ATV}(f^{(k)})$ von $f^{(k)}$ mit (4.11).
 - (ii) Zerlege $g_{ATV}(f^{(k)})$ in seine Tetrolet-Koeffizienten $d_{j,k}(g_{ATV}(f^{(k)}))$ und $c_{-j,k}(g_{ATV}(f^{(k)}))$.
 - (iii) Berechne $P_V(g_{ATV}(f^{(k)}))$ mittels der inversen Tetrolet-Transformation aus den Koeffizienten gemäß (4.12).
 - (iv) Setze $f^{(k+1)} = f^{(k)} - t_k P_V(g_{ATV}(f^{(k)}))$.

Output: Bild $f^{(k)}$.

Jetzt erläutern wir die Wahl der Gewichte $\sqrt{w_{p,p'}}$ und ihre Auswirkungen. Die ATV in (4.10) war als gewichtete Summe von benachbarten Funktionswerten (im Falle von Bildern: Bildwerten) definiert. Dabei ist es wünschenswert, dass solche Pixel p' , die einen ähnlichen Pixelwert aufweisen wie das Zentrumspixel p , einen größeren Einfluss in der Mittelwertbildung haben als Pixel, deren Pixelwert $f_{p'}$ völlig verschieden von f_p ist. Mit anderen Worten: Befinden wir uns also im Pixel p an einer Bildkante, so sollen die Pixel p' , die zwar räumlich benachbart sind aber auf der anderen Seite der Kante liegen, ein sehr kleines Gewicht bekommen. Für die Gewichte $w_{p,p'}$ heißt das, dass sie von der Differenz der Bildwerte $|f_p - f_{p'}|$ abhängig sein müssen. Genauer gesagt soll der Glättungsprozess über Kanten hinweg verhindert werden (also dort wo $|f_p - f_{p'}|$ groß ist) während er in flachen Bildregionen (dort wo $|f_p - f_{p'}|$ klein ist) verstärkt werden soll. Deshalb fordern wir, dass $w_{p,p'}$ monoton fallend für $f_p - f_{p'} \geq 0$ sein soll, und zwar $w_{p,p'} \rightarrow 0$ für $|f_p - f_{p'}| \rightarrow \infty$. Außerdem ist es vernünftig $w_{p,p'}$ symmetrisch zu wählen, d. h. $w_{p,p'} = w_{p',p}$ für alle $p, p' \in I$.

Es gibt verschiedene Möglichkeiten die Gewichte auf diese Art und Weise zu wählen, siehe z. B. [DMW07] und die dort angeführten Referenzen. Eine populäre Wahl, die sich als

effektiv erwiesen hat, ist die Wahl der Gewichte des bilateralen Filters [TM98]. Die Gewichte bestehen aus zwei Komponenten, die zum einen die räumliche Nähe und zum anderen die Ähnlichkeit des Pixelwertes berücksichtigen:

$$w_{p,p'} = w_{p,p'}(f_p - f_{p'}, p - p') = \exp\left(-\frac{1}{\sigma_i^2}(f_p - f_{p'})^2\right) \cdot \exp\left(-\frac{1}{\sigma_s^2}\|p - p'\|_2^2\right). \quad (4.13)$$

Beide Komponenten sind Gauß'sche Glockenkurven, die symmetrisch und positiv sind, und für betragsmäßig große Argumente gegen Null streben. Der erste Term wird *Intensitäts-Gewicht* genannt, er bestraft Pixelwerte $f_{p'}$ aus einem anderen Bildgebiet und stellt somit ein Kanten erhaltendes Glätten sicher. Der zweite Term ist das *räumliche Gewicht*, das räumlich weit entfernten Nachbarpixeln p' einen geringeren Einfluss einräumt. Die Parameter $\sigma_i, \sigma_s > 0$ sind Kontrollparameter, um das Abfallen der beiden Gaußkurven zu bestimmen.

Zum Schluss zeigen wir noch, dass es eine zweite Möglichkeit gibt, ein diskretes TV-Funktional zu definieren. Wie erwähnt war unser Ansatz in (4.9) eine Diskretisierung des stetigen TV-Funktional in (4.1) und brachte die Notwendigkeit der Verwendung eines Subgradienten mit sich. Eine Diskretisierung des *regularisierten* TV-Funktional in (4.2) führt zu der anisotropen diskreten TV der Form

$$ATV(f) = \sum_{p \in I} \sqrt{\sum_{p' \in N(p)} w_{p,p'} |f_{p'} - f_p|^2 + \beta^2}, \quad (4.14)$$

vgl. [TBEL07]. In diesem Fall benötigen wir keinen Subgradienten $g_{ATV}(f)$, weil die Differenzierbarkeit überall gewährleistet ist. Wir erhalten den Gradienten

$$\nabla_f ATV(f)|_p = \sum_{p' \in N(p)} w_{p',p} (Z(p, p') + Z(p', p'')) (f_p - f_{p'})$$

mit $Z(p_1, p_2) = \left(\sum_{p_2 \in N(p_1)} w_{p_1, p_2} (f_{p_1} - f_{p_2})^2 + \beta^2\right)^{-1/2}$ für $p_1, p_2 \in I$. Diese Gradientenformel an der Stelle $p \in I$ ergibt sich, indem wir erneut die Kettenregel anwenden:

$$\begin{aligned} \nabla_f ATV(f)|_p &= \frac{\partial ATV(f)}{\partial f_p} = \frac{\partial}{\partial f_p} \sqrt{\sum_{p' \in N(p)} w_{p,p'} (f_p - f_{p'})^2 + \beta^2} \\ &\quad + \frac{\partial}{\partial f_p} \left(\sum_{p' \in N(p) \setminus p} \sqrt{\sum_{p'' \in N(p')} w_{p',p''} (f_{p'} - f_{p''})^2 + \beta^2} \right) \\ &= Z(p, p') \left(\sum_{p' \in N(p)} w_{p,p'} (f_p - f_{p'}) \right) + \sum_{p' \in N(p) \setminus p} Z(p', p'') \sum_{p'' \in N(p')} w_{p',p''} \frac{\partial}{\partial f_p} (f_{p'} - f_{p''})^2. \end{aligned}$$

Der Ausdruck $\frac{\partial}{\partial f_p} (f_{p'} - f_{p''})^2$ im letzten Term verschwindet für alle $p'' \neq p$, d. h.

$$\begin{aligned} \nabla_f ATV(f)|_p &= Z(p, p') \left(\sum_{p' \in N(p)} w_{p,p'} (f_p - f_{p'}) \right) + \sum_{p' \in N(p) \setminus p} Z(p', p'') w_{p',p} (f_{p'} - f_p) (-1) \\ &= \sum_{p' \in N(p)} w_{p',p} (Z(p, p') + Z(p', p'')) (f_p - f_{p'}), \end{aligned}$$

wobei wir wieder die Symmetrie der Gewichte $w_{p',p} = w_{p,p'}$ benutzt haben.

4.3 Anwendungen in der Bildapproximation

Ursprünglich wurde das Postprocessing-Verfahren mit der TV-Minimierung von Durand und Froment [DF03] für die Entstörung von Signalen vorgeschlagen. Wir setzen es hier erstmals zur Bildapproximation ein. Die Übertragung ist gerechtfertigt, weil auch die Approximation auf einem Wavelet-Shrinkage beruht, das zu Gibbs-Phänomenen im Hochpassbereich führen kann, die es dann im Postprocessing zu beheben gilt.

Dafür müssen wir die beschriebene ATV-Minimierung mit einer für Approximation geeigneten Wavelettransformation verbinden. Wir erinnern uns, dass die Tetrolets im Gegensatz zu den Curvelets [CD04, CDDY06] oder Contourlets [DV05] nicht-redundant und adaptiv sind, was sie besonders für dünne Bilddarstellungen prädestiniert. Die Verbindung von Tetrolet-Transformation und ATV-Minimierung wird sich als hervorragende Hybridmethode zur Approximation von Bildern erweisen.

Die Tetrolet-Transformation leidet nicht wesentlich unter hochfrequenten Gibbs-Phänomenen, weil sie auf Basisfunktionen mit kleinen Trägern beruht. Und doch treten Effekte auf, die durch die ATV-Minimierung behoben werden können. Zwei Probleme der Tetrolet-Transformation sind die Artefakte der 4×4 -Blockbildung und die fehlende Glattheit der Basisfunktionen. Letzteres macht sich besonders bei der Approximation von natürlichen Bildern mit höherer Regularität bemerkbar. Wir versuchen mit der vorgeschlagenen ATV-Minimierung diese Schwachstellen zu beheben, indem wir die wesentlichen Tetrolet-Koeffizienten fixieren und die betragsmäßig kleinen Koeffizienten im Iterationsprozess dahingehend ändern, dass die Bildregionen unter Kantenerhaltung geglättet werden.

Perfekte Rekonstruktion linearer Signale In diesem Paragraphen illustrieren wir, wie das vorgeschlagene Postprocessing-Verfahren die Regularität der Approximation erhöht. Unter gewissen Voraussetzungen können wir sogar perfekte Rekonstruktion des Inputsignals beweisen.

Die verwendete Tetrolet-Transformation ist eine verallgemeinerte, adaptive HWT. Das heißt, nach einem Shrinkage erhält man eine stückweise konstante Approximation des gegebenen Bildes. Beschränken wir uns nun auf den eindimensionalen Fall (und damit auf die klassische HWT), und betrachten wir ein lineares Inputsignal $f = (f[p])_{p=0}^{N-1}$ mit $f[p] = a p + b$ sowie seine Haarwavelet-Approximation \tilde{f} . Im Folgenden zeigen wir, dass wir in der Lage sind \tilde{f} durch die ATV-Minimierung im Postprocessing-Schritt so zu verbessern, dass wir sogar perfekte Rekonstruktion von f erreichen.

Satz 4.4. Sei $f = (f[p])_{p=0}^{N-1}$ ein lineares Signal mit $f[p] = a p + b$. Wenn wir den Algorithmus von Seite 132 auf f anwenden mit einem globalen Shrinkage-Parameter $2^{-1/2}a < \lambda < 2a$, so erhalten wir nach nur einem einzigen Postprocessing-Schritt mit $t_0 = a/2$ eine perfekte Rekonstruktion von f unter Verwendung der Nachbarschaft $N(p) = \{p - 1, p + 1\}$ und den

isotropischen Gewichten $w_{p,p'} = 1$, $p, p' \in \{0, \dots, N-1\}$.

Beweis. Wir beweisen die Eigenschaft der perfekten Rekonstruktion durch eine Anwendung des Algorithmus auf Seite 132.

