# Aufgabenbezogen progressive Repräsentation von Bildern durch Waveletnetze

vorgelegt von

**Alexander Pelc**

Kiel, im Januar 1997

**Diplomarbeit**

am Lehrstuhl für Kognitive Systeme

des Instituts für Informatik und Praktische Mathematik

Prof. Dr. G. Sommer

und

am Institut für Angewandte Physik

Prof. Dr. G. Pfister

der Christian–Albrechts–Universität zu Kiel

betreut durch Dipl.-Ing. Udo Mahlmeister

## Zusammenfassung

Das Ziel der Bildkompression ist es, Information zur Übertragung oder Speicherung von Bildern oder Bildfolgen zu unterdrücken. Die Irrelevanz stellt dabei den größten Anteil unterdrückbarer Information dar. Unter Irrelevanz versteht man den Anteil des Signals, der für die Lösung einer Aufgabe keine Bedeutung hat. Bei modernen wahrnehmungsbezogenen Bildkompressionsverfahren besteht Irrelevanzreduktion aus zwei getrennten Stufen, der Unterabtastung (Verminderung der Ortsauflösung) und der Quantisierung (Verminderung der Grauwertauflösung). In der vorgelegten Arbeit wird der Versuch unternommen, beide Komponenten in Form eines Approximationsverfahrens durch *Waveletnetze* miteinander zu verschmelzen. Waveletnetze sehen die Approximation des Grauwertverlaufes eines Bildes durch eine Menge in Skala, Orientierung und Ort optimal angepaßter Wavelets vor. Die Einstellung der Waveletparameter erfolgt über ein geeignetes Optimierungsverfahren, wobei die Anfangsparameter aus Ergebnissen eines vorgeschalteten Attentionsmechanismus' abgeleitet werden. Dadurch wird sowohl eine schnellere Konvergenz als auch eine stufenlose Beeinflussung der Approximationsgüte über das Attentionsmerkmal im Sinne einer *aufgabenbezogenen* Bildkompression erreicht. Der Kompressionsgewinn dieser neuen *Repräsentation* gründet dabei auf der Approximation des Grauwertverlaufes durch die irreguläre Abtastung mit wenigen optimal angepaßten Wavelets im Gegensatz zur Wavelettransformation, bei der die Abtastung durch das reguläre Schema einer Pyramide stattfindet. Die durch das Attentionsmerkmal bestimmte Optimierungsreihenfolge stellt in jedem Übertragungsschritt einen maximalen Informationsgewinn im Sinne einer optimalen *progressiven* Übertragung sicher. Der Einfluß der Aufgabenstellung auf die lokale Anpassung der Wavelets ist darüberhinaus durch eine geeignete Regularisierung der Parameter möglich. Das Verfahren wurde auf Sun-Workstations innerhalb des Bildverarbeitungspaketes KHOROS 2.0 implementiert.

# Inhaltsverzeichnis

# Kapitel 1

# Einführung

Ein Grauwertbild kann als diskrete $2D$-Funktion aufgefaßt werden, bei der jedem Bildpunkt (Pixel) $(x, y)$ ein Grauwert $g(x, y)$ zugeordnet wird. Die Basis dieser Signalrepräsentation ist der Delta-Impuls. Läßt man auf der Grauwertskala $256 = 2^8$ Grauwerte zu, so werden 8 Bit oder 1 Byte zur Speicherung bzw. Übertragung des Grauwertes eines Pixels benötigt. Bei einem $M \times N$ Bild sind dies $M \cdot N$ Byte, so daß große Bilder entsprechend viel Speicherplatz benötigen. Zum einen wird je nach Art des Informationsträgers (Menge der Bildpunkte) viel redundante Information gespeichert. Zum anderen, abhängig von der Aufgabenstellung, unter der ein Bild verarbeitet wird, ist auch Information gespeichert, die für die Lösung der Aufgabe keine Bedeutung hat, sogenannte Irrelevanz. Das Ziel der Bildkompression besteht nun darin, diese beiden unerwünschten Anteile des Signals bei der Speicherung bzw. Übertragung zu unterdrücken. Dies erreicht man in der Regel durch zwei getrennte Bildverarbeitungsstufen. Die eine Stufe realisiert die Unterabtastung des Signals, d.h. die Verminderung der Ortsauflösung. Die andere Stufe realisiert die Quantisierung, d.h. die Verminderung der Grauwertauflösung auf beispielsweise $64 = 2^6$ statt $256 = 2^8$ Grauwerte. Bei modernen wahrnehmungsbezogenen Bildkompressionsverfahren sucht man zunächst eine andere Signalrepräsentation. Dies kann z.B. der Ortsfrequenzraum sein, wie es bei der Fouriertransformation der Fall ist. Die zur Signalanalyse oft eingesetzte Methode der Short-Time-Fourier-Transformation nutzt in allen Frequenzbereichen des Signals dasselbe Zeit-Fenster zur Analyse. Im Gegensatz dazu ist die Wavelettransformation (WT) sehr interessant für die Analyse von nichtstationären Signalen, da sie „kurze Zeit-Fenster an hohen Frequenzen und lange Zeit-Fenster an niedrigen Frequenzen“, wie sich Rioul und Vetterli in [11] ausdrücken, zur Signalanalyse nutzt. Ein heute oft beschrittener Weg führt über die diskrete Wavelettransformation (DWT), die eine lokalspektrale Repräsentation des Signals darstellt und bei der eine Projektion des Signals auf eine Basis, die Waveletbasis, stattfindet:

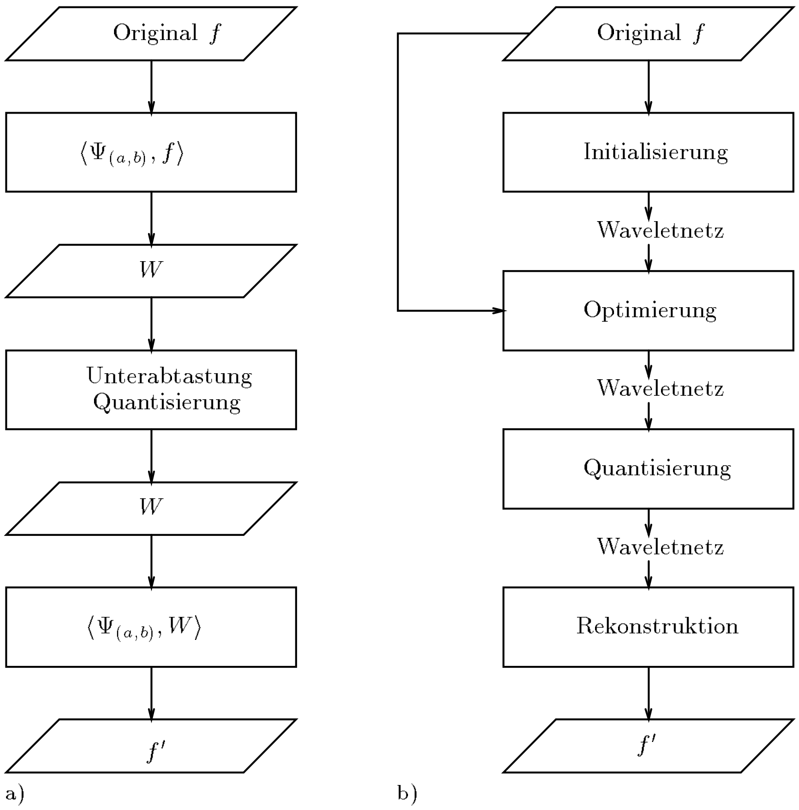



Abbildung 1.1: Schematischer Ablauf der Bildverarbeitungsschritte bei (a) der Wavelettransformation und (b) beim Konzept der Waveletnetze.

$$
\begin{aligned}
W_{(a,b)} &= \langle \psi_{(a,b)}, f \rangle && (1.1)\\
&= \sum_{m=0}^{M-1} f_m \, \psi_{(a,b)}^m, \qquad \psi_{(a,b)}^m = a^{-\frac{1}{2}} \, \psi\left(\frac{m-b}{a}\right). && (1.2)
\end{aligned}
$$

Die Waveletbasis $\psi_{(a,b)}(x)$ besteht aus skalierten und verschobenen Versionen des sogenannten Mother Wavelet $\psi(x)$, das im Gegensatz zur STFT frei und an das Problem angepaßt wählbar ist. Die Diskretisierung des Parameterraumes $(a, b)$ von Ort (man interpretiert den Ort in Analogie zur STFT auch als Zeit) und Skala (als alternative zur Frequenz bei der STFT) folgt dabei dem regulären Abtastschema einer Pyramide. Dieses Abtastschema führt zu einer sogenannten „constant-Q-Analyse“. Der Raum der Waveletkoeffizienten $W_{(a,b)}$ wird quantisiert, um eine Datenreduktion zu erreichen. Zur Decodierung wird das Signal zurück in den Ortsraum transformiert. Die Bildverarbeitungsschritte bei der DWT sind in Abbildung 1.1 (a) schematisch dargestellt.

Bei dem in dieser Arbeit verfolgten Konzept der Waveletnetze wird keine Wavelettransformation im eben beschriebenen Sinne durchgeführt. Das Signal wird durch eine Menge optimal angepaßter Wavelets approximiert, wobei die Abtastung im Gegensatz zur (DWT) irregulär und einer Aufgabenstellung entsprechend erfolgt. Die oben genannten Eigenschaften der (WT) zur Analyse nichtstationärer Signale sollen zur geeigneten Initialisierung eines Waveletnetzes genutzt werden.

Mit diesem ersten Verarbeitungsschritt beschäftigt sich das Kapitel 2 eingehend. Dort wird diskutiert, in welcher Art und Weise die die Struktur Waveletnetz die universelle Approximationseigenschaft der Neuronalen Netze mit der Wavelettransformation verbindet, um geeignete Initialwerte der Parameter für die Optimierung abzuschätzen. Im zweiten Schritt müssen die Wavelets optimal an das Signal angepaßt werden. Dies ist ein wesentlicher Unterschied zur DWT, denn im Laufe der Optimierung wird die Waveletbasis ebenso angepaßt wie die Koeffizienten. Kapitel 3 dieser Arbeit ist deshalb dem Approximationsalgorithmus gewidmet. Dort werden auch die mit den ersten Approximationsversuchen auftauchenden Probleme, wie z.B. die Regularisierung der Parameter, gelöst. Der bisher wenig diskutierte aber sehr wichtige Aspekt der progressiven Optimierung rundet das dritte Kapitel ab. Die Anfangsparameter der Optimierung werden durch einen Attentionsmechanismus so ermittelt, daß eine aufgabenbezogene Approximationsgüte erreicht und die Konvergenz der Approximation beschleunigt wird. Die Attention ist deshalb Gegenstand des vierten Kapitels. Hier werden ein passiver Ansatz, bei dem aus der Grauwertkrümmung im Bild attentiv bedeutsame Punkte berechnet werden, und ein aktiver Ansatz, bei dem der Aufgabensteller attentiv wichtige Regionen bestimmt, verfolgt. Mit der Quantisierung der optimierten Waveletparameter und der Rekonstruktion des komprimierten Bildes aus dem optimal angepaßten Waveletnetz ist auch die Entwicklung des Verfahrens der *progressiven Wavelet-Approximation* (PWA) beendet. Abbildung 1.1 (b) veranschaulicht die einzelnen Schritte noch einmal im Gegensatz zur Wavelettransformation. Das so entwickelte Verfahren wird im Kapitel fünf ausführlich unter verschiedenen Aspekten der Bildkompression getestet. Diese Auswertung beinhaltet sowohl eine qualitative als auch quantitative Untersuchung der Entwicklung der Approximationsgüte bei der Rekonstruktion des optimierten Netzes als auch einen Vergleich der Methode mit einer klassischen Variante der Bildkompression durch die DWT der *Subband-Codierung* (SBC). Ein Resumee, in dem zur Bearbeitung der eben gestellten Aufgabe Stellung genommen wird, sowie einige Perspektiven für spätere Entwicklungen, die vielleicht auf dieser Arbeit aufbauen, runden die vorliegende Arbeit ab.

# Kapitel 2

# Waveletnetze

## 2.1 Wavelets

Wavelets gehören in ein sich schnell entwickelndes Gebiet der Mathematik, jedoch wurden sie innerhalb weniger Jahre zu einem unabhängigen Forschungszweig. Mehr und mehr baut dieses Forschungsgebiet Brücken zu allen möglichen Disziplinen aus Wissenschaft und Technik und eröffnet mittlerweile ein breites Anwendungsgebiet.

### 2.1.1 Was sind Wavelets?

Die folgende Erklärung ist ein Versuch, den Begriff des Wavelets etwas näherzubringen. Eine strenge Definition wird bei der Einführung der Wavelettransformation geliefert. Eine recht einfache und prägnante Formulierung findet sich in [14]. Wavelets sind Funktionen, die ganz bestimmten Bedingungen genügen. Der Name *Wavelets* trägt der folgenden Bedingung Rechnung.

$$\int_{\mathbb{R}^n} \psi(x)dx = 0 \tag{2.1}$$

Man bezeichnet dies als „waving“ um die $x$-Achse. So wie der Sinus und der Cosinus bei der Fourier-Analyse werden Wavelets als Basisfunktionen genutzt bei der Repräsentation von Signalen. Ist einmal das sogenannte *Mother Wavelet* $\psi(x)$ festgelegt, so erhält man eine Basis durch Translation und Dilation des Mother Wavelets.

$$\psi_{a;b} = a^{-\frac{1}{2}}\psi\Big(\frac{x-b}{a}\Big), \qquad (a,b) \in \mathbb{R}^+ \times \mathbb{R} \tag{2.2}$$

Es bleibt frei, die Parameter $a$ und $b$ zu wählen um eine Basis zu bilden, z.B. $a = 2^{-j}$ und $b = k \cdot 2^j$, wobei $k, j \in \mathbb{N}$.

Viele Klassen von Funktionen können durch Wavelets kompakter repräsentiert werden. Zum Beispiel Funktionen mit Diskontinuitäten lassen sich, wie wir später sehen werden, aufgrund der „Zooming-Eigenschaft“ lokal optimal durch Wavelets anpassen. Diese Eigenschaft macht sie zu sehr hilfreichen Werkzeugen in der Datenkompression. Auch in anderen Bereichen sind Wavelets erfolgreich eingesetzt worden, Coifman nutzte sie zum glätten von verauschten Bandaufnahmen, und der Geophysiker Morlet analysierte seismische Daten mit Wavelets. Wie I. Daubechies in [5] bewiesen hat, ist es möglich, orthogonale Waveletbasen zu konstruieren. Es gibt also ein breites Anwendungsgebiet für Wavelets, in denen die Operationen auf den Waveletkoeffizienten angewendet werden. Dazu müssen die Daten zunächst auf die Waveletbasis transformiert werden. Dies geschieht durch die diskrete Wavelettransformation (DWT), die im nächsten Abschnitt diskutiert werden soll.

### 2.1.2 Die Wavelettransformation

Wie wir im letzten Abschnitt festgestellt haben, sind Wavelets Basisfunktionen, die ganz bestimmten Bedingungen genügen. Um die Wavelettransformation fundiert erläutern zu können, müssen wir diese Eigenschaften herausstellen, wofür sich am besten ein kleiner historischer Exkurs anbietet, wie er in [8] sehr gut vollzogen wird. Mathematiker wie Haar, Littlewood, Paley und Franklin haben sich schon in den frühen 30er und 40er Jahren dieses Jahrhunderts mit Wavelets beschäftigt, allerdings unter dem Namen „atomic decomposition“. Ein Zugang zu dieser Theorie bietet Calderon's Identität (1960). Diese Theorie hat die Zerlegung des Identitätsoperators zur Grundlage. Zu dieser Zeit hatten die Mathematiker jedoch noch keinen allgemeinen Formalismus entwickelt. Zwei Physiker, Grossman und Morlet, haben 1980, 20 Jahre nach Calderon's Arbeit, die Identitätszerlegung wieder aufgegriffen, und lieferten eine Definition und eine Herangehensweise, ausgehend von ihrer physikalischen Intuition, die so flexibel ist, daß sie alle wichtigen Eigenschaften umfaßt. Beginnend mit der Grossman-Morlet Definition präsentieren wir zwei weitere Definitionen, die den Vorgang der Diskretisierung beschreiben und unseren später zu gehenden Weg ebnen sollen. Um Calderon's Identität zu erläutern, starten wir mit einer Funktion $\psi(x)$ im $L^2(\mathbb{R}^n)$. Grossman und Morlet bezeichneten diese Funktion als *Analyse-Wavelet*. Sie definieren ein Wavelet wie folgt:

**Definition 1:** Ein Wavelet ist eine Funktion $\psi(x)$ in $L^2(\mathbb{R}^n)$, dessen Fouriertransformierte $\hat{\psi}(\xi)$ die Bedingung

$$\int_0^\infty |\hat{\psi}(t\xi)|^2 \frac{dt}{t} = 1 \tag{2.3}$$

erfüllt. Wenn $\psi(x) \in L^1(\mathbb{R}^n)$, dann impliziert (2.3)

$$\int_{\mathbb{R}^n} \psi(x) dx = 0. \tag{2.4}$$

Die zweite Definition bezieht sich auf die Littlewood-Paley-Stein-Theorie.