Schritt 1. Wir starten im Raum V_0 mit der zum Signal f gehörenden kontinuierlichen Funktion

$$f^c = \sum_{k=0}^{N-1} c_{0,k}(f^c) \phi_{0,k}$$

wobei $c_{0,k}(f^c) = \langle f^c, \phi_{0,k} \rangle = f[k]$ ist. Die DWT mit Haarwavelets liefert im j -ten Zerlegungslevel, $j = -1, \dots, -J$, für $k = 0, \dots, 2^j N - 1$

$$\begin{aligned} c_{j,k}(f^c) &= \frac{1}{\sqrt{2}}(c_{j+1,2k+1}(f^c) + c_{j+1,2k}(f^c)), \\ d_{j,k}(f^c) &= \frac{1}{\sqrt{2}}(c_{j+1,2k+1}(f^c) - c_{j+1,2k}(f^c)). \end{aligned}$$

Insbesondere gilt im ersten Level für $k = 0, \dots, \frac{N}{2} - 1$

$$\begin{aligned} c_{-1,k}(f^c) &= \frac{1}{\sqrt{2}}(c_{0,2k+1}(f^c) + c_{0,2k}(f^c)) = \frac{1}{\sqrt{2}}(f[2k+1] + f[2k]) = \frac{1}{\sqrt{2}}(a(4k+1) + 2b), \\ d_{-1,k}(f^c) &= \frac{1}{\sqrt{2}}(c_{0,2k+1}(f^c) - c_{0,2k}(f^c)) = \frac{1}{\sqrt{2}}(f[2k+1] - f[2k]) = \frac{1}{\sqrt{2}}a. \end{aligned}$$

Schritt 2. Wenn wir den Schwellwert λ des Shrinkage-Operators so wählen, dass $2^{-1/2}a < \lambda < 2a$ gilt, so werden alle Waveletkoeffizienten $d_{-1,k}(f^c)$ im ersten Level abgeschnitten. Die übrigen Koeffizienten $c_{-1,k}(f^c)$ bleiben erhalten.

Schritt 3. Die anschließende Rekonstruktion liefert die Haarwavelet-Approximation \tilde{f}^c , deren Koeffizienten im Raum V_0 uns die diskrete Approximation $\tilde{f} = (\tilde{f}[p])_{p=0}^{N-1}$ des Eingangssignals f liefern wird.

Die Rekonstruktionsformeln für die Koeffizienten der HWT sind

$$\begin{aligned} c_{0,2k+1}(f^c) &= \frac{1}{\sqrt{2}}(c_{-1,k}(f^c) + d_{-1,k}(f^c)), \\ c_{0,2k}(f^c) &= \frac{1}{\sqrt{2}}(c_{-1,k}(f^c) - d_{-1,k}(f^c)), \quad k = 0, \dots, \frac{N}{2} - 1. \end{aligned}$$

Weil die Waveletkoeffizienten $d_{-1,k}(f^c)$ abgeschnitten wurden, vereinfachen sich diese Ausdrücke für $k = 0, \dots, \frac{N}{2} - 1$ zu

$$\tilde{c}_{0,2k+1}(f^c) = \tilde{c}_{0,2k}(f^c) = \frac{1}{\sqrt{2}}c_{-1,k}(f^c) = \frac{1}{2}(a(4k+1) + 2b),$$

was für $k = 0, \dots, N-1$ bedeutet

$$\tilde{c}_{0,k}(f^c) = \begin{cases} \frac{1}{2}(a(2k+1) + 2b), & k \text{ gerade} \\ \frac{1}{2}(a(2k-1) + 2b), & k \text{ ungerade} \end{cases} = \begin{cases} \frac{2k+1}{2}a + b, & k \text{ gerade} \\ \frac{2k-1}{2}a + b, & k \text{ ungerade} \end{cases}.$$

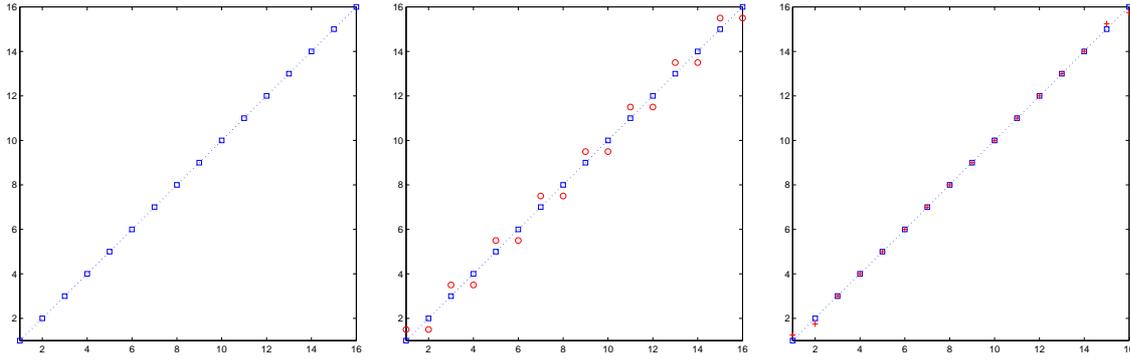


Abbildung 4.1: Approximation eines linearen Signals mit anschließendem Postprocessing. (a) Input, (b) Haarwavelet-Approximation, (c) geglättete Approximation nach einem einzigen TV-Minimierungs-Schritt.

Damit erhalten wir also

$$\tilde{f}^c = \sum_{k=0}^{N-1} \tilde{c}_{0,k}(f^c) \phi_{0,k}$$

woraus wir die diskrete Approximation $\tilde{f} = (\tilde{f}[p])_{p=0}^{N-1}$ von f erhalten durch

$$\tilde{f}[p] = \tilde{c}_{0,p}(f^c) = \begin{cases} \frac{2p+1}{2}a + b, & p \text{ gerade} \\ \frac{2p-1}{2}a + b, & p \text{ ungerade} \end{cases}, \quad (4.15)$$

siehe Abbildung 4.1(b).

Schritt 4. Setze $f^{(0)} := \tilde{f}$. Wir minimieren die TV-Seminorm von $f^{(0)}$ durch nur eine Iteration des Postprocessing-Schemas. Beachte, dass für $f^{(0)}$ gilt

$$\text{sgn}(f^{(0)}[p] - f^{(0)}[p+1]) = \begin{cases} 0, & p \text{ gerade} \\ -1, & p \text{ ungerade} \end{cases}. \quad (4.16)$$

(i) Den Subgradient $g_{ATV}(f^{(0)})$ an der Stelle p berechnen wir gemäß (4.11) als

$$g_{ATV}(f^{(0)})[p] = 2 \left(\sqrt{w_{p,p-1}} \text{sgn}(f^{(0)}[p] - f^{(0)}[p-1]) + \sqrt{w_{p,p+1}} \text{sgn}(f^{(0)}[p] - f^{(0)}[p+1]) \right).$$

Weil die Werte $f[p] = ap + b$ in $f = (f[p])_{p=0}^{N-1}$ Abtastwerte einer linearen Funktion sind, die keine Sprungstellen (Unstetigkeitsstellen) besitzt, ist es einleuchtend, isotropische Gewichte $w_{p,p+1} = 1$ für alle p zu wählen. Damit vereinfacht sich der Subgradient zu

$$g_{ATV}(f^{(0)})[p] = 2 \left(\text{sgn}(f^{(0)}[p] - f^{(0)}[p-1]) + \text{sgn}(f^{(0)}[p] - f^{(0)}[p+1]) \right).$$

(ii) Zerlegen wir den Subgradient in seine Haarwavelet-Koeffizienten, erhalten wir

$$\begin{aligned} d_{-1,k}(g_{ATV}(f^{(0)})) &= \frac{1}{\sqrt{2}}(g_{ATV}(f^{(0)})[2k+1] - g_{ATV}(f^{(0)})[2k]) \\ &= \sqrt{2} \text{sgn}(f^{(0)}[2k+1] - f^{(0)}[2k+2]) - \sqrt{2} \text{sgn}(f^{(0)}[2k] - f^{(0)}[2k+1]). \end{aligned} \quad (4.17)$$

(iii) Die inverse HWT ergibt

$$P_V(g_{ATV}(f^{(0)})) = \sum_{k=0}^{N/2-1} d_{-1,k}(g_{ATV}(f^{(0)})) \psi_{-1,k}$$

wegen $V = \{v \in X : d_{j,k}(v) = 0 \forall j < -1, c_{-1,k}(v) = 0 \forall k\}$. Eine punktweise Betrachtung für $p = 0, \dots, N-1$ zeigt uns

$$P_V(g_{ATV}(f^{(0)}))|_p = \begin{cases} -\frac{1}{\sqrt{2}} d_{-1, \frac{p}{2}}(g_{ATV}(f^{(0)})), & p \text{ gerade} \\ \frac{1}{\sqrt{2}} d_{-1, \frac{p-1}{2}}(g_{ATV}(f^{(0)})), & p \text{ ungerade} \end{cases}.$$

Mit (4.17) und der oben festgestellten Eigenschaft (4.16) von $f^{(0)}$ bekommen wir

$$\begin{aligned} P_V(g_{ATV}(f^{(0)}))|_p &= \begin{cases} -\operatorname{sgn}(f^{(0)}[p+1] - f^{(0)}[p+2]) + \operatorname{sgn}(f^{(0)}[p] - f^{(0)}[p+1]), & p \text{ gerade} \\ \operatorname{sgn}(f^{(0)}[p] - f^{(0)}[p+1]) - \operatorname{sgn}(f^{(0)}[p-1] - f^{(0)}[p]), & p \text{ ungerade} \end{cases} \\ &= \begin{cases} 1, & p \text{ gerade} \\ -1, & p \text{ ungerade} \end{cases}. \end{aligned}$$

(iv) Schließlich führen wir den Iterationsschritt durch

$$f^{(1)} = f^{(0)} - t_0 P_V(g_{ATV}(f^{(0)})),$$

und bekommen an der Stelle $p = 0, \dots, N-1$ mit (4.15) und der Schrittweite $t_0 = a/2$

$$f^{(1)}[p] = f^{(0)}[p] - t^{(0)} \begin{cases} 1, & p \text{ gerade} \\ -1, & p \text{ ungerade} \end{cases} = \begin{cases} \frac{2p+1}{2}a + b - a/2, & p \text{ gerade} \\ \frac{2p-1}{2}a + b + a/2, & p \text{ ungerade} \end{cases} = ap + b.$$

Damit ist das Ausgangssignal perfekt rekonstruiert. \square

Numerische Resultate Abbildung 4.1 veranschaulicht die Aussage von Satz 4.4. Die Eigenschaft der perfekten Rekonstruktion wird für ein lineares Signal $f = (f[p])_{p=0}^{15}$ mit $f[p] = p$ illustriert. Die Haarwavelet-Approximation \tilde{f} in 4.1(b) kann geglättet werden, indem man einen einzigen TV-Minimierungs-Schritt des vorgeschlagenen Postprocessing-Schemas durchführt. Die daraus resultierende Approximation $f^{(1)}$ in 4.1(c) ist (außer an den Rändern) eine perfekte Rekonstruktion.

Wir wenden die vorgestellte Hybridmethode nun auf die beiden Testbilder in Abbildung 4.2 an. Die Abbildungen 4.3 und 4.4 verdeutlichen den Glättungseffekt der ATV-Minimierung, die auf die Tetrolet-Approximation angewandt wird, und zum Vergleich auch auf die Wavelettransformation, die auf den biorthogonalen 9-7 Wavelets basiert. Für beide Transformationen stellen wir eine Verbesserung durch die ATV-Minimierung fest. Bei der Tetrolet-Approximation des „clock“-Bildes in Abbildung 4.3 steigt der PSNR-Wert nur leicht nach 5 Iterationen, aber die visuelle Qualität des Bildes ist beträchtlich verbessert worden, wie man z. B. am Schatten des Buches sieht. Auch die im Hintergrund sichtbaren Blocking-Artefakte der Tetrolet-Transformation sind behoben worden.

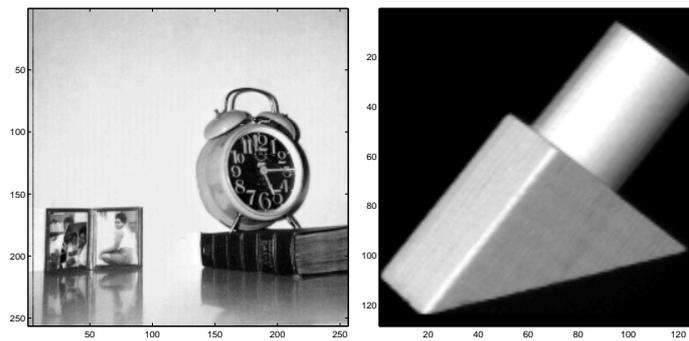


Abbildung 4.2: Testbilder.

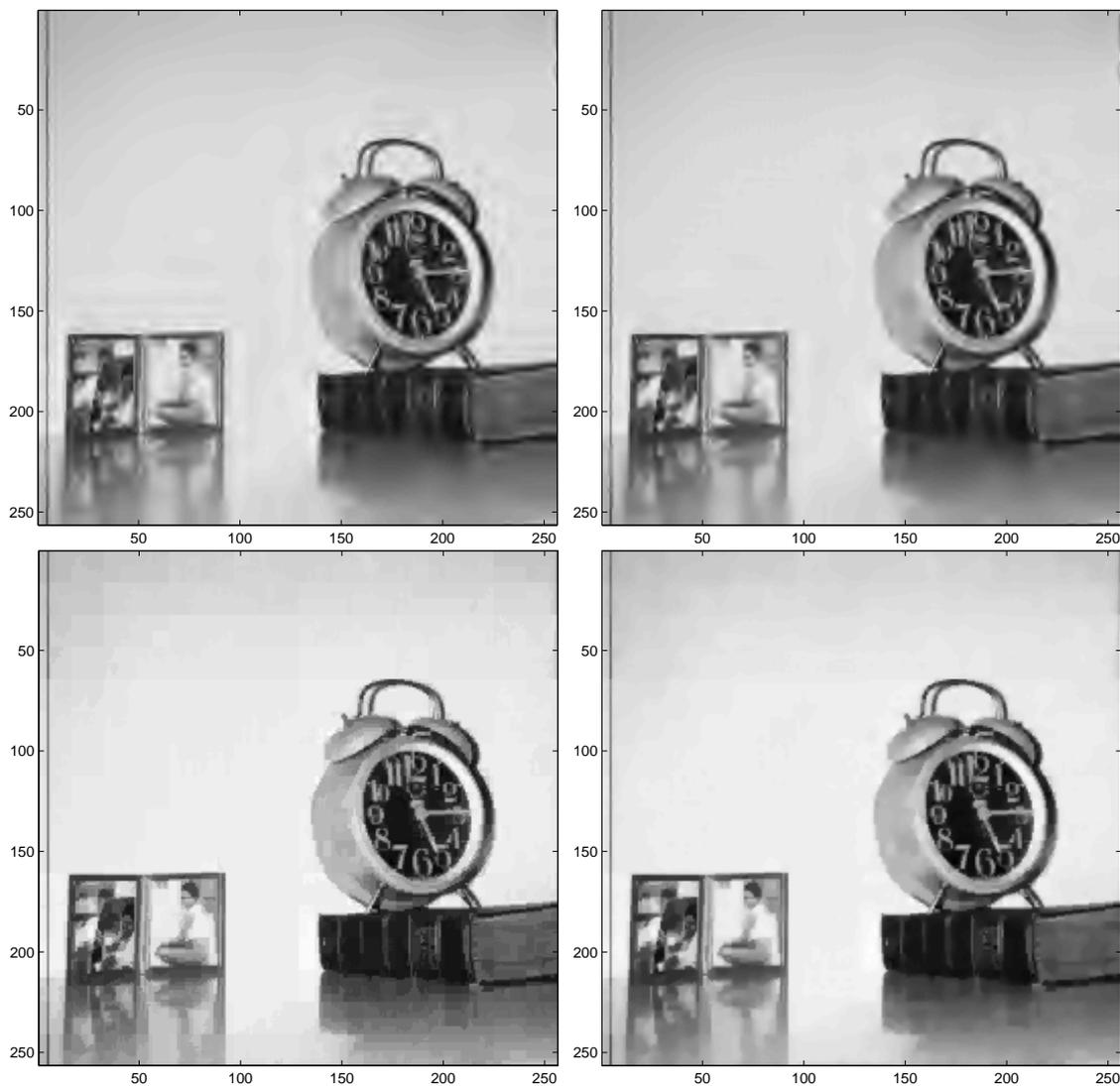


Abbildung 4.3: Approximation des „clock“-Bildes mit 2048 Koeffizienten. (a) Biorthogonal 9-7 Shrinkage, PSNR 30.07 dB. (b) Biorthogonal 9-7 Shrinkage mit ATV-Minimierung, PSNR 30.62 dB. (c) Tetrolet-Shrinkage, PSNR 32.61 dB. (d) Tetrolet-Shrinkage mit ATV-Minimierung, PSNR 33.08 dB.

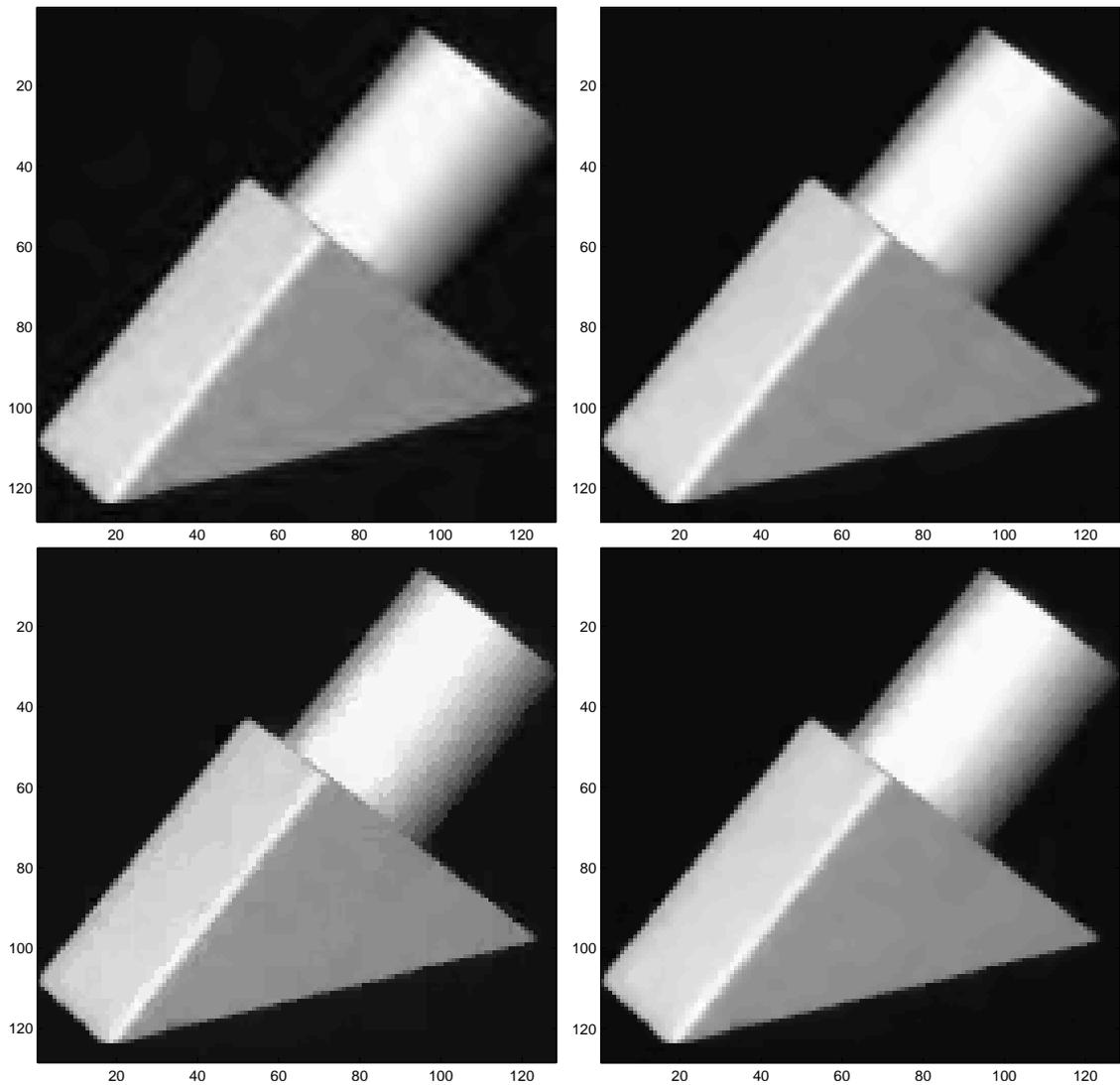


Abbildung 4.4: Approximation eines stückweise glatten Bildes mit 1024 Koeffizienten. (a) Biorthogonal 9-7 Shrinkage, PSNR 36.07 dB. (b) Biorthogonal 9-7 Shrinkage mit ATV-Minimierung, PSNR 38.24 dB. (c) Tetrolet-Shrinkage, PSNR 36.59 dB. (d) Tetrolet-Shrinkage mit ATV-Minimierung, PSNR 38.06 dB.