**Definition 2:** Ein Wavelet ist eine Funktion $\psi(x)$ in $L^2(\mathbb{R}^n)$, dessen Fouriertransformierte $\hat{\psi}(\xi)$ die Bedingung

$$\sum_{j=-\infty}^{\infty} |\hat{\psi}(2^{-j}\xi)|^2 = 1 \tag{2.5}$$

erfüllt. Bei spezieller Wahl von $t$ erfüllt solch ein Wavelet auch die Grossman-Morlet-Bedingung. Die dritte Definition entspringt der Arbeit von Franklin und Strömberg.

**Definition 3:** Ein Wavelet ist eine Funktion $\psi(x)$ in $L^2(\mathbb{R}^n)$, so daß $2^{\frac{j}{2}}\psi(2^j x - k)$ wobei $j, k \in \mathbb{Z}$, eine Orthonormalbasis für $L^2(\mathbb{R}^n)$ bilden.

Solch ein Wavelet erfüllt zwangsweise die zweite Bedingung. Gehen wir von der ersten zur dritten Definition, so erweitert sich die Zahl der Bedingungen und wir grenzen den Raum der Wavelets immer mehr ein. Dies gilt auch für die Waveletzerlegung einer Funktion. Grossman und Morlet definieren das Analyse-Wavelet $\psi(x)$ als

$$\psi_{(a,b)}(x) = a^{-\frac{n}{2}} \psi\left(\frac{x-b}{a}\right), \qquad a > 0, b \in \mathbb{R}^n. \tag{2.6}$$

Die Waveletzerlegung einer Funktion $f(x)$ führt zu einer Funktion $W(a,b)$ von $n+1$ Variablen. Diese Funktion ist definiert durch

$$W(a,b) = \langle f(x), \psi_{(a,b)}(x) \rangle, \tag{2.7}$$

wobei $\langle u, v \rangle = \int u(x)\, v^*(x)\, dx$.

In der Littlewood-Paley-Theorie wird $a = 2^{-j}$. Mit anderen Worten, ist $\Gamma$ die multiplikative Gruppe $\{2^{-j}, j \in \mathbb{Z}\}$, so erhält man die Littlewood-Paley-Analyse durch Einschränken der Grossman-Morlet-Analyse auf $\Gamma \times \mathbb{R}^n$.

In der Franklin-Strömberg-Theorie wird $a = 2^{-j}$ und $b = k \cdot a^{-1}$ wobei $j, k \in \mathbb{Z}$. Mit anderen Worten, man erhält die Franklin-Strömberg–Analyse durch Einschränkung der Littlewood-Paley-Analyse auf ein *hyperbolisches* Gitter $S$ in $(0, \infty) \times \mathbb{R}$, das durch die Punkte $(2^{-j}, k2^{-j})$, wobei $j, k \in \mathbb{Z}$, gebildet wird.

Wir bezeichnen für die weiteren Betrachtungen mit $\tilde{\psi}(x) = \psi^*(-x)$ die duale Basis, sowie $\psi_t(x) = t^{-n}\,\psi(\frac{x}{t})$ und $\tilde{\psi}_t(x) = t^{-n}\,\tilde{\psi}(\frac{x}{t})$. $Q_t$ ist der Faltungsoperator $Q_t(f) = f * \psi_t$ und $Q_t^*$ sein adjungierter $Q_t^*(f) = f * \tilde{\psi}_t$.

Calderon's Identität, die mit der Grossman-Morlet-Analyse identisch ist, beschreibt die Zerlegung des Identitätsoperators

$$I = \int_0^\infty Q_t Q_t^* \frac{dt}{t}. \tag{2.8}$$

Dies wird zu $I = \sum_{-\infty}^{\infty} \triangle_j \triangle_j^*$ in der Littlewood–Paley–Analyse, und wenn $t = 2^{-j}$ wird $Q_t(f)$ zu $\triangle_j(f)$, sowie das Integral $\int_0^\infty u(t)\frac{dt}{t}$ zur Summe $\sum_{-\infty}^{\infty} u(2^{-j})$.

Um die Beziehung zwischen der Littlewood-Paley- und der Franklin-Strömberg-Theorie herzustellen, schreiben wir $\psi_j(x) = 2^j\psi(2^j x)$ und $\tilde{\psi}_j(x) = 2^j\tilde{\psi}(2^j x)$. $\triangle_j(f) = f * \psi_j$ ist der Faltungsoperator und $\triangle_j^* = f * \tilde{\psi}_j$ sein Adjungierter. Die Koeffizienten $\alpha(j, k)$ in Strömberg's Orthonormalbasis sind dann gegeben durch:

$$\begin{aligned} \alpha(j,k) &= 2^{\frac{j}{2}} \int f(x)\psi^*(2^j x - k)dx & (2.9)\\ &= 2^{-\frac{j}{2}}(f * \tilde{\psi}_j)(k2^{-j}) & (2.10)\\ &= 2^{-\frac{j}{2}}\triangle_j^* f(k2^{-j}). & (2.11) \end{aligned}$$

Somit erhält man die Koeffizienten durch Einschränkung von $\triangle_j^*$ auf das hyperbolische Gitter. In allen drei Fällen folgt der Analyse die Synthese zur Rekonstruktion der Funktion $f(x)$ aus ihrer Wavelettransformierten. Im Grossman-Morlet-Falle sieht die Rekonstruktion wie folgt aus:

$$f(x) = \int_0^\infty \int_{\mathbb{R}^n} W(a,b)\,\psi_{(a,b)}(x)db\frac{da}{a^{n+1}}. \tag{2.12}$$

Im Falle der Littlewood–Paley–Analyse wird das Integral $\int_0^\infty u(a)\frac{da}{a}$ zur Summe $\sum_{j=-\infty}^{\infty} u(2^{-j})$ und (2.12) wird zu:

$$f(x) = \sum_{-\infty}^{\infty} \int_{\mathbb{R}^n} (\triangle_j^* f)(b)\, \psi_j(x-b) db \frac{da}{a^{n+1}}. \tag{2.13}$$

Und schließlich wird für den eindimensionalen Fall und für Strömberg's orthogonale Wavelets das letzte Integral zur Summe und (2.13) zu:

$$f(x) = \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} \alpha(j,k)\, \psi_{j,k}(x). \tag{2.14}$$

Somit sind wir bei der Diskreten-Wavelet-Transformation (DWT) angelangt. Die eben gemachten Betrachtungen beziehen sich nur auf den $L^2(\mathbb{R})$, was aber für die folgende Arbeit völlig ausreicht. Wie weiter oben schon erwähnt, kann mit Hilfe der DWT eine Datenkompression erreicht werden. Dazu ist es notwendig eine geeignete Basis zu wählen, auf die das Signal projiziert wird. In der so erhaltenen Waveletzerlegung des Signales ist die Information in den Waveletkoeffizienten $\alpha(j,k)$ enthalten, wobei man davon ausgeht, daß der Informationsgehalt kleiner Koeffizienten eher gering ist. Durch anschließende Schwellwertoperationen und Quantisierung eben dieser Koeffizienten erreicht man die Minimierung des Codes, der zur Speicherung der Signalrepräsentation notwendig ist und somit die angestrebte Kompression. Im Gegensatz zur Wavelettransformation, in der die Basis ein für alle mal gewählt wird, wird sie bei der Anpassung in Waveletnetzen eingehend auf die Signalstruktur modelliert. Und hier kommt der Begriff der Waveletnetze ins Spiel, der im folgenden Abschnitt etwas näher gebracht wird.

## 2.2 Vom Wavelet zum Netz

Den Schritt vom Wavelet zum Waveletnetz zur universellen Approximation gehen wir mit Zhang und Benveniste, die in [15] den Begriff „Wavelet Network“ als eine Komposition aus neuronaler Feed-Forward-Architektur und der Waveletzerlegung eingeführt haben. Deshalb wird zunächst die Rolle von Neuronalen Netzen zur universellen Approximation von Funktionen untersucht. Anschließend soll eine direkte Beziehung zwischen den Netzwerkkoeffizienten und der Wavelettransformation hergestellt werden, die schließlich zu der neuen Struktur *Waveletnetz* führt.

### 2.2.1 Neuronale Netze zur Approximation

Seit langem werden Neuronale Netze mit Feed-Forward-Architektur zur Approximation beliebiger n-dimensionaler Funktionen untersucht. Galka hat in [6] Feed-

Forward-Netze analogelektronisch realisiert und beschreibt die zugrundeliegende Architektur sehr treffend. Feed-Forward-Architektur bedeutet, daß die Informationen während der Lernphase nur von einer Schicht zur nächsten, aber nie zurück, gegeben werden. Es bestehen auch innerhalb der einzelnen Schichten keine Verbindungen. Solche Netze wurden 1958 von F. Rosenblatt [12] eingeführt und *Perzeptrone* genannt. Die einfachsten bestehen aus nur einer datenverarbeitenden Schicht sowie einer Eingabe- und einer Ausgabeschicht. Die beiden letzten Schichten werden in der Regel nicht mitgezählt.

Die Ergebnisse der Beschreibung der Approximationseigenschaft dieser Neuronalen Netze werden in [3] im einzelnen geprüft und von Zhang und Benveniste wie folgt zusammengefaßt: Ist $\sigma(\cdot)$ eine kontinuierliche diskriminatorische Funktion, dann ist die Summe der Form

$$g(\boldsymbol{x}) = \sum_{i=1}^{N} \omega_i \sigma(\boldsymbol{a}_i^T \boldsymbol{x} + b_i) \tag{2.15}$$

dicht im Raum der auf $[0,1]^n$ definierten kontinuierlichen Funktionen, wobei $\omega_i$, $b_i \in \mathbb{R}$ und $\boldsymbol{a}_i \in \mathbb{R}^n$. Mit anderen Worten, für eine gegebene auf $[0,1]^n$ definierte Funktion $f$ und ein $\epsilon > 0$ gibt es eine Summe der Form (2.15), für die $|g(\boldsymbol{x}) - f(\boldsymbol{x})| < \epsilon$ für alle $\boldsymbol{x} \in [0,1]^n$ gilt.

Als Vertreter dieser Netze sollen hier die *Radialen-Basis-Funktions-Netze* (RBF-Netze) (siehe auch [9]) der Veranschaulichung dienen. Abbildung 2.1 zeigt die Architektur eines RBF-Netzes. Die mittlere Schicht enthält sogenannte Radiale-Basis-Funktions-Knoten. Als Radiale-Basis-Funktionen wählt man meist die Gauß'sche Aktivierungsfunktion. Im Falle einer linearen Ausgabeschicht berechnet sich die Ausgabe wie folgt.

$$y_j(\boldsymbol{x}) = \sum_{i=1}^{N} \omega_{ij} e^{-\frac{(\boldsymbol{x}-\boldsymbol{\mu}_i)^2}{\boldsymbol{\sigma}_i^2}} \tag{2.16}$$

Die Gewichte $\omega_{ij}$ sowie die Skalen $\boldsymbol{\sigma}_i$ und die Zentren $\boldsymbol{\mu}_i$ müssen durch geeignete Lernalgorithmen angepaßt werden. Die Algorithmen für diese Anpassungen gehören zur Backpropagation-Klasse. Der Begriff Backpropagation trägt der Tatsache Rechnung, daß der Approximationsfehler in der Ausgabeschicht berechnet und in die davor liegenden Schichten „zurück projiziert" wird. Anschließend werden die Parameter ($\boldsymbol{\sigma}_i$, $\boldsymbol{\mu}_i$) dieser Schichten und die Gewichte ($\omega_{ij}$) so verändert, daß der Ausgabefehler minimiert wird. Im Laufe der Arbeit wird ein Vertreter dieser Klasse zur Anwendung kommen und an entsprechender Stelle diskutiert.

Damit der Weg, den Zhang und Benveniste gegangen sind, um die Waveletnetze zu motivieren, von uns verfolgt werden kann, sollen ihre Anforderungen an die neue Struktur herausgestellt werden:

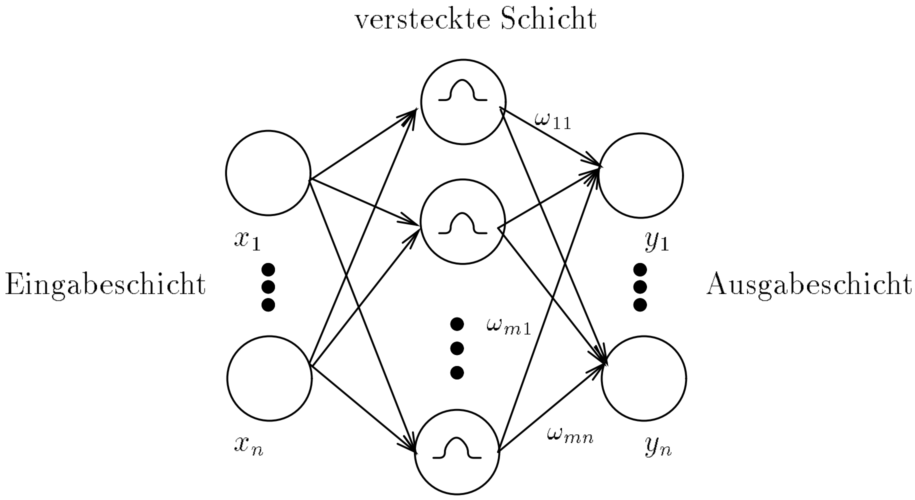



Abbildung 2.1: Die Feed-Forward-Architektur eines RBF-Netzes.

- Die universelle Approximationseigenschaft muß erfüllt sein, d.h. es ist eine Klasse von Netzwerken gesucht, die dieselben Eigenschaften hat wie (2.15).
- Es muß explizit eine Beziehung zwischen den Netzwerkkoeffizienten und einer geeigneten Transformation hergestellt werden. Diese Eigenschaft wird sehr nützlich sein, um die Anfangswerte für den Optimierungsalgorithmus zu bestimmen.

Im nächsten Abschnitt wird die Wavelettransformation als die geeignete Transformation speziell für diese Anforderungen noch einmal diskutiert.