Für das stückweise glatte Bild in Abbildung 4.4 bemerken wir nach 10 Iterationen einen starken Glättungseffekt, der die stückweise konstante Tetrolet-Approximation deutlich verbessert. Hier sind die Ergebnisse für die zugrunde liegende Tetrolet-Transformation in der gleichen Größenordnung wie bei der Verwendung des 9-7 Filters. Ein Grund dafür liegt darin, dass die Tetrolets schon an sich eine Richtungseigenschaft besitzen, während die Tensorproduktwavelets im 9-7 Filter keinerlei Anisotropie aufweisen. Deshalb ist der qualitative Gewinn der anisotropen Postprocessing-Technik bei der Tensorprodukt-Wavelettransformation größer als bei der adaptiven Tetrolet-Transformation. In Zahlen: Bei der Hybridmethode mit 9-7 Filter steigt der PSNR-Wert von 36.07 dB auf 38.24 dB, während er bei der Tetrolet-Transformation nur von 36.59 dB auf 38.06 dB anwächst.

In beiden Abbildungen haben wir als Schrittweite die Folge $t_k = \frac{1}{k+1}$ gewählt (gemäß (4.8)) sowie die Gewichtsparameter $\sigma_s = 2$ und $\sigma_i = 100$. Außerdem haben wir das regulierte, diskrete ATV-Funktional aus (4.14) mit einer 8er-Nachbarschaft bevorzugt.

Anhang A

Konstruktion eines dualen Frames

In diesem Abschnitt beweisen wir Satz 2.4, der die Konstruktion eines dualen Frames in endlich-dimensionalen Unterräumen des $L^2(\Omega)$ beschreibt.

Sei $F := (f_i)_{i=0}^{N-1}$, $N \in \mathbb{N}$, ein Vektor mit Funktionen $f_i \in L^2(\Omega)$, $\Omega \subset \mathbb{R}^d$, und

$$V := \text{clos}_{L^2(\Omega)} \text{span} \{f_0, \dots, f_{N-1}\}$$

der durch F aufgespannte Unterraum von $L^2(\Omega)$. Wir suchen einen dualen Frame $\tilde{F} := (\tilde{f}_k)_{k=0}^{N-1}$, $\tilde{f}_k \in V$, derart, dass für alle $g \in V$ die Beziehung

$$g = \sum_{k=0}^{N-1} \langle g, \tilde{f}_k \rangle f_k \tag{A.1}$$

gilt. Dazu starten wir mit dem Ansatz $\tilde{F} = A^0 F$, wobei die Koeffizientenmatrix $A^0 \in \mathbb{R}^{N \times N}$ in geeigneter Weise berechnet werden muss. Mit

$$H := \langle \tilde{F}, F \rangle = \left(\int_{\mathbb{R}^d} \tilde{f}_k(x) \overline{f_l(x)} dx \right)_{k,l=0}^{N-1}$$

findet man

$$H = \langle A^0 F, F \rangle = A^0 \langle F, F \rangle = A^0 G, \tag{A.2}$$

hierbei ist $G = \langle F, F \rangle$ die Gram'sche Matrix von F . Wenn F keine Basis von V ist, sondern lediglich ein Frame, ist G nicht invertierbar. Sei in diesem Fall also $\text{Rang}(G) = m \leq N$. In drei Schritten konstruieren wir einen Funktionenvektor \tilde{F} , der die gesuchten dualen Framefunktionen enthält.

1. Wir betrachten zunächst den endlich-dimensionalen Raum $W := \text{span} \{F(x) : x \in \mathbb{R}^d\} \subseteq \mathbb{R}^N$, und das dazugehörige orthogonale Komplement $W^\perp \subset \mathbb{R}^N$ in \mathbb{R}^N . Wegen $\text{Rang}(G) = m$ folgt $\dim W = m$. Weiter sei $\{e_0, \dots, e_{N-m-1}\}$ eine ONB von W^\perp , d. h. $e_l^T F(x) = 0$ für alle $x \in \mathbb{R}^d$, $l = 0, \dots, N - m - 1$.

Betrachten wir jetzt $G e_l = \langle F, F \rangle e_l = 0$ für $l = 0, \dots, N - m - 1$, so folgt $H e_l = A^0 G e_l = 0$, $l = 0, \dots, N - m - 1$. Sei nun U eine unitäre Matrix von Eigenvektoren von G , deren letzten $N - m$ Spalten mit e_0, \dots, e_{N-m-1} übereinstimmen.

Die verallgemeinerte Moore-Penrose-Inverse von G ist gegeben durch $G^\dagger = U^* D' U$, wobei $G = U^* D U$, und D' ist eine Diagonalmatrix, die die reziproken Werte der Nicht-Nullelemente von D in der gleichen Reihenfolge enthält. Das heißt, wegen $G = G G^\dagger G$ folgt

$$H = A^0 G = A^0 G G^\dagger G = H G^\dagger G,$$

so dass die Matrix $A^0 = H G^\dagger$ die Bedingung (A.2) erfüllt.

2. Um A^0 zu berechnen, genügt es also, die Matrix H zu konstruieren. Aus (A.1) wissen wir, dass jede Funktion $g \in V$ die Form $g = c^T F$ besitzt, mit $c := (\langle g, \tilde{f}_k \rangle)_{k=0}^{N-1}$. Wir erhalten dann

$$c^T F = \left((\langle g, \tilde{f}_k \rangle)_{k=0}^{N-1} \right)^T F = \left((\langle c^T F, \tilde{f}_k \rangle)_{k=0}^{N-1} \right)^T F = c^T \langle F, \tilde{F} \rangle F = c^T H^T F$$

und damit $c^T (I - H^T) F = 0$ für alle $c \in \mathbb{R}^N$. Wie im ersten Schritt gesehen, folgt dass $(I - H)c \in W^\perp$ für alle $c \in \mathbb{R}^N$ gilt. Insbesondere ist jede Spalte von $(I - H)$ in W^\perp , d. h. jede Spalte ist eine Linearkombination von e_0, \dots, e_{N-m-1} . Also gibt es Vektoren $b_0, \dots, b_{N-m-1} \in \mathbb{R}^N$ mit $I - H = \sum_{l=0}^{N-m-1} e_l b_l^T$, das bedeutet

$$H = I - \sum_{l=0}^{N-m-1} e_l b_l^T.$$

3. Wir setzen jetzt voraus, dass der duale Frame \tilde{F} die gleichen Beziehungen erfüllt wie F , dass also $e_i^T \tilde{F}(x) = 0$ für alle $x \in \mathbb{R}^d$, $i = 0, \dots, N - m - 1$, gilt. Diese Beziehungen führen dann zu

$$0^T = e_i^T H = e_i^T - \sum_{l=0}^{N-m-1} e_i^T e_l b_l^T = e_i^T - b_i^T,$$

so dass wir die Vektoren $b_i = e_i$, $i = 0, \dots, N - m - 1$, erhalten. Die Matrix H ist dann durch

$$H = I - \sum_{l=0}^{N-m-1} e_l e_l^T.$$

gegeben. Zusammen mit den obigen Bemerkungen führt das zu dem dualen Frame

$$\tilde{F} = A^0 F = H G^\dagger F = \left(I - \sum_{l=0}^{N-m-1} e_l e_l^T \right) G^\dagger F = G^\dagger F. \quad (\text{A.3})$$

Die letzte Gleichung folgt aus der Beobachtung $e_l^T G^\dagger = e_l^T U^* D' U = 0$ für $l = 0, \dots, N - m - 1$, weil die ersten m Spalten von U^* gemäß unserer Konstruktion in W sind und die letzten $N - m - 1$ Diagonaleinträge von D' Null sind.

Bemerkung A.1. Beachte, dass das obige Konstruktionsverfahren uns auch die Gram'sche Kreuz-Matrix

$$H = \langle \tilde{F}, F \rangle = (\langle \tilde{f}_i, f_j \rangle)_{i,j=0}^{N-1}$$

liefert.

Wenden wir das Verfahren jetzt auf unseren Approximationsraum V_0 an (siehe Satz 2.3), der durch den Frame $\{\phi_i(\cdot - k) : i = 0, \dots, 15; k \in \mathbb{Z}^2\}$ erzeugt wird, so betrachten wir zu diesem Zweck die translatierten Framefunktionen, die einen kompakten Träger in $[0, 1]^2$ haben. Weil es keine Träger-Überlappungen gibt, deren Maß größer als 0 in \mathbb{R}^2 ist, können wir uns auf den Funktionenvektor Φ in (2.10) beschränken. Die Abhängigkeiten (2.16) unter den Framefunktionen führen zu den Vektoren $\hat{e}_i \in \mathbb{R}^{16}$, so dass $\hat{e}_i^T \Phi = 0$ für $i = 0, \dots, 4$, mit

$$\hat{e}_0 = (1, 1, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 0, 0, 0, 0)^T,$$

$$\hat{e}_1 = (0, 0, 1, 1, 0, 0, 0, 0, -1, -1, 0, 0, 0, 0, 0, 0)^T,$$

$$\hat{e}_2 = (0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1)^T,$$

$$\hat{e}_3 = (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, -1, -1, 0, 0)^T,$$

$$\hat{e}_4 = (1, 1, 1, 1, -1, -1, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0)^T.$$

Die orthonormalisierten Vektoren sind dann $e_i = \frac{1}{2}\hat{e}_i$ für $i = 0, \dots, 3$ und $e_4 = \frac{1}{4}(1, 1, 1, 1, -1, -1, -1, -1, 1, 1, 1, 1, -1, -1, -1, -1)^T$. Also ist die Matrix $H = \langle \tilde{\Phi}, \Phi \rangle$ gegeben durch

$$H = I - \sum_{l=0}^4 e_l e_l^T.$$

Anhang B

Herleitungen zu Kapitel 2.5

B.1 Skalarprodukte der Splinefunktionen ϕ_1 und ϕ_2

Wir berechnen die Skalarprodukte für die in Abschnitt 2.5.1 konstruierten Splinefunktionen ϕ_1 und ϕ_2 . Diese Ergebnisse benötigen wir insbesondere für die Berechnung der Spline-Wavelets, die wir in B.3 vornehmen.