### 2.2.2 Die Wavelettransformation als universelle Approximation

Das Ziel ist es, eine Funktion $\psi : \mathbb{R}^n \mapsto \mathbb{R}$ mit der folgenden Eigenschaft zu finden: Es gibt eine Familie der Form

$$\Psi = \left\{ \det(\boldsymbol{D}_k^{\frac{1}{2}})\, \psi[\boldsymbol{D}_k \boldsymbol{x} - \boldsymbol{t}_k] \,:\, \boldsymbol{t}_k \in \mathbb{R}^n, \boldsymbol{D}_k = \operatorname{diag}(\boldsymbol{d}_k), \boldsymbol{d}_k \in \mathbb{R}_+^n, k \in \mathbb{Z} \right\} \quad (2.17)$$

die die folgende *Frame*-Eigenschaft erfüllen: Es gibt zwei Konstanten $c_{min} > 0$ und $c_{max} < \infty$, so daß für alle $f$ in $L^2(\mathbb{R}^n)$ die Ungleichungen

$$c_{min}\|f\|^2 \leq \sum_{\varphi\in\Psi} |\langle\varphi, f\rangle|^2 \leq c_{max}\|f\|^2 \tag{2.18}$$

erfüllt sind. Es ist eine Konsequenz von (2.18), daß die Familie $\Psi$ dicht in $L^2(\mathbb{R}^n)$ ist. Die Qualität der Approximation für eine feste Anzahl Terme in der Summe ist abhängig davon, wie nahe die Koeffizienten $c_{min}$ und $c_{max}$ bei 1 liegen. Daraus folgt, daß die Summe aller Linearkombinationen der Elemente von $\Psi$

$$g(\boldsymbol{x}) = \sum_{k=1}^{N} \omega_k \varphi_k(\boldsymbol{x}), \qquad \varphi_k \in \Psi \tag{2.19}$$

dicht in $L^2(\mathbb{R}^n)$ ist. Darüberhinaus folgt auch, daß die Summe aller endlichen Summen der Form

$$g(\boldsymbol{x}) = \sum_{i=1}^{N} \omega_i \det(\boldsymbol{D}_i^{\frac{1}{2}})\, \psi[\boldsymbol{D}_i\boldsymbol{x} - \boldsymbol{t}_i], \tag{2.20}$$

wobei die $\boldsymbol{t}_i$'s die willkürlichen Translationsvektoren und die $\boldsymbol{d}_i$'s die willkürlichen Dilationsvektoren sind, die die diagonale Dilationsmatrix $\boldsymbol{D}$ bestimmen, dicht in $L^2(\mathbb{R}^n)$ ist, da sie alle endlichen Linearkombinationen der Elemente von $\Psi$ enthält. Um die Familien $\Psi$ zu bestimmen, ziehen wir die Wavelet-Theorie heran. Die kontinuierliche Wavelettransformation erlaubt es uns, jede Funktion $f(\boldsymbol{x}) \in L^2(\mathbb{R}^n)$ mit einer Familie von Funktionen zu zerlegen, die durch Translation und Dilation des Mother Wavelet $\psi : \mathbb{R}^n \mapsto \mathbb{R}$ gebildet werden. Betrachten wir zunächst ein *skalares* „Grossman-Morlet-Wavelet", d.h. eine Funktion $\psi_s : \mathbb{R} \mapsto \mathbb{R}$ mit der Fouriertransformierten $\hat{\psi}_s(\xi)$, die die Bedingung

$$C_{\psi_s} = \int_0^{+\infty} \frac{|\hat{\psi}_s(\xi)|^2}{\xi}\, d\,\xi < \infty \tag{2.21}$$

erfüllt. Mit diesem skalaren Wavelet ist es möglich, das gesuchte Wavelet $\psi : \mathbb{R}^n \mapsto \mathbb{R}$ durch

$$\psi(\boldsymbol{x}) = \psi_s(x_1)\cdots\psi_s(x_n) \qquad \text{für} \qquad \boldsymbol{x} = (x_1, \cdots, x_n) \tag{2.22}$$

zu konstruieren. Führen wir außerdem die Konstante

$$C_\psi = C_{\psi_s}^n \tag{2.23}$$

ein, so gilt die Wavelettransformation in der Form

$$f(\boldsymbol{x}) = \frac{1}{C_\psi} \int\limits_{\mathbb{R}^n} \int\limits_{\mathbb{R}^n_+} \Omega(\boldsymbol{d}, \boldsymbol{t}) (\det \boldsymbol{D})^{\frac{1}{2}} \, \psi[\boldsymbol{D}(\boldsymbol{x} - \boldsymbol{t})] \, d\boldsymbol{s} \, d\boldsymbol{t} \tag{2.24}$$

$$\Omega(\boldsymbol{d}, \boldsymbol{t}) = \int\limits_{\mathbb{R}^n} f(\boldsymbol{x}) (\det \boldsymbol{D})^{\frac{1}{2}} \, \psi[\boldsymbol{D}(\boldsymbol{x} - \boldsymbol{t})] \, d\boldsymbol{x}. \tag{2.25}$$

Zu diesem Ergebnis kommt I. Daubechies in [5]. Dort hat sie auch gezeigt, daß die Familie

$$\Psi_s(\alpha, \beta) = \left\{ \alpha^{\frac{k}{2}} \, \psi_s(\alpha^k \, x - \beta l) \, : \, k, l \in \mathbb{Z} \right\} \tag{2.26}$$

die Frame-Bedingung (2.18) für eine geeignete Wahl der Parameter $(\alpha, \beta)$ erfüllt. Häufig verwendete Wavelets, wie z.B. die „Morlet-Wavelets“, liefern $c_{min}$ und $c_{max}$ nahe bei 1 für $\alpha = 2$ und $\beta = 1$. Zhang und Benveniste zeigen in [15], daß (2.18) auch für direkte Produkte der Elemente von $\psi_s(\alpha, \beta)$ gilt, d.h.

$$\Psi(\alpha, \beta) = \left\{ (\det \boldsymbol{D})^{\frac{1}{2}} \, \psi(\boldsymbol{D}\boldsymbol{x} - \beta \boldsymbol{l}) \, : \, k, l \in \mathbb{Z} \right\}, \tag{2.27}$$

wobei $\boldsymbol{d} = (d_1, \cdots, d_n)$ und $\boldsymbol{D} = \mathrm{diag}(\alpha^{d_1}, \cdots, \alpha^{d_n})$ gilt, was der geforderten Form (2.17) entspricht. Wenn wir stattdessen $\psi$ so wählen, daß

$$C_\psi = \int\limits_0^{+\infty} \frac{|\hat{\psi}_s(h\xi)|^2}{h} \, d\,h \tag{2.28}$$

existiert und unabhängig von $\xi$ ist (Isotropiebedingung), dann sind (2.24) sowie die Frame-Eigenschaft ebenfalls erfüllt, was gleibedeutend ist damit, daß neben separierbaren auch isotrope Wavelets verwendet werden können.

Im Ergebnis dieser Diskussion sollen die Anforderungen, die eingangs gestellt wurden, überprüft werden.

- Die universelle Approximationseigenschaft wird durch endliche Summen der Form (2.20) erfüllt, wenn $\psi$ ein Wavelet entsprechend der eben diskutierten Form ist.
- Die explizite Beziehung zwischen Netzwerkkoeffizienten und der Wavelettransformation ist hergestellt.

Dieser zweite Aspekt wird deutlicher, wenn man die Transformation (2.24) für den diskreten Fall betrachtet. Dann wird (2.24) angenähert durch die Summe

$$f(\boldsymbol{x}) \approx \sum_{i=1}^{N} \omega_i (\det \boldsymbol{D})^{\frac{1}{2}} \, \psi[\boldsymbol{D}_i \, (\boldsymbol{x} - \boldsymbol{t}_i)]. \tag{2.29}$$

$\Omega(\boldsymbol{s}, \boldsymbol{t})$ erhält man aus $f(\boldsymbol{x})$ mit Hilfe der diskreten Form der Analyseformel (2.25)

$$\omega_i = \int_{\mathbb{R}^n} f(\boldsymbol{x}) (\det \boldsymbol{D}_i)^{\frac{1}{2}} \, \psi[\boldsymbol{D}_i \, (\boldsymbol{x} - \boldsymbol{t}_i)] \, d\,\boldsymbol{x}. \tag{2.30}$$

Entsprechend der verschiedenen Typen der Approximation (2.19) und (2.20) nennen wir (2.19) Wavelettransformation, und in Analogie zu (2.15) im Falle der Neuronalen Netze bezeichnen wir (2.20) mit Waveletnetz. Bei der Wavelettransformation sind die Gewichte $\omega_i$ die einzigen freien Parameter und erscheinen als Koeffizienten bei der Projektion $\langle f, \psi \rangle$ auf die Basis $\psi$, während die Dilation und die Translation dem regulären Gitter folgen. Im Gegensatz dazu werden beim Waveletnetz auch die Dilations- und Translationsparameter an die Funktion $f(\boldsymbol{x})$ angepaßt.

### 2.2.3 Parametrisierung eines Waveletnetzes

Es ist in der Anwendung üblich, die Dilation auch mit Skala zu bezeichnen. Wir übernehmen für die weiteren Betrachtungen diese Notation und bezeichnen die Dilationsmatrix $\boldsymbol{D}$ mit $\boldsymbol{S}$ und den Dilationsvektor $\boldsymbol{d}$ mit $\boldsymbol{s}$. Ausgehend von der Netzwerkstruktur (2.29) ändern wir diese etwas ab, indem wir noch zwei Größen einführen, die Rotation $\alpha$ und den (redundanten) Gleichwert $\bar{g}$. Die endgültige Struktur hat die Form

$$g(\boldsymbol{x}) = \sum_{i=1}^{N} \omega_i \, \psi[\boldsymbol{S}_i \boldsymbol{R}_i (\boldsymbol{x} - \boldsymbol{t}_i)] + \bar{g}. \tag{2.31}$$

Die $\boldsymbol{R}_i$ 's sind die aus $\alpha$ gebildeten Rotationsmatrizen, die die Orientierung eines Wavelets als sechsten Parameter spezifizieren und das Netz flexibler machen. Die Größe $\bar{g}$ wurde eingeführt, um den Mittelwert der zu approximierenden Funktionen zu tragen. Bei Grauwertbildern entspricht $\bar{g}$ dem Gleichwert des Bildes.

$$\bar{g} = \frac{\sum_{i=1}^{N} f_i(x)}{N}$$

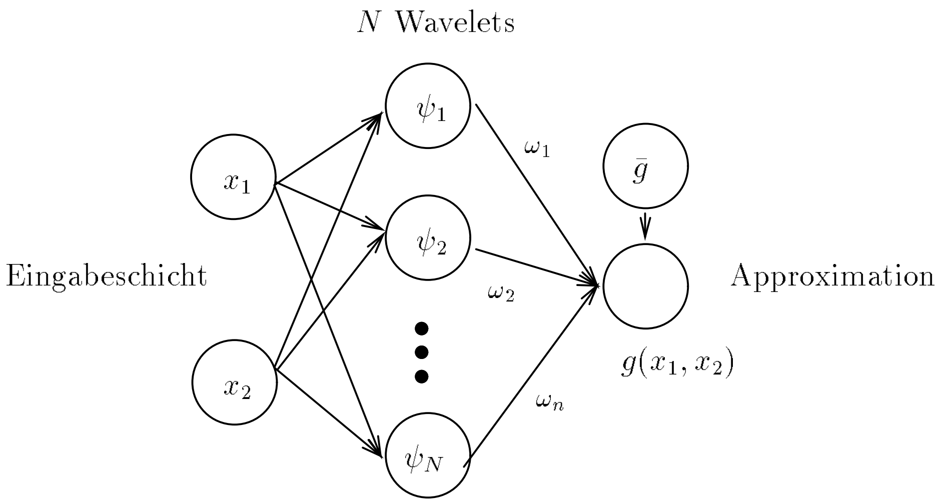



Abbildung 2.2: Die Architektur eines Waveletnetzes.

Abbildung 2.2 veranschaulicht die so erhaltene Waveletnetzstruktur.

Abschließend soll der Parametersatz eines Wavelets noch einmal zusammengefaßt und der Parametrisierung der diskreten Wavelettransformation gegenüber gestellt werden. Unser Waveletnetz in Abbildung 2.2 ist durch folgende Parametermenge gekennzeichnet:

**1. Translation (Waveletzentren):**

$$\boldsymbol{t} = (t_1, t_2)^T$$

**2. Dilation (Waveletskalen):**

$$\boldsymbol{S} = \mathrm{diag}\,(s_1, s_2)$$

**3. Gewichte:**

$$\omega$$

**4. Orientierungen:**

$$\boldsymbol{R} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix}$$

Es ergibt sich somit für jedes Wavelet ein Parametervektor

$$\boldsymbol{a} = (t_1, t_2, s_1, s_2, \omega, \alpha). \tag{2.32}$$

Im Gegensatz zur DWT können Waveletzentren außerhalb des Definitionsbereiches der Funktion liegen. Die Skalen und die Rotationen können beliebige Werte annehmen. Die wesentlichen Unterschiede der Parametermenge zur DWT sind in Tabelle 2.1 gegenübergestellt.

| Parameter | DWT | Waveletnetz |
|---|---|---|
| $t_1, t_2$ | $2^n$ | $\mathbb{R}$ |
| $s_1, s_2$ | $s_0\, 2^n$ | $\mathbb{R}^+$ |
| $\alpha$ | $0, \frac{\pi}{2}$ | $[0, \pi[$ |
| $\omega$ | $\mathbb{R}$ | $\mathbb{R}$ |

Tabelle 2.1: Die Gegenüberstellung der Wertebereiche der Parametermenge von DWT und Waveletnetz.

Wird das Waveletnetz durch $N$ Wavelets gebildet, erhalten wir einen Gesamtparametervektor $\boldsymbol{a} = (\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N)$, der im Laufe der Approximation angepaßt werden muß. Durch die Freistellung aller sechs Parameter bleibt die universelle Approximationseigenschaft erhalten.

### 2.2.4 Initialisierung eines Waveletnetzes

Um das Waveletnetz zwecks Approximation an einen Algorithmus weitergeben zu können, brauchen wir Anfangswerte für alle Parameter. Ein erster Versuch, diese abzuschätzen, nutzt das Prinzip der Auflösungshierarchien und Bildpyramiden. In [13] wird dieses Prinzip wie folgt beschrieben. Eine Auflösungshierarchie nutzt die Freiheitsgrade der Skala, indem sie an einem Ort gleichzeitig verschiedene Filtergrößen, d.h. verschiedene Auflösungen, bereitstellt. Das bekannteste Beispiel einer Auflösungshierarchie ist die Gaußpyramide, deren Skala durch die Breite der Gaußfunktion $\sigma$ gegeben ist. Bei Einhaltung des Abtasttheorems können wir die Stufen bei wachsender Skala mit zunehmend großen Intervallen abtasten, ohne Information zu verlieren. Wir wollen die Gaußfunktion, die nach $x$ und $y$ separierbar ist, als Basis einer solchen Auflösungshierarchie verwenden.

$$\psi(\boldsymbol{x}) = e^{-\left(\frac{x^2}{2\sigma_x{}^2} + \frac{y^2}{2\sigma_y{}^2}\right)} = e^{-\frac{x^2}{2\sigma_x{}^2}}\, e^{-\frac{y^2}{2\sigma_y{}^2}} \tag{2.33}$$

Somit können wir uns bei der Diskussion der Initialisierung der Skalen $\boldsymbol{s} = (s_1, s_2)$ und der Zentren $\boldsymbol{t} = (t_1, t_2)$ auf eine Dimension beschränken, für die zweite erfolgt sie analog. Die Pyramide wird von der Spitze her aufgebaut. Da sich in jeder Stufe der Pyramide die Abtastrate verdoppelt, gehen zwei Größen in die Initialisierung ein, die Bildgröße und die Stufennummer. Diese Art der Abtastung bedeutet,

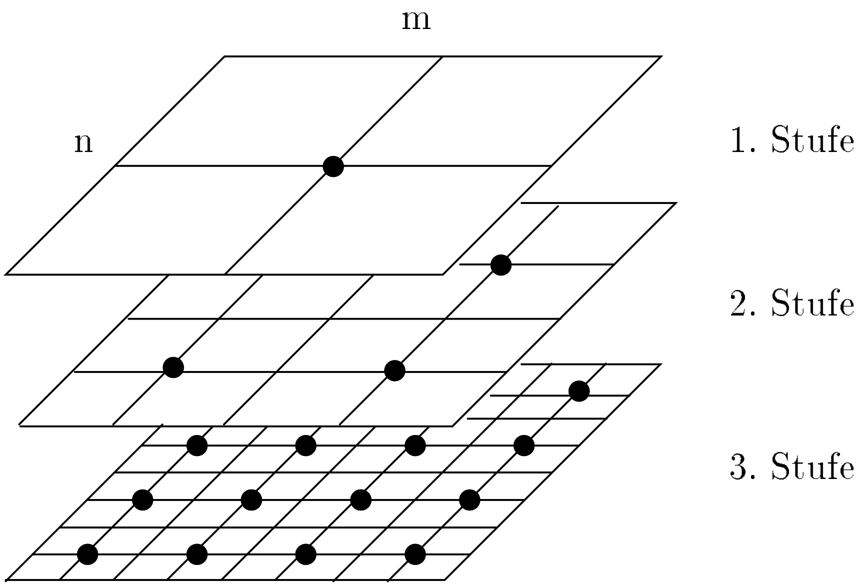



Abbildung 2.3: Die Initialisierung der Waveletzentren bei einer 3-stufigen Pyramide.

daß sich die Anzahl Wavelets in jeder Stufe und Dimension verdoppelt. In einer $2D$-Pyramide vervierfacht sich also deren Zahl von Stufe zu Stufe. Abbildung 2.3 veranschaulicht diese Abtastung.

Die Initialisierung der Skalen erfolgt ähnlich. Bei Betrachtung der Gaußfunktion 2.33 ergibt sich deren Skala als das Reziproke der Skala des Wavelets $\psi(S\,\boldsymbol{x})$ zu $\sigma = \frac{1}{s}$. Eine plausible Strategie ist es, die Skala von der Bildgröße $m$ abhängig zu machen, was man dadurch erreicht, daß man die Bildgröße zur Skala macht. In jeder weiteren Stufe wird sie halbiert. Dabei muß jedoch noch eine Konstante $c$ eingeführt werden, die den Maßstab des Bildes repräsentiert. Wir wollen die Skala so wählen, daß die Gaußfunktion am Bildrand auf die Hälfte ihrer Amplitude in der Bildmitte $(\frac{m}{2}, \frac{n}{2})$ abgefallen ist. Daher berechnet sich die Skala in einer Dimension wie folgt.

$$\begin{array}{rcccc} \psi(m) & = & e^{-\frac{(m-\frac{m}{2})^2}{2\sigma^2}} & \stackrel{!}{=} & \frac{1}{2} \\ & \Leftrightarrow & \sigma & = & \frac{m}{c} \\ & \Rightarrow & c & = & 2\sqrt{2\ln 2} \end{array}$$

Es ergibt sich also folgende Bestimmung für die Skala der Stufe $n$.

$$\sigma_n = \frac{m}{c\,2^{n-1}}, \qquad s_n = \frac{1}{\sigma_n}, \qquad c = 2\sqrt{2ln2}$$

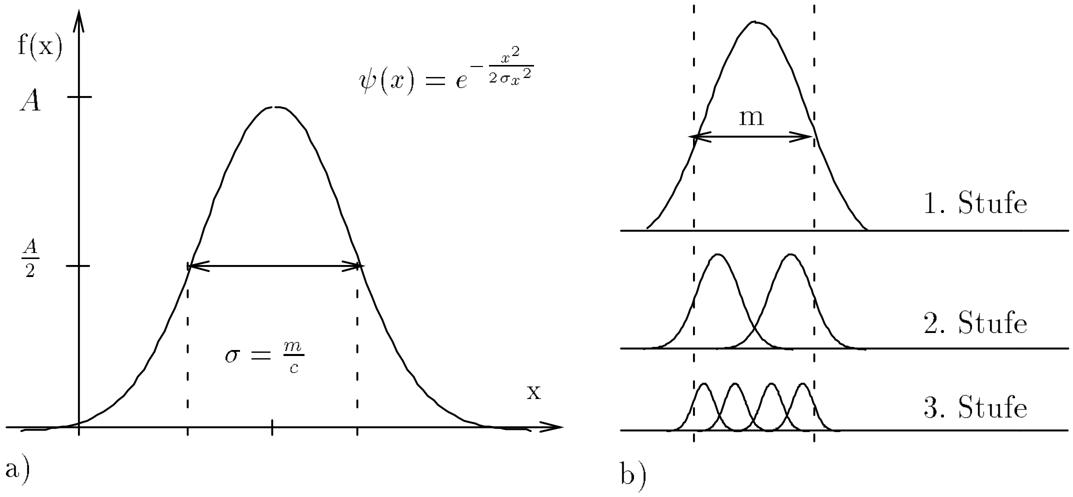



Abbildung 2.4: Initialisierung der Skalen (a) in der 1. Stufe, die Gaußfunktion fällt am Bildrand auf die Hälfte ab (b) Schnitt durch eine dreistufige Pyramide (schematisch).

Diese Strategie der Initialisierung der Skala ist in Abbildung 2.4 veranschaulicht. Entsprechend wird die zweite Dimension behandelt, wodurch die ersten vier Parameter initialisiert sind. In unserem Parametervektor $\boldsymbol{a} = (t_1, t_2, s_1, s_2, \omega, \alpha)$ stehen noch die Gewichte und Orientierungen aus. Bei der Initialisierung der Gewichte lassen wir uns von der Überlegung leiten, daß bei der späteren Optimierung kleine Skalen die Anpassung der feinen Strukturen übernehmen sollen, was unter anderem dadurch erreicht wird, daß jede Pyramidenstufe, angefangen mit der ersten, einzeln optimiert wird und den Restfehler bei der Projektion der vorhergehenden minimieren soll. Diese Strategie reduziert den Approximationsfehler in jeder Stufe, und somit wird auch bei der Übertragung der einzelnen Stufen der Informationsgewinn in jedem Rekonstruktionsschritt (Stufe) maximiert. Dies legt nahe, angefangen bei einem geeigneten $\omega_0$ auch die Initialgewichte mit steigender Stufe zu verringern. Aus der Amplitudenstatistik des Fourierspektrums natürlicher Bilder ist die Proportionalität $A(f) \sim \frac{1}{f}$ bekannt. Bei einem $m \times n$ Bild kann man demnach die Gewicht in der Form

$$\omega_n = \frac{\omega_0}{2^{(n-1)}}, \qquad \omega_0 = \sqrt{mn}$$

berechnen. Eine andere Überlegung geht davon aus, schon bei der Initialisierung der Gewichte Bildinformation zu nutzen. Hierbei werden die Gewichte der Wavelets durch eine Projektion auf das Bild gemäß der Analyseformel (2.30) berechnet. Wir werden später beide Strategien als Phasenraum-Startpunkte für den Optimierungsalgorithmus einsetzen. Der letzte Parameter, die Orientierung $\alpha$, wird willkürlich auf einen Startwert gesetzt. Auch hier ist größtenteils Heuristik im Spiel, wenn man sich zwei geeignete Orientierungen, z.B. $\alpha = 0$ und $\alpha = \frac{\pi}{2}$, vorgibt. Dabei entspricht eine um $\frac{\pi}{2}$ gedrehte Ableitung der Gaußfunktion in $x$-Richtung gerade der ersten Ableitung in $y$-Richtung.

$$\psi^0(\boldsymbol{x}) = -\frac{x}{{\sigma_x}^2 + {\sigma_y}^2} e^{-\frac{x^2+y^2}{2({\sigma_x}^2+{\sigma_y}^2)}} \qquad \psi^{\frac{\pi}{2}}(\boldsymbol{x}) = -\frac{y}{{\sigma_x}^2 + {\sigma_y}^2} e^{-\frac{x^2+y^2}{2({\sigma_x}^2+{\sigma_y}^2)}}$$

Somit stehen also zwei geeignete Strategien zur Initialisierung unserer Waveletpyramide zur Verfügung. Der nun folgende Schritt ist die Anpassung des Waveletnetzes an das Signal durch einen geeigneten Optimierungsalgorithmus, der Gegenstand des nächsten Kapitels ist.

# Kapitel 3

# Approximation

Das hier zugrundeliegende Modell des Waveletnetzes (2.31) hängt nichtlinear von sechs Parametern pro Wavelet, bei $N$ Wavelets also von $6\,N$ Parametern $a_k, k = 1, 2, ..., 6\,N$ ab. Wollen wir dieses Modell an das Signal, unser Bild, anpassen, müssen wir eine Fehlerfunktion $\chi^2(\boldsymbol{a})$ definieren und die Parameter $\boldsymbol{a}_{min}$ finden, die diese Fehlerfunktion minimieren. Da die Abhängigkeit von den Parametern nichtlinear ist, muß die Minimierung iterativ erfolgen. Angefangen mit einem Satz Initialparameter (wie im letzten Kapitel diskutiert) wird solange iteriert, bis der Fehler sein (relatives) Minimum erreicht.

## 3.1 Die Minimierung der Fehlerfunktion

Üblicherweise wählt man die Fehlerfunktion als Summe der Fehlerquadrate, damit sich Fehler mit unterschiedlichen Vorzeichen nicht gegenseitig wegheben. In [10] (Kap. 10 und 14) geht man den Weg der Minimierung wie folgt. Die Fehlerfunktion $\chi^2$ läßt sich, wie jede Funktion, durch eine quadratische Form, d.h. die Taylorreihe zweiter Ordnung, approximieren, so daß

$$f(\boldsymbol{x}) = f(\boldsymbol{x_0}) + \sum_i \frac{\partial f}{\partial x_i} x_i + \frac{1}{2} \sum_{i,j} \frac{\partial^2 f}{\partial x_i \partial x_j} x_i x_j + \cdots \tag{3.1}$$

$$\approx \gamma - \boldsymbol{b} \cdot \boldsymbol{x} + \frac{1}{2} \boldsymbol{x}^T \cdot \boldsymbol{B} \cdot \boldsymbol{x} \tag{3.2}$$

wobei

$$\gamma \equiv f(\boldsymbol{x_0}), \qquad \boldsymbol{b} \equiv -\nabla f|_{x_0}, \qquad \text{und} \qquad [\boldsymbol{B}]_{ij} \equiv \left. \frac{\partial^2 f}{\partial x_i \partial x_j} \right|_{x_0}$$

ist. Jedes Wavelet ist durch einen Parametervektor $\boldsymbol{a} = (t_1, t_2, s_1, s_2, \omega, \alpha)$, die Modellfunktion des gesamten Netzes also durch einen Vektor $\boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_N)$ gekennzeichnet. Unsere Fehlerfunktion (3.2) wird zu:

$$\chi^2(\boldsymbol{a}) \approx \gamma - \boldsymbol{b} \cdot \boldsymbol{a} + \frac{1}{2}\, \boldsymbol{a}^T \cdot \boldsymbol{B} \cdot \boldsymbol{a}. \tag{3.3}$$

Der Gradient von (3.3) berechnet sich zum derzeitigen Wert $\boldsymbol{a}_{curr}$ zu

$$\nabla\, \chi^2(\boldsymbol{a}_{curr}) = -\boldsymbol{b} + \boldsymbol{B} \cdot \boldsymbol{a}_{curr} \tag{3.4}$$

wobei er im Minimum $\boldsymbol{a}_{min}$ verschwindet:

$$0 = -\boldsymbol{b} + \boldsymbol{B} \cdot \boldsymbol{a}_{min}. \tag{3.5}$$

Die Differenz der beiden Gleichungen (3.5) und (3.4) führt auf

$$\boldsymbol{B}(\boldsymbol{a}_{min} - \boldsymbol{a}_{curr}) = -\nabla\, \chi^2(\boldsymbol{a}_{curr}) \tag{3.6}$$

bzw.

$$\boldsymbol{B}\, \delta\, \boldsymbol{a} = -\nabla\, \chi^2(\boldsymbol{a}_{curr}). \tag{3.7}$$

Vorausgesetzt die Inverse der Hessematrix $\boldsymbol{B}^{-1}$ existiert, kann bei einer guten Approximation der Fehlerfunktion durch die Taylorreihe zweiter Ordnung in einem einzigen Sprung

$$\boldsymbol{a}_{min} = \boldsymbol{a}_{cur} + \boldsymbol{B}^{-1}[-\nabla\chi^2(\boldsymbol{a}_{curr})] \tag{3.8}$$

das Minimum $\boldsymbol{a}_{min}$ vom derzeitigen Wert $\boldsymbol{a}_{curr}$ erreicht werden (Hesse-Methode).

Ist jedoch die Approximation schlecht, so werden die eben gemachten Betrachtungen obsolet, und es muß auf die Methode des Gradientenabstiegs zurückgegriffen werden, also

$$\boldsymbol{a}_{next} = \boldsymbol{a}_{curr} - constant\, [\nabla\chi^2(\boldsymbol{a}_{curr})] \tag{3.9}$$

bzw.

$$\delta\, \boldsymbol{a} = -\, constant\, [\nabla\chi^2(\boldsymbol{a}_{curr})]. \tag{3.10}$$

Beide Methoden zu einer zu vereinen ist das Ziel der Levenberg-Marquardt-Methode. Zunächst soll jedoch die Form unserer Fehlerfunktion und die Berechnung des Gradienten $\boldsymbol{b}$ sowie der Hessematrix $\boldsymbol{B}$ diskutiert werden.

### 3.1.1 Die Berechnung des Gradienten und der Hessematrix

Das Modell, welches an unser Grauwertbild angepaßt werden soll, ist (2.31) und soll im Folgenden mit

$$y = y(x; \boldsymbol{a}) = g(x) \tag{3.11}$$

bezeichnet werden. Die Summe der Fehlerquadrate ergibt sich dann zu

$$\chi^2(\boldsymbol{a}) = \frac{1}{2} \sum_{i=1}^{N} [y_i - y(x_i; \boldsymbol{a})]^2. \tag{3.12}$$

Der Gradient von $\chi^2$ ist Null im Minimum und hat die Form

$$\frac{\partial \chi^2}{\partial a_k} = -\sum_{i=1}^{N} [y_i - y(x_i; \boldsymbol{a})] \frac{\partial y(x_i; \boldsymbol{a})}{\partial a_k} \qquad k = 1, 2, \cdots, M. \tag{3.13}$$

Leitet man diesen Ausdruck noch einmal ab, so erhält man die gemischten Ableitungen

$$\frac{\partial^2 \chi^2}{\partial a_k \partial a_l} = \sum_{i=1}^{N} \Big[ \frac{\partial y(x_i; \boldsymbol{a})}{\partial a_k} \frac{\partial y(x_i; \boldsymbol{a})}{\partial a_l} - [y_i - y(x_i; \boldsymbol{a})] \frac{\partial^2 y(x_i; \boldsymbol{a})}{\partial a_k \partial a_l} \Big]. \tag{3.14}$$

Führt man zur Abkürzung die folgende Notation ein

$$\beta_k \equiv -\frac{\partial \chi^2}{\partial a_k}, \qquad \alpha_{kl} \equiv \frac{\partial^2 \chi^2}{\partial a_k \partial a_l}, \qquad \text{und} \qquad [\alpha] = \boldsymbol{B},$$

so kann Gleichung (3.7) geschrieben werden als

$$\sum_{k=1}^{M} \alpha_{kl} \delta a_l = \beta_k \qquad , l = 1, \cdots, M. \tag{3.15}$$

Der Gradientenabstieg (3.10) wird dann zu

$$\delta a_l = constant\, \beta_l. \tag{3.16}$$

Bei Betrachtung der Hessematrix (3.14) fällt auf, daß sie sowohl von ersten als auch zweiten Ableitungen abhängt. Der Faktor der zweiten Ableitungen ist $[y_i - y(x_i; \boldsymbol{a})]$. Bei einem guten Modell sollte dieser Wert klein sein und etwa dem zufälligen Meßfehler in diesem Punkt entsprechen. Da er mit positivem und negativem Vorzeichen auftritt, ist dieser Fehler im Allgemeinen unkorelliert mit dem Modell und kann bei Summation vernachlässigt werden. Zur Vereinfachung schreiben wir die Hessematrix im Folgenden als:

$$\alpha_{kl} = \sum_{i=1}^{N} \Big[ \frac{\partial y(x_i; \boldsymbol{a})}{\partial a_k} \frac{\partial y(x_i; \boldsymbol{a})}{\partial a_l} \Big]. \tag{3.17}$$

### 3.1.2 Die Levenberg-Marquardt Methode

Die Levenberg-Marquardt-Methode ist eine elegante Kombination aus Hesse-Methode und dem Gradientenabstieg, da sie zwischen beiden Extremen variiert, je nachdem ob wir nahe am Minimum sind oder nicht. Ausgehend von der Gleichung (3.16) ist die Konstante in Betrag und Einheit zu bestimmen. $\beta_k$ hat die Dimension $\frac{1}{a_k}$. Die Proportionalitätskonstante zwischen $\beta_k$ und $\delta a_k$ muß deshalb die Dimension $a_k^2$ haben. Betrachtet man die Komponenten von $[\alpha]$, so trifft dies nur auf $\frac{1}{\alpha_{kk}}$ zu, also auf die Reziproke der Diagonalelemente. Mit einer geeigneten Skalierung, z.B. $\lambda$, können diese Diagonalelemente die Proportionalitätskonstante stellen. Somit wird aus (3.16)

$$\delta a_l = \frac{1}{\lambda \alpha_{ll}} \beta_l \qquad \text{oder} \qquad \lambda \alpha_{ll} \delta a_l = \beta_l. \tag{3.18}$$

Es ist notwendig, daß $\alpha_{ll}$ positiv ist, was aber durch seine Definition (3.14) garantiert wird. Marquardt's Einfall war es nun, die beiden Gleichungen (3.15) und (3.16) zu einer zu kombinieren durch die Einführung einer neuen Matrix $\alpha'$:

$$\begin{aligned} \alpha'_{jj} &= \alpha_{jj}(1+\lambda) \\ \alpha'_{jk} &= \alpha_{jk}, \qquad (j \neq k). \end{aligned} \tag{3.19}$$

Dann werden die beiden Gleichungen (3.15) und (3.16) zu

$$\sum_{k=1}^{M} \alpha'_{kl} \delta a_l = \beta_k, \qquad l = 1, \cdots, M. \tag{3.20}$$

Ist $\lambda$ groß, so wird die Diagonale in $\alpha'$ dominant und (3.20) geht über in (3.16). Andererseits, geht $\lambda$ gegen Null, so geht (3.20) über in (3.15). Hat man nun eine Menge von Anfangsparametern $\boldsymbol{a}$, so empfiehlt Marquardt folgende Schritte.