Zunächst betrachten wir die Skalarprodukte der Basisfunktionen ϕ_1 und ϕ_2 aus V_0 .

$$\begin{aligned}
 (a) \quad & \langle \phi_1, \phi_1 \rangle = \|\phi_1\|_2^2 = \frac{1}{6}, \\
 (b) \quad & \langle \phi_2, \phi_2 \rangle = \|\phi_2\|_2^2 = \frac{1}{3}, \\
 (c) \quad & \langle \phi_2, \phi_2(2 \cdot -k) \rangle = \frac{1}{24} \quad \text{für } k = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \\
 (d) \quad & \langle \phi_2, \phi_1(2 \cdot -k) \rangle = \frac{1}{24} \quad \text{für } k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}.
 \end{aligned}$$

Für alle anderen $k \in \mathbb{Z}^2$ sind die Skalarprodukte gleich Null.

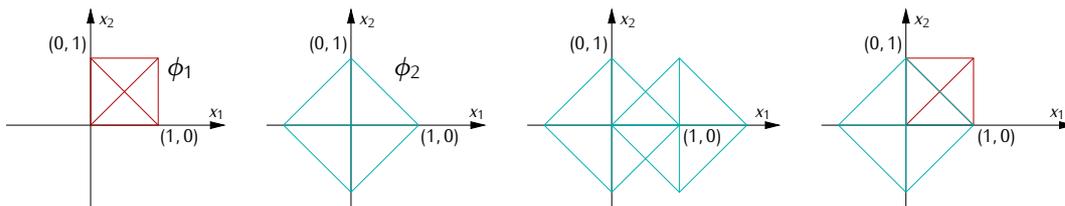


Abbildung B.1: Skalarprodukte von ϕ_1 und ϕ_2 aus V_0 .

Jetzt betrachten wir die Skalarprodukte der Basisfunktionen ϕ_1 und ϕ_2 aus V_0 sowie der Basisfunktionen $\phi_1(2 \cdot)$ und $\phi_2(2 \cdot)$ aus V_1 .

$$\begin{aligned}
 (a) \quad & \langle \phi_1, \phi_1(2 \cdot -k) \rangle = \frac{3}{96} \quad \text{für } k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\
 (b) \quad & \langle \phi_1, \phi_2(2 \cdot -k) \rangle = \frac{1}{192} \quad \text{für } k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \\
 (c) \quad & \langle \phi_1, \phi_2(2 \cdot -k) \rangle = \frac{1}{48} \quad \text{für } k = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \\
 (d) \quad & \langle \phi_1, \phi_2(2 \cdot -k) \rangle = \frac{5}{48} \quad \text{für } k = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\
 (e) \quad & \langle \phi_2, \phi_1(2 \cdot -k) \rangle = \frac{1}{24} \quad \text{für } k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix},
 \end{aligned}$$

$$\begin{aligned}
 (f) \quad & \langle \phi_2, \phi_1(2 \cdot -k) \rangle = \frac{1}{192} \quad \text{für } k = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \begin{pmatrix} -1 \\ -2 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} -2 \\ -1 \end{pmatrix} \\
 (g) \quad & \langle \phi_2, \phi_2(2 \cdot -k) \rangle = \frac{1}{8} \quad \text{für } k = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \\
 (h) \quad & \langle \phi_2, \phi_2(2 \cdot -k) \rangle = \frac{1}{16} \quad \text{für } k = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \\
 (i) \quad & \langle \phi_2, \phi_2(2 \cdot -k) \rangle = \frac{1}{192} \quad \text{für } k = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \\
 (j) \quad & \langle \phi_2, \phi_2(2 \cdot -k) \rangle = \frac{1}{64} \quad \text{für } k = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}.
 \end{aligned}$$

Für alle anderen $k \in \mathbb{Z}^2$ sind die Skalarprodukte gleich Null.

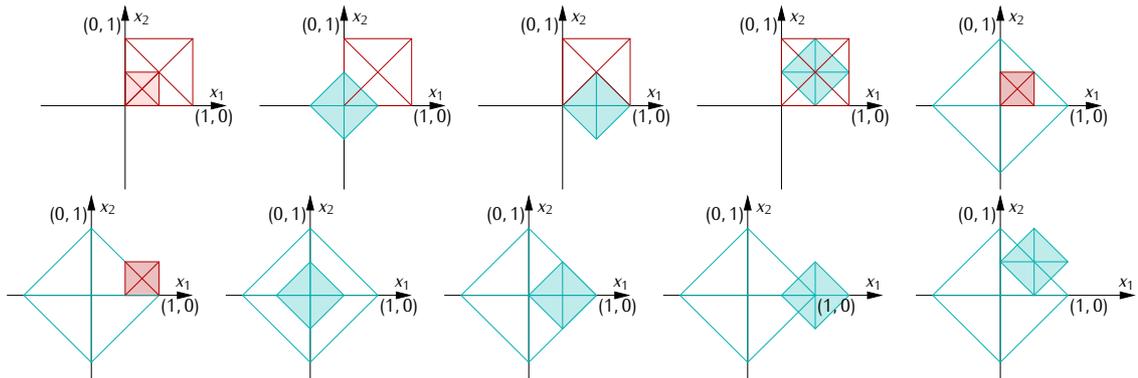


Abbildung B.2: Skalarprodukte von ϕ_1, ϕ_2 aus V_0 und $\phi_1(2 \cdot), \phi_2(2 \cdot)$ aus V_1 .

B.2 Konstruktion der dualen Splinefunktionen $\tilde{\phi}_1$ und $\tilde{\phi}_2$ im Abschnitt 2.5.1

Wir leiten die gesuchten dualen Funktionen $\tilde{\phi}_1$ in (2.53) und $\tilde{\phi}_2$ in (2.54) her. Wegen der Trägerbedingung (2.51) erhält man aus dem allgemeinen Ansatz (2.52) der gesuchten Funktionen die beiden Gleichungen

$$\begin{aligned}
 \tilde{\phi}_1 &= a^1_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} \phi_1(2 \cdot) + a^1_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + a^1_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + a^1_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \\
 &\quad + b^1_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}} \phi_2(2 \cdot - \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \\
 \tilde{\phi}_2 &= a^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} \phi_1(2 \cdot) + a^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) + a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) + a^2_{\begin{pmatrix} -1 \\ -1 \end{pmatrix}} \phi_1(2 \cdot - \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \\
 &\quad + b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} \phi_2(2 \cdot) + b^2_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}} \phi_2(2 \cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) + b^2_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}} \phi_2(2 \cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) + b^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}} \phi_2(2 \cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) \\
 &\quad + b^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} \phi_2(2 \cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}).
 \end{aligned}$$

Zuerst wenden wir uns der Funktion $\tilde{\phi}_1$ zu. Aus der Dualitätsbedingung (2.49) bekommen wir jetzt folgende Gleichungen

$$\begin{aligned}
 1. \quad & \langle \tilde{\phi}_1, \phi_1 \rangle = \|\phi_1\|_2^2 = \frac{1}{6} \\
 & \Rightarrow \frac{1}{32} a^1_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{1}{32} a^1_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}} + \frac{1}{32} a^1_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}} + \frac{1}{32} a^1_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}} + \frac{5}{48} b^1_{\begin{pmatrix} 1 \\ 1 \end{pmatrix}} = \frac{1}{6}
 \end{aligned}$$

$$2. \langle \tilde{\phi}_1, \phi_2 \rangle = 0$$

$$\Rightarrow \frac{1}{24}a^1_{\binom{1}{0}} + \frac{1}{192}a^1_{\binom{1}{1}} + \frac{1}{192}a^1_{\binom{0}{1}} + \frac{3}{192}b^1_{\binom{1}{1}} = 0$$

$$3. \langle \tilde{\phi}_1, \phi_2(\cdot - \binom{1}{0}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^1_{\binom{1}{0}} + \frac{1}{24}a^1_{\binom{1}{1}} + \frac{1}{192}a^1_{\binom{0}{1}} + \frac{3}{192}b^1_{\binom{1}{1}} = 0$$

$$4. \langle \tilde{\phi}_1, \phi_2(\cdot - \binom{0}{1}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^1_{\binom{1}{0}} + \frac{1}{24}a^1_{\binom{0}{1}} + \frac{1}{192}a^1_{\binom{1}{1}} + \frac{3}{192}b^1_{\binom{1}{1}} = 0$$

$$5. \langle \tilde{\phi}_1, \phi_2(\cdot - \binom{1}{1}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^1_{\binom{1}{1}} + \frac{1}{192}a^1_{\binom{0}{1}} + \frac{1}{24}a^1_{\binom{1}{1}} + \frac{3}{192}b^1_{\binom{1}{1}} = 0$$

Fasst man diese fünf Gleichungen in Matrix-Vektor-Schreibweise zusammen, so erhalten wir das lineare Gleichungssystem

$$\frac{1}{8} \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{5}{6} \\ \frac{1}{3} & \frac{1}{24} & \frac{1}{24} & 0 & \frac{3}{24} \\ \frac{1}{24} & \frac{1}{3} & 0 & \frac{1}{24} & \frac{3}{24} \\ \frac{1}{24} & 0 & \frac{1}{3} & \frac{1}{24} & \frac{3}{24} \\ 0 & \frac{1}{24} & \frac{1}{24} & \frac{1}{3} & \frac{3}{24} \end{pmatrix} \mathbf{a}^1 = \begin{pmatrix} \frac{1}{6} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

dessen Lösungsvektor $\mathbf{a}^1 := [a^1_{\binom{1}{0}}, a^1_{\binom{1}{1}}, a^1_{\binom{0}{1}}, a^1_{\binom{1}{1}}, b^1_{\binom{1}{1}}]^T = [-\frac{3}{4}, -\frac{3}{4}, -\frac{3}{4}, -\frac{3}{4}, \frac{5}{2}]^T$ ist.