- Berechne $\chi^2(\boldsymbol{a})$
- Wähle ein geeignetes $\lambda$ z.B. $\lambda = 0.001$
- ($\star$) Löse das Gleichungssystem (3.20) für $\delta\boldsymbol{a}$ und berechne $\chi^2(\boldsymbol{a} + \delta\boldsymbol{a})$
- Wenn $\chi^2(\boldsymbol{a} + \delta\boldsymbol{a}) \geq \chi^2(\boldsymbol{a})$, so erhöhe $\lambda$ um den Faktor 10 und gehe zu ($\star$)
- Wenn $\chi^2(\boldsymbol{a} + \delta\boldsymbol{a}) \leq \chi^2(\boldsymbol{a})$, so verringere $\lambda$ um den Faktor 10, aktualisiere $\boldsymbol{a} = \boldsymbol{a} + \delta\boldsymbol{a}$ und gehe zu ($\star$)

Nun ist noch eine geeignete Abbruchbedingung zu suchen, um nicht unnötige Iterationen in Kauf zu nehmen, die den Fehler nicht mehr wesentlich verringern. Die relative Fehlerabnahme ist ein gutes Konvergenzkriterium. Dieses bedeutet: unterschreitet die relative Fehlerdifferenz

$$\chi^2_{rel} := \frac{\chi^2(\boldsymbol{a}) - \chi^2(\boldsymbol{a} + \delta\boldsymbol{a})}{\chi^2(\boldsymbol{a})} \tag{3.21}$$

zwischen zwei Iterationsschritten einen bestimmten Wert, z.B. $\chi^2_{stop} = 0.01\%$, so wird die Approximation beendet. Der Levenberg-Marquardt Algorithmus ist in Abbildung 3.1 noch einmal veranschaulicht.

## 3.2 Erste Approximationsversuche

Nun, da für die Initialisierung mehrere Strategien vorliegen und auch der Optimierungsalgorithmus diskutiert ist, können erste Tests an einfachen $2D$-Funktionen vorgenommen werden. Es bleibt noch aus, eine geeignete Basisfunktion, unser sogenanntes Mother Wavelet, zu wählen. Wir haben die Initialisierung des Waveletnetzes auf der Basis einer Gaußpyramide diskutiert, an dieser Stelle soll die erste Ableitung der Gaußfunktion

$$\psi(\boldsymbol{x}) = -\frac{x_1}{\sigma^2_{x_1}} e^{-\frac{1}{2}\left(\frac{x_1^2}{\sigma^2_{x_1}} + \frac{x_2^2}{\sigma^2_{x_1}}\right)} \tag{3.22}$$

aufgrund ihrer allseits bekannten selektiven Eigenschaften als Mother Wavelet dienen. Die bei der Initialisierung besprochene Strategie kommt hier erstmals

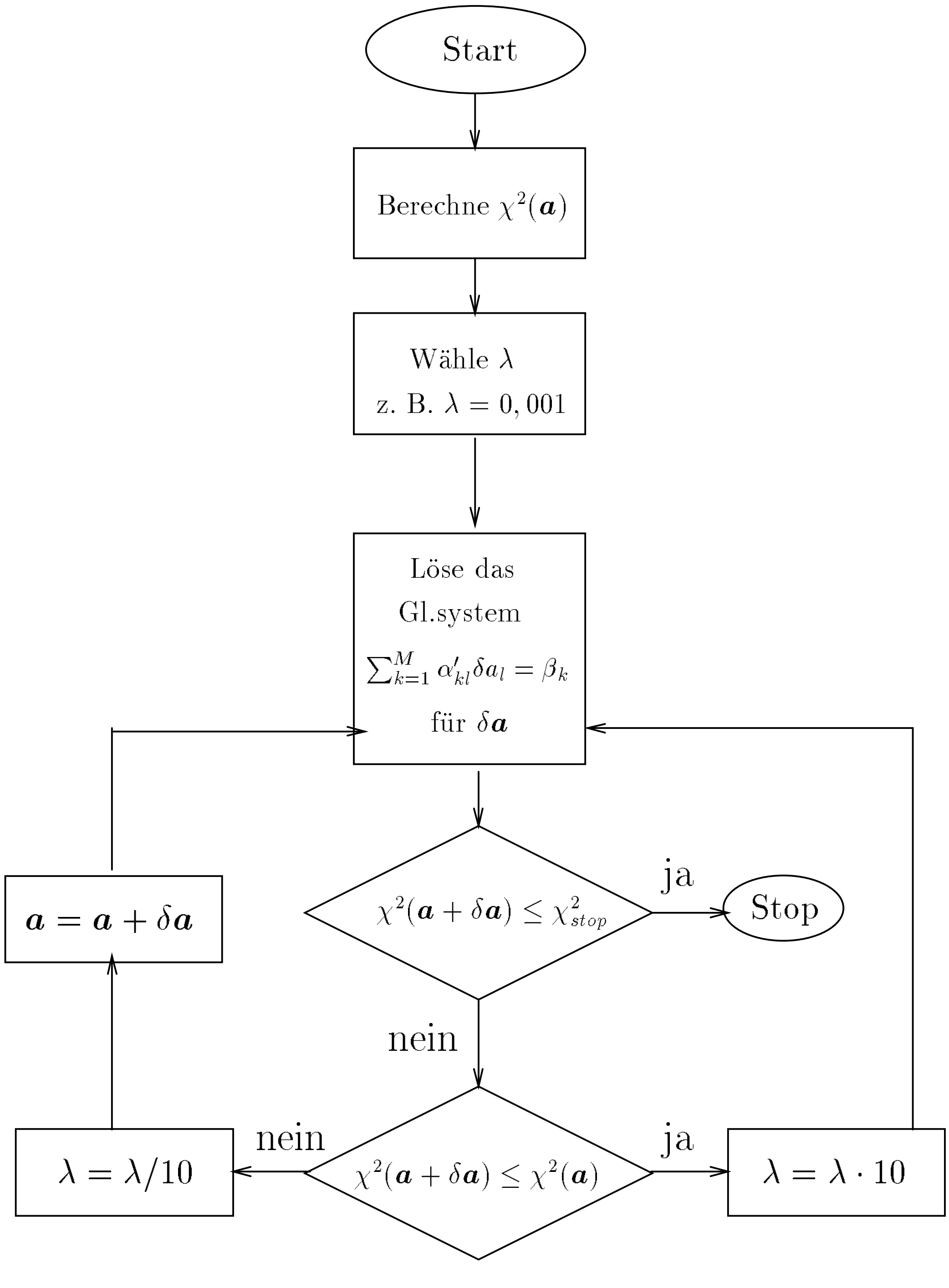



Abbildung 3.1: Die Levenberg - Marquardt - Methode.

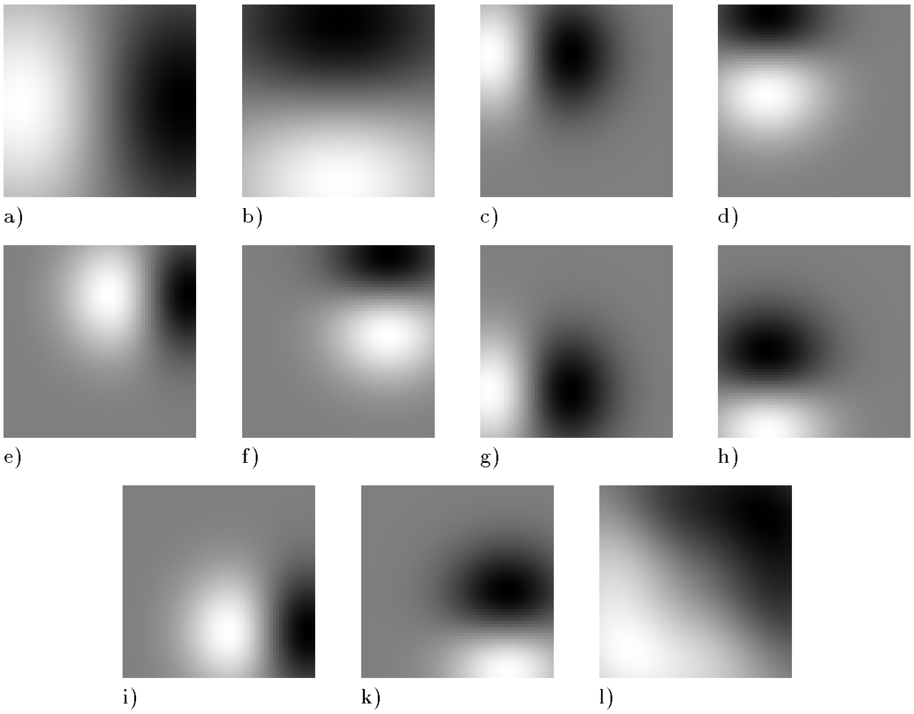
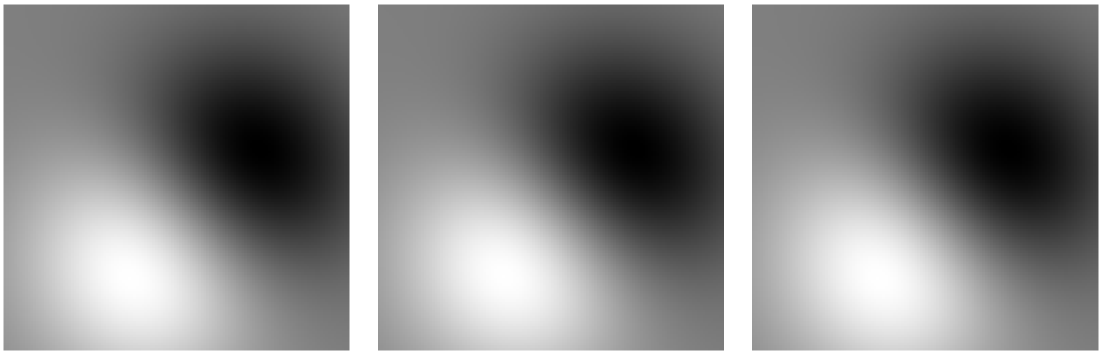



Abbildung 3.2: Die zehn Wavelets einer zweistufigen Pyramide. (a) und (b) zwei Wavelets (Orientierung 0 bzw. $\frac{\pi}{2}$) in der ersten Stufe, (c) bis (k) acht Wavelets (jeweils zwei Orientierungen an einem Ort) in der zweiten Stufe (l) die Rekonstruktion.

zur Anwendung. Durch die Tatsache, daß wir die erste Ableitung und nicht die Funktion selbst gewählt haben, ändert sich an unserer Initialisierung nichts Wesentliches, lediglich der Wert der Funktion am Bildrand entspricht nicht mehr genau der Hälfte, worauf aber nicht weiter eingegangen werden soll. Entscheiden wir uns zunächst für eine zweistufige Pyramide, so besteht diese aus zehn Wavelets, welche in Abbildung 3.2 als $64 \times 64$ Grauwertbilder abgebildet sind. Die Initialisierung wurde nach der Pyramidenmethode gewählt, also betragen die Gewichte der ersten Stufe $\omega = 64$ und die der zweiten Stufe $\omega = 16$. Die Initialisierung der Zentren und Skalen erfolgt wie im letzten Kapitel beschrieben, die Anfangswerte für die Orientierungen sind $\alpha = 0$ und $\alpha = \frac{\pi}{2}$, was bedeutet, daß am selben Ort zwei Wavelets mit einer Phasenverschiebung von $\frac{\pi}{2}$ vorhanden sind.

Eine damit einfach zu approximierende willkürliche $2D$-Funktion ist ein beliebig skaliertes und verschobenes Mother Wavelet. Wollen wir diese Funktion mit unserem zweistufigen Netz approximieren, so setzen wir unser Abbruchkriterium, die schon besprochene relative Fehlerabnahme, beispielsweise auf $\chi^2_{stop} = 5\%$ und starten die Optimierung, wobei alle Parameter frei sind. Es besteht selbstverständlich auch die Möglichkeit, bestimmte Parameter einzufrieren, wodurch aber die Eigenschaft der universellen Approximation verloren geht. Die Approxi-



a) b) c)

Abbildung 3.3: (a) Die $2D$-Funktion sowie die zwei Stufen (b) und (c) der Rekonstruktion der Pyramide.

mation konvergiert nach wenigen Iterationsschritten. Dieser erste Versuch wurde an einem $128 \times 128$ Bild durchgeführt. Die Funktion und die zwei Rekonstruktionsstufen, jeweils nach jeder Pyramidenstufe, sind in der Abbildung 3.3 zu sehen.

Der Approximationsfehler ist in diesem Beispiel so minimal, daß er hier nicht wahrzunehmen ist. Auffällig ist auch, daß die erste Stufe den Großteil der Bildinformation trägt, an dieser Stelle hätte also schon die erste Stufe der Pyramide ausgereicht, unser Mother Wavelet in der Rekonstruktion wiederzuerkennen. Theoretisch reicht ***ein*** optimal angepaßtes Wavelet aus, diese Funktion zu approximieren. Die Frage ist nun, wie dieses Waveletnetz auf ein beliebiges Bild, möglicherweise mit viel Struktur reagieren wird. Zu diesem Zweck testen wir unser zweistufiges Netz an dem altbekannten $128 \times 128$ Bild „Lena“. Wir geben also wieder alle sechs Parameter frei und brechen bei $\chi^2 = 5\%$ als relativen Fehler ab. Das Ergebnis zeigt Abbildung 3.4.

Wie an der Rekonstruktion Abbildung 3.4 (d) zu erkennen, ist der Approximationsfehler in diesem Beispiel erheblich größer, und die Rekonstruktion läßt keine „Lena“ erkennen, obgleich jedoch die Anpassung der groben Hell-Dunkel-Strukturen andeutungsweise in die richtige Richtung geht. Dies bedeutet also, daß die fehlerminimierenden Wavelets in diesem Beispiel große Skalen besitzen und deshalb für feine Strukturen mehr Wavelets benötigt werden. Ein zweistufiges Netz reicht hier nicht mehr aus. Eine weitere Erklärung könnte der niedrige Abbruchfehler sein oder lokale Minima, die die Iterationen entsprechend abbrechen lassen. Auch das Herabsetzen des Abbruchfehlers bringt keinen Fortschritt, weshalb wir weitere (feinere) Pyramidenstufen hinzunehmen müssen. Im nächsten Versuch wird eine dritte Stufe ergänzt und wiederum mit den sechs freien Parametern optimiert. Hier taucht ein weiteres schwerwiegendes Problem auf, denn der Optimierungsalgorithmus bricht in der Anpassung der letzten Stufe bei der Invertierung der Hessematrix mit einem numerischen Fehler ab. Die Matrix wird singulär. Um herauszufinden, was die Ursache hierfür ist, geben wir die Parame-

a) b) c) d)

Abbildung 3.4: (a) Das Originalbild „Lena“, sowie die Rekonstruktionen (b) nach zwei Wavelets (erste Stufe), (c) nach sechs Wavelets (zweite Stufe) und (d) nach zehn Wavelets (zweite Stufe vollständig rekonstruiert).

ter nacheinander einzeln frei. Die Ursache ist schnell erkannt, gibt man die Skalen und Zentren in beiden Dimensionen gleichzeitig frei, so rutscht der Algorithmus in Richtung eines (offenbar lokalen) Minimums, das die Wavelets aus dem Bildbereich zieht. Sie geben daher keinen Beitrag mehr zur Projektion des Bildes auf die Basis, und die Hessematrix wird singulär. Die Lösung dieses Problems kann nur darin liegen, die Wavelets am „Auswandern“ zu hindern. Man spricht in diesem Zusammenhang auch von Regularisierung der Parameter.

### 3.2.1 Regularisierung der Parameter

Wie bereits angesprochen, ist es notwendig, einige Parameter in bestimmten Grenzen zu halten. Die beste Strategie in diesem Falle ist es, an unsere Fehlerfunktion (3.12) einen Bestrafungsterm $\varphi(\boldsymbol{a})$ anzuhängen, der nur dann ins Gewicht fällt, wenn die Parameter ihre vorgegebenen Grenzen verlassen. Die Fehlerfunktion ändert sich zu

$$\tilde{\chi}^2(\boldsymbol{a}) = \frac{1}{2}\sum_{i=1}^{N}[y_i - y(x_i;\boldsymbol{a})]^2 + \varphi(\boldsymbol{a}). \tag{3.23}$$

Dies hat für die Ableitungen folgende Konsequenzen:

$$\frac{\partial\tilde{\chi}^2}{\partial a_k} = \frac{\partial\chi^2}{\partial a_k} + \frac{\partial\varphi(\boldsymbol{a})}{\partial a_k} = \tilde{\beta}_k \tag{3.24}$$

$$\frac{\partial^2\tilde{\chi}^2}{\partial a_k\partial a_l} = \frac{\partial^2\chi^2}{\partial a_k\partial a_l} + \frac{\partial^2\varphi(\boldsymbol{a})}{\partial a_k\partial a_l} = \tilde{\alpha}_{kl}. \tag{3.25}$$

Die entsprechenden Ableitungsterme verhalten sich additiv. Nun kann also eine geeignete Bestrafungsfunktion $\varphi(\boldsymbol{a})$ gefunden werden.

**1. Regularisierung der Zentren $\boldsymbol{t_1}, \boldsymbol{t_2}$**

Wollen wir verhindern, daß die Wavelets aus dem Bild wandern, so müssen sie bestraft werden, sobald sie das tun. Daher scheint es ratsam, ein Potential an den Bildrand zu legen, das ein den Bildbereich verlassendes Wavelet wieder in diesen hineinzieht. Dieses Potential sollte geeignet mit dem Abstand vom Bild zunehmen. Eine allgemeine Form eines solchen Potentials ist $A\,x^n$ mit einer Amplitude $A$ und einem Exponenten $n \geq 1$. Hierfür müssen noch geeignete Werte gefunden werden. Beschränken wir uns zunächst auf die Zentren, so könnte das Potential wie folgt aussehen:

$$\varphi(\boldsymbol{a}) = \varphi(t_1, t_2) = \varphi(t_1) + \varphi(t_2) \tag{3.26}$$

wobei wie erläutert

$$\varphi(t_i) = \begin{cases} A(t_u - t_i)^n & : \quad t_i < t_u \\ A(t_i - t_o)^n & : \quad t_i > t_o \\ 0 & : \quad t_u \leq t_i \leq t_o \end{cases} \qquad i = 1, 2. \tag{3.27}$$

Hier sind $t_u$ und $t_o$ die untere bzw. obere Grenze des Parameters $t$, die nach unseren Vorüberlegungen in etwa mit den Bildgrenzen übereinstimmen sollten. Ein erneuter Versuch, die „Lena“ mit einem dreistufigen Netz zu approximieren, zeigt die gewünschte Wirkung für die Zentren. Jedoch scheitert eine gute Anpassung daran, daß die Skalen viel zu hoch werden, offensichtlich ein Nebeneffekt der Regularisierung der Zentren. Um das Problem vollständig zu lösen, nehmen wir eine Regularisierung der restlichen Parameter vor. Eine Ausnahme bildet die Rotation, da sie periodisch in $2\,\pi$ ist.

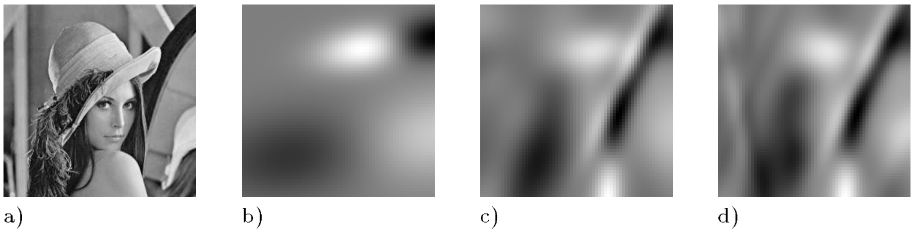



Abbildung 3.5: (a) Das Originalbild „Lena“, sowie die drei Stufen (b), (c) und (d) der Anpassung mit der Regularisierung aller Parameter.

### 2. Regularisierung der Skalen $\boldsymbol{\sigma_1, \sigma_2}$ und der Gewichte $\omega$

Wir lösen das Problem auf dieselbe Weise, indem wir den Bestrafungsterm auch von den Skalen und dem Gewicht abhängig machen. Unser Potential nimmt damit folgende Form an:

$$\varphi(\boldsymbol{a}) = \varphi(t_1, t_2, \sigma_1, \sigma_2, \omega) = \varphi(t_1) + \varphi(t_2) + \varphi(\sigma_1) + \varphi(\sigma_2) + \varphi(\omega). \tag{3.28}$$

Es ist naheliegend, auch für die Skalen und die Gewichte zwei Grenzen festzulegen, in denen diese Parameter frei sind und außerhalb derer sie bestraft werden. Wir machen denselben Ansatz wie bei den Zentren

$$\varphi(\sigma_i) = \begin{cases} A(\sigma_u - \sigma_i)^n & : \quad \sigma_i < \sigma_u \\ A(\sigma_i - \sigma_o)^n & : \quad \sigma_i > \sigma_o \\ 0 & : \quad \sigma_u \leq \sigma_i \leq \sigma_o \end{cases} \qquad i = 1, 2. \tag{3.29}$$

$\sigma_u$ und $\sigma_o$ sind die noch zu bestimmende untere bzw. obere Grenze der Skaleneinschränkung. Die Regularisierung der Gewichte verläuft analog (3.29). Um mit einem ersten Skalenbereich die Gesamtregularisierung testen zu können, übernehmen wir als Grenzen die Skalen der ersten und der $n$-ten Pyramidenstufe. Somit sind die Skalen kontinuierlich frei über drei Stufen. Die Grenzen für die Gewichte sollen an dieser Stelle auf $\omega_u = -512$ bzw. $\omega_o = 512$ gesetzt werden. Die dreistufige Pyramide wird mit Regularisierungspotential und einem Abbruchfehler von $\chi^2_{stop} = 0.1\,\%$ angepaßt. Diese Form der Regularisierung hat sich ausgezahlt, denn sowohl die Zentren als auch die Skalen bewegen sich vorwiegend innerhalb der Regularisierungsgrenzen, was den Algorithmus in unserem Sinne enden läßt. Das Ergebnis dieses Versuches zeigt Abbildung 3.5.

„vorwiegend“ soll hier andeuten, daß es den Parametern nicht unmöglich ist, ihre Regularisierungsgrenzen zu verlassen. Je nachdem wie die Fehlerlandschaft an dem entsprechenden Punkt gestaltet ist, verursacht diese Überschreitung eine Fehlerverringerung oder einen Fehleranstieg. Um Rechenzeit zu sparen, werden die folgenden Approximationen auf $64 \times 64$ Bildern gemacht. Auf die damit verbundenen Konsequenzen des Informationsverlustes soll später eingegangen werden. Wie in der Abbildung 3.5 zu erkennen, löst die dritte Stufe der Pyramide mehr Struktur auf als in unserem letzten Versuch (Abbildung 3.4), jedoch bei weitem nicht genug, was unter anderem auch an der Wahl des Mother Wavelet liegen könnte und deshalb dessen Güte im Hinblick auf die Selektivität genauer untersucht werden muß.

### 3.2.2 Das Gaborfilter

Wie im letzten Abschnitt gesehen, leistet auch eine dreistufige Pyramide nicht wesentlich mehr im Hinblick auf Originaltreue der Rekonstruktion. Sicherlich sind in der inhaltsbasierten Kodierung von Bildern andere Merkmale wichtig als originalgetreue Rekonstruktion. Letztere ist aber in jedem Falle ein Prüfstein des Verfahrens. Wie aus der Computer Vision [13] bekannt, ist die Gaborfunktion als Quadraturfilterpaar sehr gut geeignet, lokalspektrale Signalkomponenten zu extrahieren. Das Gaborfilter ist ein Bandpaß mit gaußförmigem Übertragungsverhalten. Es bildet die Basis der Short-Time-Fourier-Transformation und minimiert dabei, wie die Gaußfunktion als ihre Einhüllende, das Unschärfeprodukt von Orts- und Frequenzbandbreite. Als Gaborfunktion bezeichnet man eine gaußmodulierte komplexharmonische Funktion

$$h(x) = e^{-\frac{x^2}{\sigma^2}} \cdot e^{-2\pi i u_0 x}. \tag{3.30}$$

Da man in der digitalen Signalverarbeitung die Behandlung komplexer Signale vermeiden will, erzeugt man ein Quadraturfilterpaar $h^e$ (even) und $h^o$ (odd) reeller Filter. In unserem Falle handelt es sich um die $2D$-Gaborfunktion

$$h_{\boldsymbol{C};\boldsymbol{u}_0}(\boldsymbol{x}) = e^{-\boldsymbol{x}^T \boldsymbol{C}^{-2} \boldsymbol{x}} \cdot e^{-2\pi i \boldsymbol{u}_0^T \boldsymbol{x}} \tag{3.31}$$

als Quadraturfilterpaar:

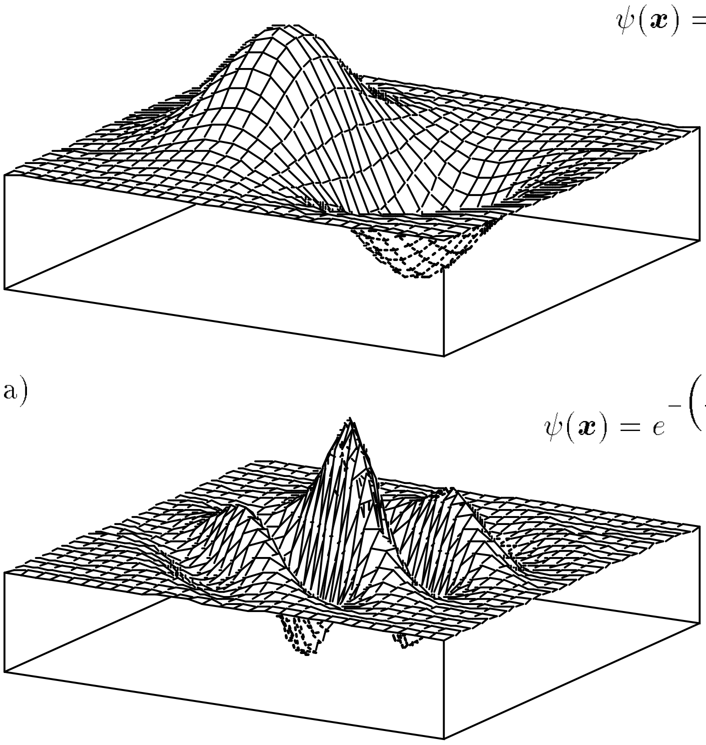
$$h^e_{\boldsymbol{C};\boldsymbol{u}_0}(\boldsymbol{x}) = e^{-\boldsymbol{x}^T \boldsymbol{C}^{-2} \boldsymbol{x}} \cdot \cos(-2\pi \boldsymbol{u}_0^T \boldsymbol{x}) \tag{3.32}$$

$$h^o_{\boldsymbol{C};\boldsymbol{u}_0}(\boldsymbol{x}) = e^{-\boldsymbol{x}^T \boldsymbol{C}^{-2} \boldsymbol{x}} \cdot \sin(-2\pi \boldsymbol{u}_0^T \boldsymbol{x}). \tag{3.33}$$

Durch Drehung mit der Rotationsmatrix $R_\alpha$ entsteht eine Gaborfunktion, die zur orientierungsselektiven spektralen Analyse von Grauwertverläufen geeignet ist.

$$\psi(\boldsymbol{x}) = -x\, e^{-\left(\frac{x^2}{2}+\frac{y^2}{2}\right)}$$

$$\psi(\boldsymbol{x}) = e^{-\left(\frac{x^2}{2}+\frac{y^2}{2}\right)} \cdot \cos(-\pi(x+y))$$



a)

b)

Abbildung 3.6: Die beiden Basisfunktionen im Vergleich, (a) die Ableitung der Gaußfunktion in $x$-Richtung und (b) das ungerade Gaborfilter.

$$h_{\boldsymbol{C};\boldsymbol{u}_0;\alpha}(\boldsymbol{x}) = h_{\boldsymbol{C};\boldsymbol{u}_0}(R_\alpha \boldsymbol{x}) \tag{3.34}$$

Figur 3.6 zeigt die erste Ableitung der Gaußfunktion und das ungerade Gaborfilter im Vergleich.

Die Vorzüge des Gaborfilters hinsichtlich Selektion gegenüber der ersten Gaußableitung werden deutlich, wenn man das Bild auf die entsprechenden Netze projiziert. In Abbildung 3.7 ist die „Lena“ auf drei-, vier-, und fünfstufige Gauß- bzw. Gabornetze projiziert worden. Deutlich sichtbar ist die höhere Qualität des Gabors bei der Kantendetektion.

Für einen direkten Vergleich ist eine Optimierung mit beiden Netzen notwendig. Um die Qualität der Rekonstruktionen andererseits weiter zu verbessern, nehmen wir eine weitere Stufe in der Pyramide hinzu und lassen, da die feinen Strukturen bisher nicht extrahiert wurden, den Skalen einen größeren Spielraum. Eine gute Strategie scheint hierbei die Bereitstellung weiterer „virtueller“ Pyramidenstufen. Dies wird dadurch erreicht, daß die Grenzen der Skalenregularisierung als Vielfache der Grundskala in der ersten Stufe gesetzt werden. Mit Regularisierungspotential werden die Gaußpyramide und die Gaborpyramide mit vier Stufen

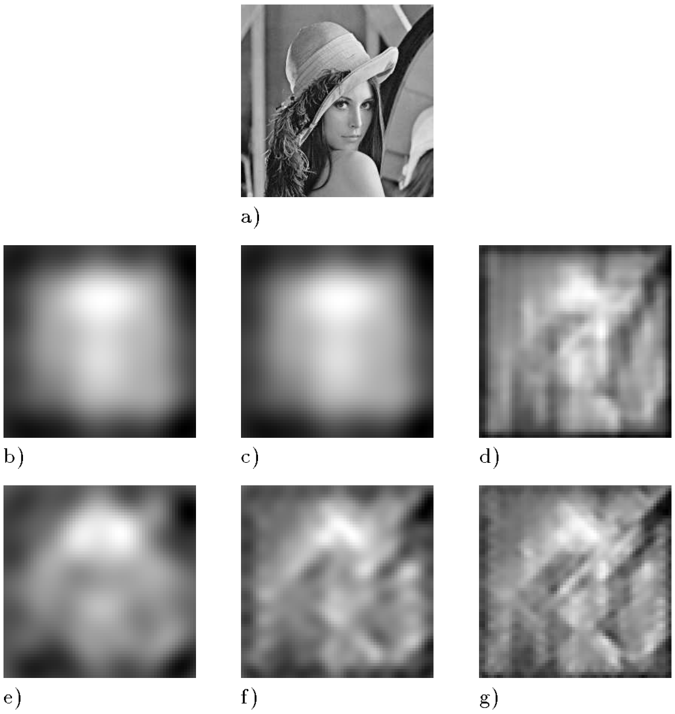



Abbildung 3.7: (a) Das Original sowie seine Projektion auf ein Netz von Gaußableitungen (b) dreistufig (c) vierstufig (d) fünfstufig und entsprechend seine Projektion (e), (f) und (g) auf ein Netz von ungeraden Gaborfiltern.

angepaßt. Der Abbruchfehler beträgt jeweils $\chi^2_{stop} = 0.1\%$. Das recht eindeutige Ergebnis zeigt Abbildung 3.8.