Jetzt betrachten wir in analoger Weise die duale Funktion $\tilde{\phi}_2$.

$$1. \langle \tilde{\phi}_2, \phi_2 \rangle = \|\phi_2\|_2^2 = \frac{1}{3}$$

$$\Rightarrow \frac{1}{24} \left(a^2_{\binom{0}{0}} + a^2_{\binom{0}{-1}} + a^2_{\binom{0}{-1}} + a^2_{\binom{-1}{-1}} \right) + \frac{3}{24} b^2_{\binom{0}{0}} + \frac{3}{48} \left(b^2_{\binom{1}{0}} + b^2_{\binom{0}{1}} + b^2_{\binom{-1}{0}} + b^2_{\binom{0}{-1}} \right) = \frac{1}{3}$$

$$2. \langle \tilde{\phi}_2, \phi_1 \rangle = 0$$

$$\Rightarrow \frac{3}{96}a^2_{\binom{0}{0}} + \frac{1}{192}b^2_{\binom{0}{0}} + \frac{1}{48}b^2_{\binom{1}{0}} + \frac{1}{48}b^2_{\binom{0}{1}} = 0$$

$$3. \langle \tilde{\phi}_2, \phi_1(\cdot - \binom{-1}{0}) \rangle = 0$$

$$\Rightarrow \frac{3}{96}a^2_{\binom{-1}{0}} + \frac{1}{192}b^2_{\binom{0}{0}} + \frac{1}{48}b^2_{\binom{-1}{0}} + \frac{1}{48}b^2_{\binom{0}{1}} = 0$$

$$4. \langle \tilde{\phi}_2, \phi_1(\cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{3}{96}a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{1}{48}b^2_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}} + \frac{1}{48}b^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} = 0$$

$$5. \langle \tilde{\phi}_2, \phi_1(\cdot - \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{3}{96}a^2_{\begin{pmatrix} -1 \\ -1 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{1}{48}b^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}} + \frac{1}{48}b^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} = 0$$

$$6. \langle \tilde{\phi}_2, \phi_2(\cdot - \begin{pmatrix} 1 \\ 0 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{1}{192}a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{3}{48}b^2_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}} = 0$$

$$7. \langle \tilde{\phi}_2, \phi_1(\cdot - \begin{pmatrix} 0 \\ 1 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{1}{192}a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{3}{48}b^2_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}} = 0$$

$$8. \langle \tilde{\phi}_2, \phi_1(\cdot - \begin{pmatrix} -1 \\ 0 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^2_{\begin{pmatrix} -1 \\ -1 \end{pmatrix}} + \frac{1}{192}a^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{3}{48}b^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}} = 0$$

$$9. \langle \tilde{\phi}_2, \phi_1(\cdot - \begin{pmatrix} 0 \\ -1 \end{pmatrix}) \rangle = 0$$

$$\Rightarrow \frac{1}{192}a^2_{\begin{pmatrix} -1 \\ -1 \end{pmatrix}} + \frac{1}{192}a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} + \frac{1}{192}b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}} + \frac{3}{48}b^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}} = 0$$

Fasst man diese neun Gleichungen in Matrix-Vektor-Schreibweise zusammen, so erhalten wir das lineare Gleichungssystem

$$\frac{1}{8} \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{4} & 0 & 0 & 0 & \frac{1}{24} & \frac{1}{6} & \frac{1}{6} & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & \frac{1}{24} & 0 & \frac{1}{6} & \frac{1}{6} & 0 \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{24} & 0 & \frac{1}{6} & 0 & \frac{1}{6} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{24} & 0 & 0 & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{24} & 0 & \frac{1}{24} & 0 & \frac{1}{24} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{24} & \frac{1}{24} & 0 & 0 & \frac{1}{24} & 0 & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{24} & 0 & \frac{1}{24} & \frac{1}{24} & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{24} & \frac{1}{24} & \frac{1}{24} & 0 & 0 & 0 & \frac{1}{2} \end{pmatrix} \mathbf{a}^2 = \begin{pmatrix} \frac{1}{3} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

dessen Lösungsvektor

$$\begin{aligned} \mathbf{a}^2 &:= [a^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}, a^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}}, a^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}}, a^2_{\begin{pmatrix} -1 \\ -1 \end{pmatrix}}, b^2_{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}, b^2_{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}, b^2_{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}, b^2_{\begin{pmatrix} -1 \\ 0 \end{pmatrix}}, b^2_{\begin{pmatrix} 0 \\ -1 \end{pmatrix}}]^T \\ &= [-\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}, \frac{7}{2}, -\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}]^T \end{aligned}$$

ist.

B.3 Konstruktion der Wavelet-Splinefunktionen im Abschnitt 2.5.2

Aus der Definition der Wavelets in (2.59),

$$\psi_i(\cdot - k) := \phi_i(2 \cdot - k) - \sum_{\substack{i'=1,2 \\ k' \in \mathbb{Z}^2}} \frac{\langle \phi_i(2 \cdot - k), \tilde{\phi}_{i'}(\cdot - k') \rangle}{\|\phi_{i'}\|_2^2} \phi_{i'}(\cdot - k'),$$

lässt sich eine vereinfachte Darstellung herleiten, indem wir überlegen, wann das Skalarprodukt

$$\langle \phi_i(2 \cdot - k), \tilde{\phi}_{i'}(\cdot - k') \rangle = \langle \phi_i(2 \cdot - (k - 2k')), \tilde{\phi}_{i'} \rangle$$

tatsächlich von Null verschieden ist. Mit Hilfe der Definition der dualen Splinefunktionen (2.55) und (2.56) und den Skalarprodukten in B.1 erhalten wir schließlich

$$\langle \phi_1(2 \cdot - (k - 2k')), \tilde{\phi}_1 \rangle = \begin{cases} -\frac{1}{192}, & k - 2k' = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\ 0, & \text{sonst} \end{cases}$$

$$\langle \phi_1(2 \cdot - (k - 2k')), \tilde{\phi}_2 \rangle = \begin{cases} \frac{1}{48}, & k - 2k' = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \\ -\frac{1}{384}, & k - 2k' = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \begin{pmatrix} -1 \\ -2 \end{pmatrix}, \begin{pmatrix} -2 \\ -1 \end{pmatrix}, \\ 0, & \text{sonst} \end{cases}$$

$$\langle \phi_2(2 \cdot - (k - 2k')), \tilde{\phi}_1 \rangle = \begin{cases} -\frac{1}{128}, & k - 2k' = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \\ \frac{1}{96}, & k - 2k' = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \\ \frac{17}{96}, & k - 2k' = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\ 0, & \text{sonst} \end{cases}$$

$$\langle \phi_2(2 \cdot - (k - 2k')), \tilde{\phi}_2 \rangle = \begin{cases} \frac{13}{48}, & k - 2k' = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \\ \frac{1}{96}, & k - 2k' = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \\ -\frac{1}{128}, & k - 2k' = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \\ -\frac{1}{384}, & k - 2k' = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \\ 0, & \text{sonst.} \end{cases}$$

Jetzt fragen wir uns, für welche $k' \in \mathbb{Z}^2$ bei festem $k \in \mathbb{Z}^2$ die Skalarprodukte einen von Null verschiedenen Wert liefern. Betrachten wir das erste Skalarprodukt, so muss k' die Gestalt $k' = \frac{k-m}{2}$ annehmen mit $m = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Das führt uns wieder zu der Polyphasen-Zerlegung der Ebene \mathbb{Z}^2 . Denn wir erhalten die folgenden vier Fälle: Ist $k \in 2\mathbb{Z}^2$, so gilt $k' = \frac{k}{2}$, für $k \in 2\mathbb{Z}^2 + \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ gilt $k' = \frac{k - \begin{pmatrix} 1 \\ 0 \end{pmatrix}}{2}$, für $k \in 2\mathbb{Z}^2 + \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ gilt $k' = \frac{k - \begin{pmatrix} 0 \\ 1 \end{pmatrix}}{2}$ und für $k \in 2\mathbb{Z}^2 + \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ gilt $k' = \frac{k - \begin{pmatrix} 1 \\ 1 \end{pmatrix}}{2}$. Analog können wir auch für die anderen drei Skalarprodukte diese Fallunterscheidung vornehmen. Damit erhalten wir schließlich die vier Wavelet-Darstellungen auf Seite 88.

Literaturverzeichnis

- [ACD⁺08] ARÁNDIGA, F., A. COHEN, R. DONAT, N. DYN und B. MATEI: *Approximation of piecewise smooth functions and images by edge-adapted (ENO-EA) nonlinear multi-resolution techniques*. Appl. Comput. Harmon. Anal., 24(2):225–250, 2008.
- [BA83] BURT, P. J. und E. H. ADELSON: *The Laplacian pyramid as a compact image code*. IEEE Trans. Comm., 31(4):532–540, 1983.
- [BBD⁺06] BERKELS, B., M. BURGER, M. DROSKE, O. NEMITZ und M. RUMPF: *Cartoon extraction based on anisotropic image classification*. In: *Vision, Modeling, and Visualization Proceedings*, Seiten 293–300, 2006.
- [BDG03] BUHMANN, M. D., O. DAVYDOV und T. N. T. GOODMAN: *Cubic spline prewavelets on the four-directional mesh*. Found. Comput. Math., 3(2):113–133, 2003.
- [BHF98] BÖLCSKEI, H., F. HLAWATSCH und H. G. FEICHTINGER: *Frame-theoretic analysis of oversampled filter banks*. IEEE Trans. Signal Process., 46(12):3256–3268, 1998.
- [BNB00] BACHMANN, G., L. NARICI und E. BECKENSTEIN: *Fourier and Wavelet Analysis*. Springer, New York, 2000.
- [Bow97] BOWNIK, M.: *Tight frames of multidimensional wavelets*. J. Fourier Anal. Appl., 3(5):525–542, 1997.
- [BT01] BENEDETTO, J. und O. TREIBER: *Wavelet frames: Multiresolution analysis and extension principles*. In: DEBNATH, L. (Herausgeber): *Wavelet Transforms and Time-Frequency Signal Analysis*, Seiten 3–36. Birkhäuser, Boston, 2001.
- [CD99] CANDÈS, E. J. und D. L. DONOHO: *Ridgelets: a key to higher-dimensional intermitency?* Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 357(1760):2495–2509, 1999.
- [CD00] CANDÈS, E. J. und D. L. DONOHO: *Curvelets – A surprisingly effective nonadaptive representation for objects with edges*. In: COHEN, A., C. RABUT und L. L. SCHUMAKER (Herausgeber): *Curve and Surface Fitting: Saint-Malo 1999*, Seiten 105–120, Nashville, 2000.