Bei Betrachtung der Abbildung 3.8 stellt man mit einem Blick fest, daß die Rekonstruktionen der Gaborpyramide jeweils etwa um eine Stufe besser sind, was unsere vorherigen Überlegungen bezüglich der Orientierungsselektivität bestätigt. Nun haben wir aus Gründen der Komplexität unsere Optimierung an unterabgetasteten Bildern vorgenommen, was an sich eine sehr gute Möglichkeit darstellt, erheblich Rechenzeit zu sparen. Um ein komprimiertes Bild in der ursprünglichen Größe zu erhalten, liegt es nun nahe, die entsprechenden Waveletparameter, also Zentren und Skalen, zu vergrößern. Dies ist in Abbildung 3.9 mit unserem bisher besten Ergebnis, der vierstufigen Gaborpyramide, geschehen.

Mit zunehmender Bildgröße nimmt die Rekonstruktionsgüte im Bezug auf die Originaltreue ab. Dieser Effekt hat seine Ursache natürlich in der Unterabtastung, die wir vorgenommen haben, um die Komplexität der ersten Versuche zu senken. Es ist aber auch festzustellen, daß ein Bild durch ein Waveletnetz beliebig skaliert werden kann, ohne die üblichen Fehler beim Vergrößern der Pixel zu erhalten. Ein Waveletnetz mit vier Stufen umfaßt 170 Wavelets. Bleiben wir bei unserer Strategie und lassen hinreichend kleine Skalen zu, so sollten mit dieser Netzgröße bessere Rekonstruktionen erreichbar sein als in den vorangegangenen

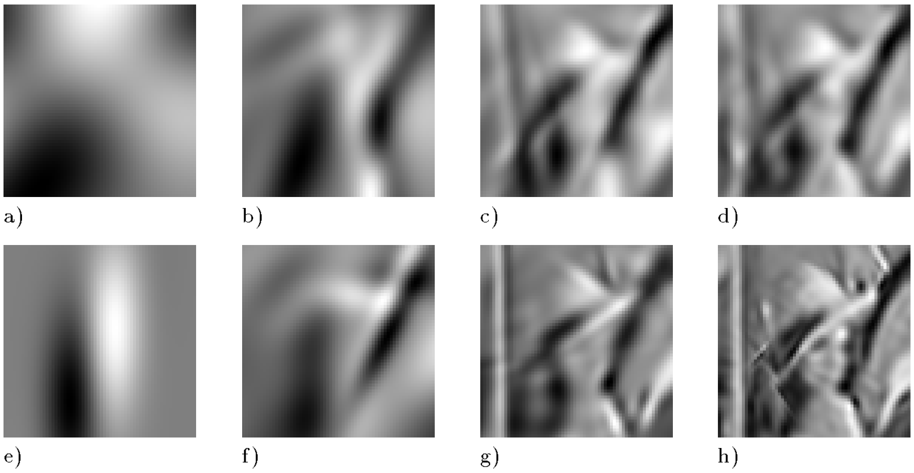



Abbildung 3.8: Die vier Stufen der Rekonstruktion des Netzes von Gaußableitungen (a) bis (d), sowie für das Netz von ungeraden Gaborfiltern (e) bis (h).

Versuchen. Jedoch nimmt die Komplexität mit der Bildgröße derart zu, daß der Aufwand schnell den Nutzen minimiert. Deshalb sollen die nächsten Überlegungen der Methode der Optimierung gelten. Derzeit werden die Stufen der Pyramide, beginnend mit der größten Skala, nacheinander (progressiv) optimiert. Die Anzahl der Wavelets vervierfacht sich dabei von Stufe zu Stufe, was genauso auf die Größe der Matrizen zutrifft, mit denen der Algorithmus arbeitet. Dies führt zu einer Optimierungsdauer von mehreren Stunden bei größeren Bildern und hat uns dazu veranlaßt, uns auf $64 \times 64$ Bilder zu beschränken, was selbstverständlich unakzeptabel ist. Deshalb gilt unser Interesse zunächst der progressiven Optimierung.

### 3.2.3 Die progressive Optimierung

Die Motivation für den Schritt der Untersuchung der Optimierung war die Senkung der Komplexität. Die Beobachtung der bisherigen Methode zeigt, daß die Optimierung der ersten Stufe (ein bzw. zwei Wavelets, je nach Initialisierung) wenige Sekunden dauert, die der fünften Stufe (256 Wavelets) einige Stunden. Durch diese Vorgehensweise wird aber eine Abnahme des Gesamtfehlers sichergestellt, die dadurch zustande kommt, daß jedes Wavelet einer Stufe lokal in seinem Skalenbereich den Fehler minimiert. Überlegt man sich aber, daß die Änderung der Waveletparameter nur zu einer Änderung des Gesamtfehlers im Bereich des jeweiligen Wirkungsbereiches des Wavelets führt, so ist die Wirkung der Optimierung eines Wavelets ohnehin lokal, was im folgenden Versuch überprüft werden soll. Wir optimieren die Wavelets nacheinander zeilenweise in jeder Stufe, wodurch die Grob-Fein-Strategie gewahrt bleibt.

Abbildung 3.10 zeigt die Rekonstruktion der fünf Stufen einer $128 \times 128$ Gaborpyramide, optimiert mit Regularisierungspotential und einem Abbruchfehler von $\chi^2_{stop} = 0.0001\%$. Die Optimierungszeit von $17,5$ min bestätigt unsere Überlegungen. Offensichtlich hat die Verringerung des Abbruchfehlers die Approximationsgüte verglichen mit Abbildung 3.9 erheblich verbessert. Die Methode der progressiven Optimierung erweist sich also als sehr brauchbar und wird im folgenden als Standard verwendet.

Zuerst aber soll noch eine andere Idee verfolgt werden, die Idee der attentiven Initialisierung, bei der nach einem geeigneten Attentionsmerkmal entsprechend der Anwendung das Waveletnetz so initialisiert wird, daß ein erheblicher Optimierungsanteil gespart werden kann. Aus diesem Grund ist die Attention Gegenstand des nächsten Kapitels.


Abbildung 3.9: Verschiedene Bildgrößen und die entsprechend skalierten Rekonstruktionen der $64 \times 64$ Pyramide. Hier zeigt sich, daß ein Bild mit einem Waveletnetz ohne die üblichen „Überabtastfehler“ beim Skalieren einer Pixelrepräsentation beliebig vergrößert werden kann.

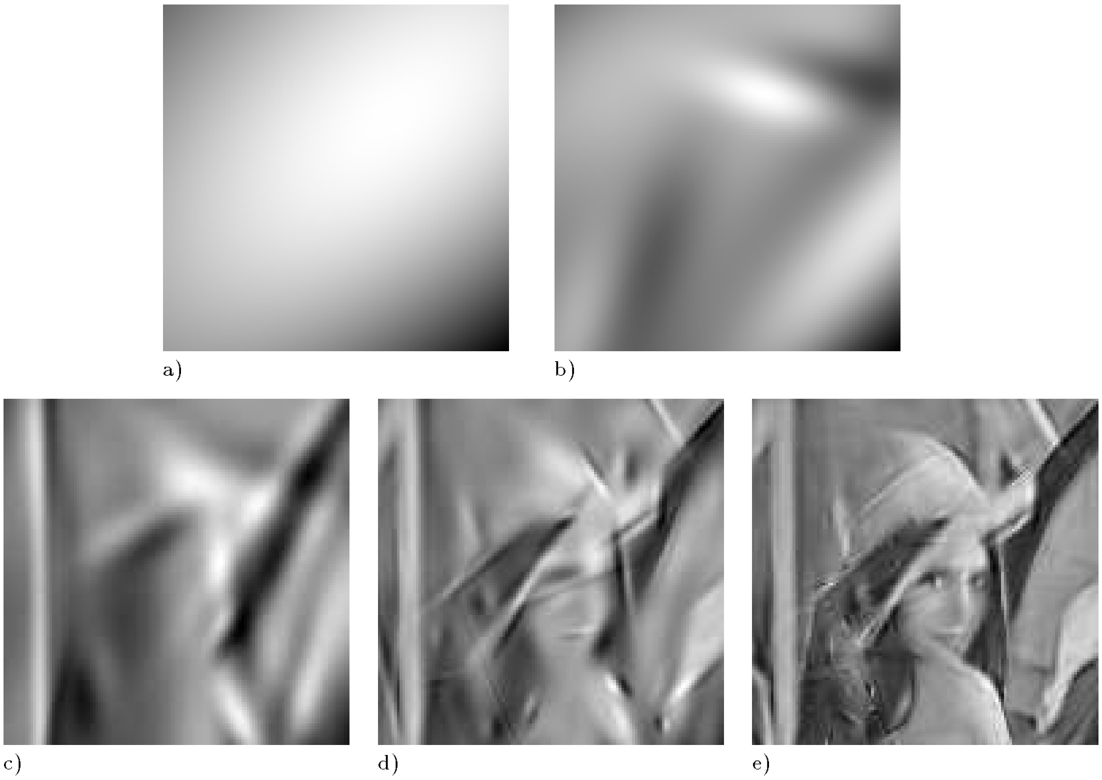


Abbildung 3.10: Die fünf Stufen eines mit der Methode der progressiven Optimierung angepaßten $128 \times 128$ Gabornetzes. In den Rekonstruktionen der letzten beiden Stufen (d) und (e) wird die lokale Wirkung der Methode im Bereich der Augen und des Federbusches deutlich.

# Kapitel 4

# Attentive Initialisierung

In diesem Abschnitt sollen einige Möglichkeiten der attentiven Initialisierung vorgestellt und getestet werden, mit deren Hilfe eine geeignete Wahl der Waveletparameter erreicht wird, um sowohl eine schnellere Konvergenz des Optimierungsverfahrens als auch eine aufgabenbezogene Approximationsgüte zu erreichen. Dabei soll auch das Ziel der progressiven Bildkompression, die geeignete Rekonstruktion, nicht außer acht gelassen werden. Im Folgenden werden zwei gänzlich verschiedene Ansätze verfolgt. Zum einen ein passiver Ansatz, bei dem ein Operator, in unserem Falle der Gaußsche Krümmungsoperator, eingesetzt wird, um die Grauwertkrümmung für jeden Bildpunkt zu berechnen. Auf diese Weise sollen die Attentionspunkte geeignet gewichtet werden. Zum anderen ein aktiver Ansatz, bei dem der Aufgabensteller (Beobachter) selbst Regionen der Attention bestimmt und damit auch die Netzstruktur festlegt.

## 4.1 Das Gaußsche Krümmungsmaß

Betrachtet man ein Bild, das ein $2D$-Signal darstellt, als Fläche im $3D$-Raum, so stehen alle Werkzeuge der Differentialgeometrie zur Beschreibung und Klassifikation der wesentlichen Oberflächentypen zur Verfügung. Barth, Caelli und Zetsche nutzten in [1] die Repräsentation von Bildern als Flächen und ihre formale Charakterisierung durch Differentialoperatoren. Die verschiedenen Flächentypen (planar, parabolisch, elliptisch ...) entsprechen bei ihnen verschiedenen Graden von Redundanz. Wir wollen uns die Differentialgeometrie in ähnlicher Weise zunutze machen und mit Hilfe der Differentialoperatoren die Grauwertflächen nach ihrem Attentionspotential charakterisieren. Das Ziel ist eine Multiskalenanalyse zur Gewinnung geeigneter Initialwerte für die Waveletskalen und -zentren, wobei wir von der Heuristik „Hohe Krümmung $\Longrightarrow$ Hohes Attentionspotential“ aus-gehen. Zunächst sollen jedoch einige fundamentale Aspekte der geometrischen Signalrepräsentation betrachtet werden.

### 4.1.1 Flächen in der Differentialgeometrie

In der klassischen Differentialgeometrie ist eine Fläche definiert in der vektoriellen Parameterform.

$$\boldsymbol{S}(u,v) = \{(X(u,v), Y(u,v), Z(u,v))\} \tag{4.1}$$

Hierbei sind $(u,v)$ die internen Parameter der Fläche und $X, Y$ und $Z$ reelle Funktionen von $(u,v)$. Die Fläche wird also gebildet durch eine Menge von Punkten im $\mathbb{R}^3$. Die sie beschreibenden fundamentalen Differentialoperatoren sind

$$\{\boldsymbol{S}_u, \boldsymbol{S}_v, \boldsymbol{S}_{uv}, \boldsymbol{S}_{uu}, \boldsymbol{S}_{vv}\}. \tag{4.2}$$

Die Größe $\boldsymbol{S}_z$ entspricht dabei der partiellen Differentiation von $\boldsymbol{S}$ nach $z$. Wir favorisieren an dieser Stelle die explizite Form, die sogenannten *Monge patches*, da es sich bei unseren Bildern um die Oberflächen von Grauwertgebirgen, also Grauwerten $f(x,y)$ über der Projektion $(x,y)$ handelt:

$$\boldsymbol{f}(x,y) = \boldsymbol{f}(\boldsymbol{x}) = \{x, y, f(x,y))\}. \tag{4.3}$$

Die entsprechenden Differentialoperatoren lauten

$$\{\boldsymbol{f}_x, \boldsymbol{f}_y, \boldsymbol{f}_{xy}, \boldsymbol{f}_{xx}, \boldsymbol{f}_{yy}\}. \tag{4.4}$$

Die Eigenschaften der Fläche werden durch die nun folgenden Fundamentalformen beschrieben.

### 4.1.2 Von der Fundamentalform zum Krümmungsmaß

Die schon genannten Differentialoperatoren bestimmen zwei Matrizen, die wiederum die Koeffizienten der ersten und zweiten Fundamentalform liefern. Die **erste Fundamentalform $\Phi_1$**

$$(d\boldsymbol{f})^2 = d\boldsymbol{x}^T\,\boldsymbol{G}\,d\boldsymbol{x} = \begin{pmatrix} dx \\ dy \end{pmatrix}^T \begin{pmatrix} 1+{f_x}^2 & f_x f_y \\ f_x f_y & 1+{f_y}^2 \end{pmatrix} \begin{pmatrix} dx \\ dy \end{pmatrix}$$

beschreibt die metrischen Eigenschaften einer Fläche. Die **zweite Fundamentalform $\Phi_2$**

$$(d\boldsymbol{f})^2 = d\boldsymbol{x}^T\,\boldsymbol{B}\,d\boldsymbol{x} = \frac{1}{\sqrt{1+{f_x}^2+{f_y}^2}}\begin{pmatrix} dx \\ dy \end{pmatrix}^T \begin{pmatrix} f_{xx} & f_x f_y \\ f_x f_y & f_{yy} \end{pmatrix}\begin{pmatrix} dx \\ dy \end{pmatrix}$$

beschreibt die Krümmung einer Fläche. Aus diesen Fundamentalformen definieren sich die folgenden Krümmungsmaße:

**Normalkrümmung $\kappa$**

$$\kappa = \frac{\Phi_2}{\Phi_1} = \frac{d\boldsymbol{x}^T\,\boldsymbol{B}\,d\boldsymbol{x}}{d\boldsymbol{x}^T\,\boldsymbol{G}\,d\boldsymbol{x}}$$

**Hauptkrümmungen $\kappa_{min}$, $\kappa_{max}$**

$$\frac{\partial\kappa}{\partial(d\boldsymbol{x})} = 0 \Longrightarrow (\kappa_{min},\,\kappa_{max}) = EW\,\boldsymbol{B}\boldsymbol{G}^{-1}$$

**Mittlere Krümmung H**

$$H = \frac{\kappa_{max}+\kappa_{min}}{2} = \frac{1}{2}\,\mathrm{trace}\,\boldsymbol{B}\boldsymbol{G}^{-1}$$

Monge patches:

$$H = \frac{f_{xx} + f_{xx}{f_y}^2 + f_{yy} + f_{yy}{f_x}^2 - 2f_{xy}f_xf_y}{(1+{f_x}^2+{f_y}^2)^{\frac{3}{2}}}$$

**Gaußkrümmung K**

$$K = \kappa_{min}\kappa_{max} = \det\,\boldsymbol{B}\boldsymbol{G}^{-1}$$

Monge patches:

$$K = \frac{f_{xx}f_{yy} - f_{xy}}{(1+{f_x}^2+{f_y}^2)^2}$$

Die Gaußkrümmung des Grauwertgebirges soll nun im weiteren die Grundlage für die Berechnung der Attentionspunkte eines Bildes durch eine Multiskalenanalyse bilden. Betrachtet man die Berechnung der Gaußkrümmung über die Monge patches, so liegt eine Approximation der partiellen Grauwertableitungen durch Binomialfilterantworten nahe. Bei der Multiskalenanalyse wird also mit Binomialmasken unterschiedlicher Skalen für jeden Bildpunkt die Gaußkrümmung berechnet und über eine Kreisscheibe der entsprechenden Skala das lokale Maximum gebildet. Die lokalen Maxima entsprechen dann nach unserer Annahme den Attentionspunkten. An diesen Stellen werden Wavelets mit der entsprechenden Skala (der Binomialmasken) initialisiert. Die Gewichte werden anschließend über die Projektion ermittelt und die Initialrotationen bleiben, wie gehabt, zufällig. Damit ist eine vollständige attentive Initialisierung erreicht.