- [CD04] CANDÈS, E. J. und D. L. DONOHO: *New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities*. *Comm. Pure Appl. Math.*, 57(2):219–266, 2004.
- [CDDY06] CANDÈS, E. J., L. DEMANET, D. L. DONOHO und L. YING: *Fast discrete curvelet transforms*. *Multiscale Model. Simul.*, 5(3):861–899, 2006.
- [CEPY06] CHAN, T., S. ESEDOGLU, F. PARK und A. YIP: *Total variation image restoration: Overview and recent developments*. In: PARAGIOS, N., Y. CHEN und O. FAUGERAS (Herausgeber): *Handbook of Mathematical Models of Computer Vision*, Seiten 17–32. Springer, New York, 2006.
- [CHM04] CABRELLI, C. A., C. HEIL und U. M. MOLTER: *Self-similarity and multiwavelets in higher dimension*. *Mem. Amer. Math. Soc.*, 170(802), 2004.
- [Chr03] CHRISTENSEN, O.: *An Introduction to Frames and Riesz Bases*. Birkhäuser, Boston, 2003.
- [CHS02] CHUI, C. K., W. HE und J. STÖCKLER: *Compactly supported tight and sibling frames with maximum vanishing moments*. *Appl. Comput. Harmon. Anal.*, 13(3):224–262, 2002.
- [Chu88] CHUI, C. K.: *Multivariate Splines*. SIAM, Philadelphia, 1988.
- [CJS94] CHUI, C. K., K. JETTER und J. STÖCKLER: *Wavelets and frames on the four-directional mesh*. In: CHUI, C. K., L. MONTEFUSCO und L. PUCCIO (Herausgeber): *Wavelets: Theory, Algorithms, and Applications*, Seiten 213–230. Academic Press, San Diego, 1994.
- [COS01] CHAN, T., S. OSHER und J. SHEN: *The digital TV filter and nonlinear denoising*. *IEEE Trans. Image Process.*, 10(2):231–241, 2001.
- [CSS08] CHAN, R. H., S. SETZER und G. STEIDL: *Inpainting by flexible Haar-wavelet shrinkage*. *SIAM J. Imaging Sci.*, 1(3):273–293, 2008.
- [CZ00] CHAN, T. und H. ZHOU: *Total variation improved wavelet thresholding in image compression*. In: *Proceedings of the IEEE International Conference on Image Processing*, Band 2, Seiten 391–394, Vancouver, 2000.
- [dBDVR93] BOOR, C. DE, R. A. DE VORE und A. RON: *On the construction of multivariate (pre) wavelets*. *Constr. Approx.*, 9(2):123–166, 1993.
- [dBHR93] BOOR, C. DE, K. HÖLLIG und S. RIEMENSCHNEIDER: *Box Splines*. Springer, New York, 1993.

- [DDI06] DEMARET, L., N. DYN und A. ISKE: *Image compression by linear splines over adaptive triangulations*. Signal Process., 86(7):1604–1616, 2006.
- [DE03] DONOHO, D. L. und M. ELAD: *Optimally sparse representation in general (non-orthogonal) dictionaries via l^1 minimization*. Proc. Natl. Acad. Sci. USA, 100(5):2197–2202, 2003.
- [DF03] DURAND, S. und J. FROMENT: *Reconstruction of wavelet coefficients using total variation minimization*. SIAM J. Sci. Comput., 24(5):1754–1767, 2003.
- [DGH00] DONOVAN, G. C., J. S. GERONIMO und D. P. HARDIN: *Compactly supported, piecewise affine scaling functions on triangulations*. Constr. Approx., 16(2):201–219, 2000.
- [DHRS03] DAUBECHIES, I., B. HAN, A. RON und Z. SHEN: *Framelets: MRA-based constructions of wavelet frames*. Appl. Comput. Harmon. Anal., 14(1):1–46, 2003.
- [DJ94] DONOHO, D. L. und I. JOHNSTONE: *Ideal spatial adaptation by wavelet shrinkage*. Biometrika, 81(3):425–455, 1994.
- [DMA97] DAVIS, G., S. MALLAT und M. AVELLANEDA: *Adaptive greedy approximations*. Constr. Approx., 13(1):57–98, 1997.
- [DMW07] DIDAS, S., P. MRÁZEK und J. WEICKERT: *Energy-based image simplification with nonlocal data and smoothness terms*. In: ISKE, A. und J. LEVESLEY (Herausgeber): *Algorithms for Approximation*, Seiten 51–60. Springer, Heidelberg, 2007.
- [Don99] DONOHO, D. L.: *Wedgelets: Nearly-minimax estimation of edges*. Ann. Statist., 27(3):859–897, 1999.
- [DV02] Do, M. N. und M. VETTERLI: *Contourlets: A new directional multiresolution image representation*. In: *Conference Record of the Asilomar Conference on Signals, Systems and Computers*, Band 1, Seiten 497–501, 2002.
- [DV05] Do, M. N. und M. VETTERLI: *The contourlet transform: an efficient directional multiresolution image representation*. IEEE Trans. Image Process., 14(12):2091–2106, 2005.
- [Ehl07] EHLER, M.: *The Construction of Nonseparable Wavelet Bi-Frames and Associated Approximation Schemes*. Doktorarbeit, Philipps-Universität Marburg, Marburg, 2007.
- [ELC09] EASLEY, G. R., D. LABATE und F. COLONNA: *Shearlet-based total variation diffusion for denoising*. IEEE Trans. Image Process., 18(2):260–268, 2009.

- [ELL08] EASLEY, G. R., D. LABATE und W.-Q. LIM: *Sparse directional image representations using the discrete shearlet transform*. Appl. Comput. Harmon. Anal., 25(1):25–46, 2008.
- [EO04] ESEDOGLU, S. und S. OSHER: *Decomposition of images by the anisotropic Rudin-Osher-Fatemi model*. Comm. Pure Appl. Math., 57(12):1609–1626, 2004.
- [FDF06] FÜHR, H., L. DEMARET und F. FRIEDRICH: *Beyond wavelets: New image representation paradigms*. In: BARNI, M. und F. BARTOLINI (Herausgeber): *Document and Image Compression*, Seiten 179–206. CRC Press, Boca Raton, 2006.
- [FDFW07] FRIEDRICH, F., L. DEMARET, H. FÜHR und K. WICKER: *Efficient moment computation over polygonal domains with an application to rapid wedgelet approximation*. SIAM J. Sci. Comput., 29(2):842–863, 2007.
- [FK03] FODOR, I. K. und C. KAMATH: *Denoising through wavelet shrinkage: An empirical study*. J. Electron. Imaging, 12(1):151–160, 2003.
- [FQ01] FLOATER, M. S. und E. G. QUAK: *Piecewise linear wavelets over type-2 triangulations*. Comput. Supplement, 14:89–103, 2001.
- [GH06] GOODMAN, T. N. T. und D. P. HARDIN: *Refinable multivariate spline functions*. In: JETTER, K., M. D. BUHMANN, W. HAUßMANN, R. SCHABACK und J. STÖCKLER (Herausgeber): *Topics in Multivariate Approximation and Interpolation*, Seiten 55–83. Elsevier, Amsterdam, 2006.
- [GL07] GUO, F. und D. LABATE: *Optimally sparse multidimensional representation using shearlets*. SIAM J. Math. Anal., 39(1):298–318, 2007.
- [GLL⁺06] GUO, F., W.-Q. LIM, D. LABATE, G. L. WEISS und E. N. WILSON: *Wavelets with composite dilations and their MRA properties*. Appl. Comput. Harmon. Anal., 20(2):202–236, 2006.
- [GLT93] GOODMAN, T. N. T., S. L. LEE und W. S. TANG: *Wavelet bases for a set of commuting unitary operators*. Adv. Comput. Math., 1(1):109–126, 1993.
- [GM92] GRÖCHENIG, K. und W. R. MADYCH: *Multiresolution analysis, Haar bases, and self-similar tilings of \mathbb{R}^n* . IEEE Trans. Inform. Theory, 38(2/II):556–568, 1992.
- [Gol94] GOLOMB, S. W.: *Polyominoes*. Princeton University Press, Princeton, 2. Auflage, 1994.
- [Goo97] GOODMAN, T. N. T.: *Constructing pairs of refinable bivariate spline functions*. In: LE MÉHAUTÉ, A., C. RABUT und L. L. SCHUMAKER (Herausgeber): *Surface Fitting and Multiresolution Methods*, Seiten 145–162, Nashville, 1997.

- [Haa10] HAAR, A.: *Zur Theorie der orthogonalen Funktionensysteme*. Math. Ann., 69(3):331–371, 1910.
- [Han08] HAN, B.: *Construction of wavelets and framelets by the projection method*. Int. J. Appl. Math. Anal. Appl., 1(1):1–40, 2008.
- [HL97] HE, W. und M. J. LAI: *Examples of bivariate nonseparable continuous compactly supported orthonormal wavelets*. In: *Proceedings of SPIE*, Band 3169, Seiten 303–314, 1997.
- [HL99] HE, W. und M. J. LAI: *Construction of bivariate compactly supported biorthogonal box spline wavelets with arbitrarily high regularities*. Appl. Comput. Harmon. Anal., 6(1):53–74, 1999.
- [Jia99] JIANG, Q.: *Multivariate matrix refinable functions with arbitrary matrix dilation*. Trans. Amer. Math. Soc., 351(6):2407–2438, 1999.
- [Kei04] KEINERT, F.: *Wavelets and Multiwavelets*. Chapman & Hall/CRC, Boca Raton, 2004.
- [Kin01] KINGSBURY, N. G.: *Complex wavelets for shift invariant analysis and filtering of signals*. Appl. Comput. Harmon. Anal., 10(3):234–253, 2001.
- [KM10] KROMMWEH, J. und J. MA: *Tetrolet shrinkage with anisotropic total variation minimization for image approximation*. Signal Process., 90(8):2529–2539, 2010.
- [Kor04] KORN, M. R.: *Geometric and algebraic properties of polyomino tilings*. Doktorarbeit, Massachusetts Institute of Technology, 2004.
- [KP09] KROMMWEH, J. und G. PLONKA: *Directional Haar wavelet frames on triangles*. Appl. Comput. Harmon. Anal., 27(2):215–234, 2009.
- [Kro09a] KROMMWEH, J.: *An orthonormal basis of directional Haar wavelets on triangles*. Results Math., 53(3-4):323–331, 2009.
- [Kro09b] KROMMWEH, J.: *Tight frame characterization of multiwavelet vector functions in terms of the polyphase matrix*. Int. J. Wavelets Multiresolut. Inf. Process., 7(1):9–21, 2009.
- [Kro10] KROMMWEH, J.: *Tetrolet transform: A new adaptive Haar wavelet algorithm for sparse image representation*. J. Vis. Commun. Image R., 21(4):364–374, 2010.
- [KRWW07] KRISHTAL, I. A., B. D. ROBINSON, G. L. WEISS und E. N. WILSON: *Some simple Haar-type wavelets in higher dimensions*. J. Geom. Anal., 17(1):87–96, 2007.

- [KV95] KOVAČEVIĆ, J. und M. VETTERLI: *Nonseparable two- and three-dimensional wavelets*. IEEE Trans. Signal Process., 43(5):1269–1273, 1995.
- [Lar37] LARSSON, H.: *Problem 2623*. Fairy Chess Review, 3(5):51, 1937.
- [Lim06] LIM, W.-Q.: *Wavelets with Composite Dilations*. Doktorarbeit, Washington University in St. Louis, St. Louis, 2006.
- [LLKW05] LABATE, D., W.-Q. LIM, G. KUTYNIOK und G. L. WEISS: *Sparse multidimensional representation using shearlets*. In: PAPADAKIS, M., A. F. LAINE und M. A. UNSER (Herausgeber): *Proceedings of SPIE*, Band 5914, Seiten 254–262, Bellingham, 2005.
- [LPM05a] LE PENNEC, E. und S. MALLAT: *Bandelet image approximation and compression*. Multiscale Model. Simul., 4(3):992–1039, 2005.
- [LPM05b] LE PENNEC, E. und S. MALLAT: *Sparse geometric image representations with bandelets*. IEEE Trans. Image Process., 14(4):423–438, 2005.
- [LS07] LAI, M. J. und L. L. SCHUMAKER: *Spline Functions on Triangulations*. Encyclopedia of Mathematics. Cambridge University Press, Cambridge, 2007.
- [Ma05] MA, J.: *Towards artifact-free characterization of surface topography using complex wavelets and total variation minimization*. Appl. Math. Comput., 170(2):1014–1030, 2005.
- [Ma07] MA, J.: *Characterization of textural surfaces using wave atoms*. Applied Physics Letters, 90(26):Art.-Nr. 264101, 2007.
- [Mal89] MALLAT, S.: *Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$* . Trans. Amer. Math. Soc., 315(1):69–87, 1989.
- [Mal99] MALLAT, S.: *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 2. Auflage, 1999.
- [Mal09] MALLAT, S.: *Geometrical grouplets*. Appl. Comput. Harmon. Anal., 26(2):161–180, 2009.
- [MC97] MEYER, F. G. und R. R. COIFMAN: *Brushlets: A tool for directional image analysis and image compression*. Appl. Comput. Harmon. Anal., 4(2):147–187, 1997.
- [Mey90] MEYER, Y.: *Ondelettes et Opérateurs. I*. Actualités Mathématiques. Hermann, Paris, 1990.
- [MF06] MA, J. und M. FENN: *Combined complex ridgelet shrinkage and total variation minimization*. SIAM J. Sci. Comput., 28(3):984–1000, 2006.

- [MP07] MA, J. und G. PLONKA: *Combined curvelet shrinkage and nonlinear anisotropic diffusion*. IEEE Trans. Image Process., 16(9):2198–2206, 2007.
- [NS05] NEUMANN, J. und G. STEIDL: *Dual-tree complex wavelet transform in the frequency domain and an application to signal classification*. Int. J. Wavelets Multiresolut. Inf. Process., 3(1):43–65, 2005.
- [Pl09] PLONKA, G.: *Easy path wavelet transform: A new adaptive wavelet transform for sparse representation of two-dimensional data*. Multiscale Model. Simul., 7(3):1474–1496, 2009.
- [Pow74] POWELL, M. J. D.: *Piecewise quadratic surface fitting for contour plotting*. In: EVANS, D. J. (Herausgeber): *Software for Numerical Mathematics*, Seiten 253–271. Academic Press, London, 1974.
- [PTI09] PLONKA, G., S. TENORTH und A. ISKE: *Optimally sparse image representation by the easy path wavelet transform*. Preprint, 2009.
- [ROF92] RUDIN, L. I., S. OSHER und E. FATEMI: *Nonlinear total variation based noise removal algorithms*. Phys. D, 60(1–4):259–268, 1992.
- [RS97] RON, A. und Z. SHEN: *Affine systems in $L^2(\mathbb{R}^d)$: The analysis of the analysis operator*. J. Funct. Anal., 148(2):408–447, 1997.
- [Rue02] RUEDIN, A. M. C.: *Construction of nonseparable multiwavelets for nonlinear image compression*. EURASIP J. Appl. Signal Process., 2002(1):73–79, 2002.
- [SN96] STRANG, G. und T. NGUYEN: *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley, 2. Auflage, 1996.
- [ST09] STEIDL, G. und T. TEUBER: *Diffusion tensors for denoising sheared and rotated rectangles*. IEEE Trans. Image Process., 18(12):2640–2648, 2009.
- [ST10] STEIDL, G. und T. TEUBER: *Removing multiplicative noise by Douglas-Rachford splitting methods*. J. Math. Imaging Vision, 36(3):168–184, 2010.
- [SWB⁺04] STEIDL, G., J. WEICKERT, T. BROX, P. MRÁZEK und M. WELK: *On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and SIDEs*. SIAM J. Numer. Anal., 42(2):686–713, 2004.
- [TBEL07] TA, V., S. BOUGLEUX, A. ELMOATAZ und O. LEZORAY: *Nonlocal anisotropic discrete regularization for image, data filtering and clustering*. Technischer Bericht, Université de Caen Basse-Normandie, Caen, 2007.

- [TM98] TOMASI, C. und R. MANDUCHI: *Bilateral filtering for gray and color images*. In: *Proceedings of the IEEE International Conference on Computer Vision*, Seiten 839–846, Bombay, 1998.
- [Tro04] TROPP, J. A.: *Greed is good: Algorithmic results for sparse approximation*. IEEE Trans. Inform. Theory, 50(10):2231–2242, 2004.
- [VBLVD06] VELISAVLJEVIC, V., B. BEFERULL-LOZANO, M. VETTERLI und P. L. DRAGOTTI: *Direction-lets: Anisotropic multidirectional representation with separable filtering*. IEEE Trans. Image Process., 15(7):1916–1933, 2006.
- [VK95] VETTERLI, M. und J. KOVAČEVIĆ: *Wavelets and Subband Coding*. Prentice-Hall, Englewood Cliffs, 1995.
- [Woj97] WOJTASZCZYK, P.: *A Mathematical Introduction to Wavelets*, Band 37 der Reihe *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1997.
- [WWS05] WELK, M., J. WEICKERT und G. STEIDL: *A four-pixel scheme for singular differential equations*. In: KIMMEL, R., N. SOCHEN und J. WEICKERT (Herausgeber): *Scale-Space and PDE Methods in Computer Vision*, Band 3459 der Reihe *Lecture Notes in Computer Science*, Seiten 610–621. Springer, Berlin, 2005.
- [Zwa73] ZWART, P. B.: *Multivariate splines with nondegenerate partitions*. SIAM J. Numer. Anal., 10(4):665–673, 1973.

Abbildungsverzeichnis

1.1	Die klassische 2-Kanal-Filterbank.	20
1.2	Eine m -Kanal-Filterbank in Modulations-Darstellung.	23
1.3	Eine m -Kanal-Filterbank in Polyphasen-Darstellung.	25
2.1	Dreieckszerlegung mit vier Richtungen.	32
2.2	Konstruktion der gerichteten \mathcal{AB} -Mutterwavelets ψ_0^1, ψ_0^2 und ψ_0^3	34
2.3	Entstörung eines synthetischen Bildes.	36
2.4	Approximation des „Elaine“-Bildes.	37
2.5	Dreieckszerlegung mit acht Richtungen.	39
2.6	Redundanzen der \mathcal{AB} -Skalierungsfunktionen.	41
2.7	Konstruktion der gerichteten Mutterwavelets ψ_0^1, ψ_0^2 und ψ_0^3	44
2.8	Die Basisfunktionen der vier Unterräume V_0^0, V_0^1, V_0^2 und V_0^3	46
2.9	Berechnung der vier Koeffizienten $c_{0,k}^0, c_{1,k}^0, c_{8,k}^0, c_{9,k}^0$ aus vier Bildwerten. . .	50
2.10	Reihenfolge der Komponenten in den Funktionen-Vektoren $\Phi_0^0, \Phi_0^1, \Phi_0^2$ und Φ_0^3 . .	51
2.11	Entstörung eines synthetischen Bildes.	62
2.12	Entstörung des „pepper“-Bildes.	63
2.13	Approximation eines synthetischen Bildes.	64
2.14	Approximation des „pepper“-Bildes.	65
2.15	Dreiecksträger der Skalierungsfunktionen.	75
2.16	Die verfeinerten Skalierungsfunktionen.	76
2.17	Träger der Spline-Skalierungsfunktionen ϕ_1 und ϕ_2	83
2.18	Die biorthogonalen Splinefunktionen $\tilde{\phi}_1$ und $\tilde{\phi}_2$	86
2.19	Die Träger der Mutterwavelets.	89
2.20	Die Träger von $\psi^{(i)}$ für $i = 1, \dots, 4$	92
3.1	Die fünf freien Tetrominoes.	95
3.2	Die 22 Fundamental-Konfigurationen, die ein 4×4 -Quadrat überdecken. . . .	96
3.3	Beispiel für zwei verschiedene Möglichkeiten der Nummerierung der Tetrominoes.	101
3.4	Beispiel einer adaptiven Tetromino-Überdeckung eines Bildblocks.	103
3.5	Die a priori ausgewählten 16 Konfigurationen mit verschiedenen Richtungen. .	104

3.6	Testbilder.	107
3.7	Approximation des synthetischen Bildes mit 512 Koeffizienten.	109
3.8	Approximation des „cameraman“-Bildes mit 2048 Koeffizienten.	109
3.9	Approximation des „pepper“-Bildes mit 2048 Koeffizienten.	110
3.10	Approximation des „Lena“-Bildes mit 2048 Koeffizienten.	110
3.11	Approximation des 64×64 -Detailbildes von „monarch“ mit 256 Koeffizienten.	111
3.12	Approximation des 128×128 -Detailbildes von „Barbara“ mit 512 Koeffizienten.	111
3.13	Approximation des „pepper“-Bildes mit 2048 Koeffizienten für die TT16.	112
3.14	Verteilung der 5461 Adaptivitätswerte beim „cameraman“-Bild für die Relaxierte TT.	112
3.15	Approximation des „cameraman“-Bildes mit 2048 Koeffizienten für die Relaxierte TT.	113
3.16	Approximation des „clock“-Bildes mit 2048 Koeffizienten für die Haar-TT.	113
3.17	Bildfunktion f eines 16×16 -Bildes und ihre Tetrolet-Approximationen f_3, f_2, f_1 und f_0	121
3.18	Adaptive Tetromino-Überdeckung eines 16×16 -Bildes und seiner Tiefpassanteile.	122
4.1	Approximation eines linearen Signals mit anschließendem Postprocessing.	136
4.2	Testbilder.	138
4.3	Approximation des „clock“-Bildes.	138
4.4	Approximation eines stückweise glatten Bildes.	139
B.1	Skalarprodukte von ϕ_1 und ϕ_2 aus V_0	144
B.2	Skalarprodukte von ϕ_1, ϕ_2 aus V_0 und $\phi_1(2\cdot), \phi_2(2\cdot)$ aus V_1	145

Tabellenverzeichnis

3.1	Vergleich der modifizierten Tetrolet-Transformationen mit gerichteten Waveletmethoden bezüglich der Berechnungszeit.	106
3.2	Vergleich der PSNR-Werte bei Approximation.	108
3.3	Vergleich zwischen Tensor-Wavelettransformationen und verschiedenen Versionen der Tetrolet-Transformation bezüglich Approximationsqualität und Speicherkosten.	115