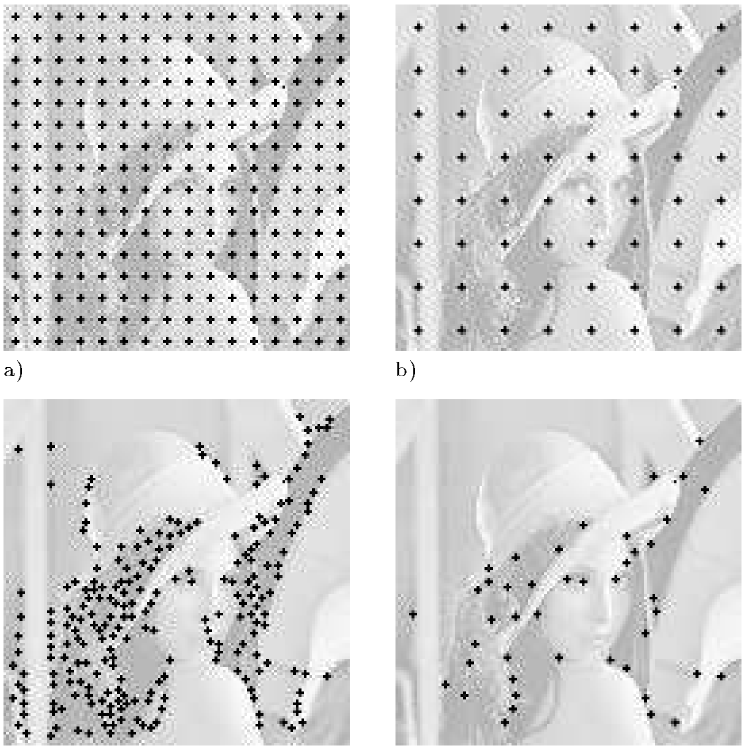
### 4.1.3 Reguläre versus irreguläre Initialisierung

Nachdem eine erste Form der attentiven Initialisierung gefunden wurde, soll diese am Bild „Lena" getestet und die Wirkung im Vergleich zur bisherigen Initialisierung betrachtet werden. Damit ein Vergleich mit dem fünfstufigen Netz aus Abbildung 3.10 gemacht werden kann, ist etwas Heuristik notwendig. Wir berechnen die lokale Krümmung des Bildes auf fünf Skalen (Stufen). Dabei wählen wir als gröbste Skala 32 und halbieren diese in jeder Stufe $(32, 16, 8, 4, 2)$, wie bei der Pyramide auch. Wir betrachten hier die Skala in Einheiten von Pixeln. Eine Form der Umrechnung wird in [13] (S. 85) gegeben. Eine Binomialmaske der Ordnung $n$ wird entweder direkt mit Hilfe der Binomialkoeffizienten gebildet oder aber durch $n$-fache lineare Autofaltung des Binomialfilters erster Ordnung (Boxfilter) und approximiert eine Gaußkurve der Varianz

$$\sigma_k^2 = \frac{(k-2)\,k^2}{12}, \qquad\qquad k = n+1, \tag{4.5}$$

wobei $k$ die Maskengröße in Pixel ist. Da Binomialmasken immer ungerade Kantenlängen haben, weil das Symmetriezentrum auf ein Maskenelement fallen muß, wird die Skala jeweils um eins erhöht $(33, 17, 9, 5, 3)$. Somit erreichen wir mit drei Pixel die kleinste mögliche Skala einer Binomialmaske und haben den Skalenbereich, der durch diese drei Pixel nach unten begrenzt ist, mit fünf Stufen abgedeckt. Um der angestrebten Strategie „Rekonstruiere die wichtigsten Attentionspunkte zuerst!" Rechnung zu tragen, werden die Attentionspunkte in jeder Stufe nach der Krümmung sortiert und nur die ersten $n$ Prozent in das Netz integriert. So hat man die Möglichkeit, die Waveletzahl in jeder Stufe gleichermaßen zu verringern.

Abbildung 4.1 zeigt jeweils zwei Stufen der Initialisierung durch ein reguläres Abtastgitter und die eben beschriebene Variante der irregulären Initialisierung



a) b) c d

Abbildung 4.1: Die Gegenüberstellung der Initialisierung mit einem regulären pyramidenartigen Abtastgitter (a) und (b) und mit einem irregulären Abtastgitter wie es bei der Initialisierung über das Gaußsche Krümmungsmaß der Fall ist (c) und (d). Zur Veranschaulichung wurde hier auf jeweils zwei verschiedenen Skalen initialisiert.

über die Gaußsche Krümmung, wobei $n$ auf 50% gesetzt wurde. Im folgenden wird ein fünfstufiges Netz mit gleicher Skaleninitialisierung und $n = 60\%$ mit einem Regularisierungspotential und einem Abbruchfehler von $\chi^2_{stop} = 0.0001\%$ optimiert.

Die Abbildungen 4.2 und 4.3 dokumentieren jeweils die Rekonstruktionen zu willkürlich gewählten Rekonstruktionsschritten der beiden Initialisierungsmethoden. Es wird sowohl während der Rekonstruktion – im Hinblick auf die progressive Übertragung – als auch im Endergebnis die Bedeutung einer geeigneten attentiven Initialisierung sichtbar. Während bei der Pyramide die Optimierung und auch die Rekonstruktion das Abtastgitter regulär zeilenweise durchläuft, geschieht dies im anderen Falle in jeder Stufe je nach Attentionspotential des Wavelets, welches ihm bei der Initialisierung in Form der Krümmung zugeordnet wird. Dieser Unterschied hat – bei gleichem $SNR$ (Signal-Rausch-Abstand) – vor allem den Effekt, daß bei der Pyramide nach Rekonstruktion etwa der Hälfte der Wavelets der Bildhintergrund Gestalt annimmt, während die Detailinformation – etwa das Gesicht – verborgen bleibt. Bei etwa gleichem Rekonstruktionsschritt des zweiten Netzes sind die entsprechenden Details bereits rekonstruiert, was im Sinne der progressi-

$n = 3$ $n = 15$ $n = 25$

$n = 30$ $n = 50$ $n = 70$

$n = 90$ $n = 150$ $n = 180$

$n = 200$ $n = 250$ $n = 300$

$n = 340$ $n = 390$ $n = 419$

Abbildung 4.2: Schrittweise Rekonstruktion des fünfstufigen Netzes bei attentiver Initialisierung über das Krümmungsmaß. $n$ gibt die Anzahl der rekonstruierten Wavelets an.

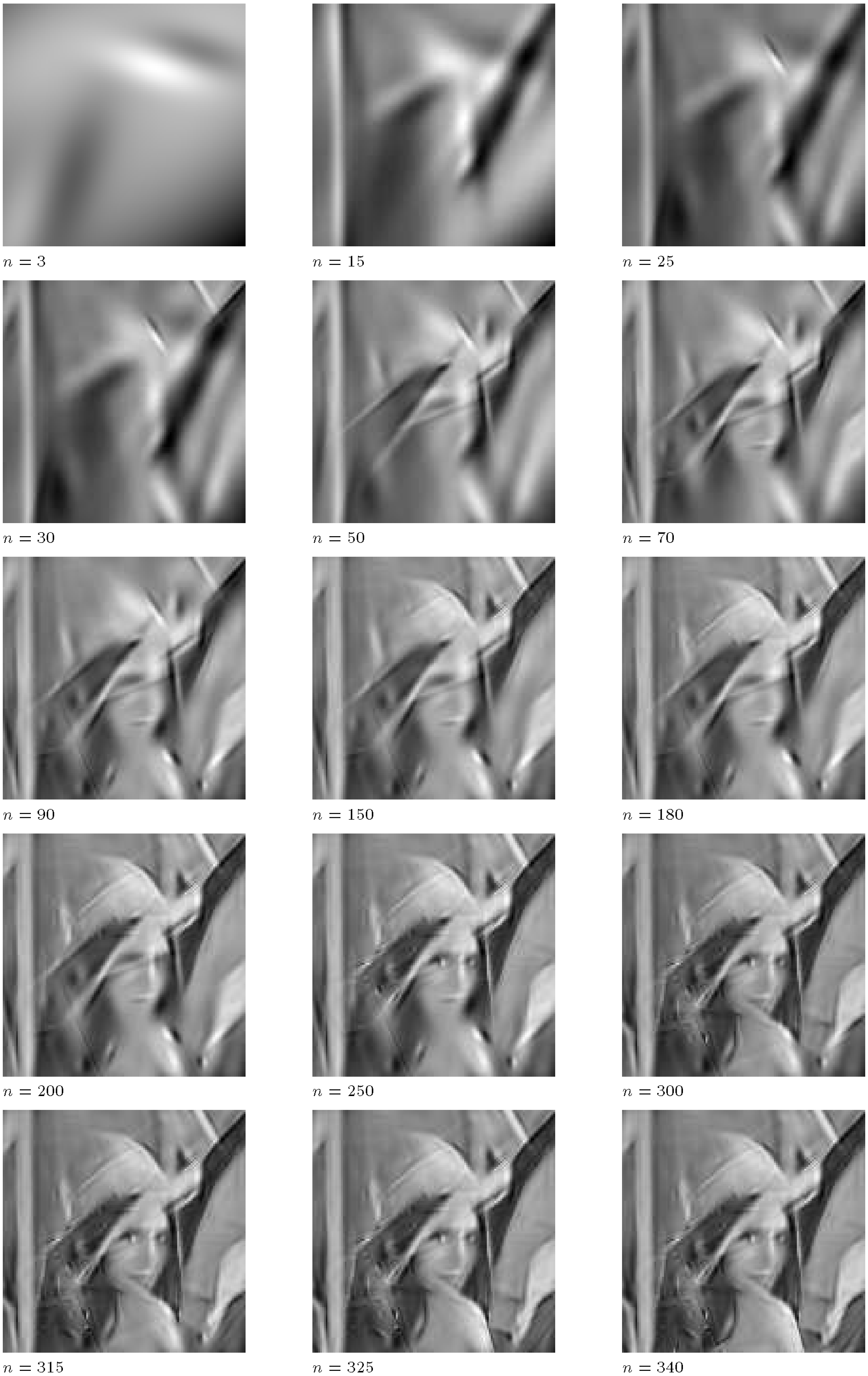



Abbildung 4.3: Schrittweise Rekonstruktion der fünfstufigen Pyramide bei regulärer Initialisierung. $n$ gibt die Anzahl der rekonstruierten Wavelets an.

ven Übertragung ist. Deshalb soll uns die Detailinformation bzw. ihre Codierung im nächsten Abschnitt beschäftigen, in dem es um eine weitere Methode der Zuordnung von Attentionspotential an bestimmte Wavelets oder Waveletgruppen geht.

## 4.2 Die Selektion von Objekten

Wie im vorigen Abschnitt gesehen, hat bestimmte Detailinformation in der aufgabenbezogenen Codierung besondere Bedeutung. Die Aufgabenstellung gibt dabei vor, welche Details oder Objekte für das Ziel der Aufgabe von Bedeutung sind. In einer Bilddatenbank von Porträts können z.B. ganze Gesichter als Objekte aufgefaßt werden oder auch nur Gesichtsdetails wie Augen oder Nasen. Ein allgemeines Modell des Objektmatchings wie z.B. das *Colour Orientation Indexing* bei Farbbildern, das in [7] beschrieben wird, wäre hier sehr hilfreich. Ein sehr spezieller, aber auch sehr einfacher Ansatz, ist durch die zweckgebundene Auswahl der bevorzugt zu codierenden Objekte durch den Aufgabensteller (Betrachter) gegeben. Nun muß zunächst erst einmal bestimmt werden, was ein Objekt ist und wie es in der Struktur des Waveletnetzes repräsentiert werden kann. An dieser Stelle liegt es nahe, die Pyramidenstruktur in der jetzigen Form aufzuheben. Motiviert wurde diese durch das Konzept der Wavelettransformation und, im Hinblick auf die Rekonstruktion, durch die Strategie, eine Stufe nach der anderen von grober zu feiner Skala zu optimieren. Da wir im folgenden lokale Objekte im Bild betrachten wollen, ist es sinnvoll, dem Netz eine innere Objektstruktur aufzuprägen, die an der Grundannahme eines Waveletnetzes (2.31) nichts ändert, wohl aber die Aufgabenstellung, unter der codiert werden soll, widerspiegelt. Es ist also der Betrachter gehalten, die Objekte auszuwählen, indem er deren Ort und Größe vorgibt. Für erste Versuche wird auf das Objekt anschließend eine Pyramide der entsprechenden Objektgröße aufgesetzt, wodurch lokal das Konzept der Wavelettransformation beibehalten wird. Die Abbildung 4.4 zeigt die Optimierungsergebnisse von Waveletnetzen mit zwei bzw. drei ausgewählten Objekten. Bei der Optimierung wurden alle Parameter frei gelassen.

Bei Betrachtung des rechten Bildes von Abbildung 4.4 fällt auf, daß die sehr gute Approximation der Objekte zu Lasten der Objektumgebung geht. Die Wavelets laufen aus den Objekten, um den Fehler im Bildhintergrund zu verringern, was nicht ungewöhnlich ist, denn der Optimierungsalgorithmus berechnet in jedem Schritt den Gesamtfehler der Rekonstruktion. Um eine lokale Anpassung der Wavelets innerhalb der Objekte zu gewährleisten, ist es notwendig, die Regularisierung ihrer Parameter an die Objektgrenzen anzupassen.

a) b)

Abbildung 4.4: Zwei Beispiele für die Objektinitialisierung der Augen (a) sowie von Augen und Mund (b).

### 4.2.1 Objektinitialisierung mit Pyramiden

Die bisherige Netzstruktur (2.31) ist dadurch gekennzeichnet, daß die Regularisierung für alle Wavelets dieselbe ist, gleich welchem Objekt sie angehören, ob dem Hintergrund oder einem anderen Objekt. Um eine kompakte Darstellung unseres Netzes zu erhalten, fassen wir nun den Hintergrund selbst als eigenständiges Objekt auf und ordnen die Wavelets eines Objektes zu einer Struktur zusammen, die der bisherigen Netzstruktur (2.31) entspricht:

$$g_j(\boldsymbol{x}) = \sum_{i=1}^{N} \omega_{ij}\, \psi[S_{ij} R_{ij}(\boldsymbol{x} - \boldsymbol{t}_{ij})]. \tag{4.6}$$

Die Superposition dieser $M$ Objektnetze ergibt das neue Netz

$$G(\boldsymbol{x}) = \sum_{j=1}^{M} g_j(\boldsymbol{x}) + \bar{g}. \tag{4.7}$$

Es ist somit eine übergeordnete Struktur geschaffen, die es erlaubt, jedes Wavelet lokal über dem zugehörigen Objekt, also einer bestimmten Region im Bild, zu optimieren. Auch die Fehlerberechnung findet nur in dieser bestimmten Region statt, so daß der Optimierungsprozeß ein lokaler Prozeß wird, der die Optimierungszeit noch einmal erheblich senken kann, je nach der Größe der Objekte. Damit allerdings die Anpassung der Objekte nicht auf Kosten ihrer Umgebungen geht – wie beispielsweise in Abbildung 4.4 rechts – beziehen wir die Umgebung der Objekte mit in die Fehlerberechnung ein. Auch hier kommt wieder etwas Heuristik ins Spiel, wenn man versucht die Umgebung eines Objektes zu bestimmen. Wir wollen an dieser Stelle die Umgebung definieren als einen Rahmen um das Objekt, der jeweils der halben Objektbreite entspricht. Auf diesem, um den

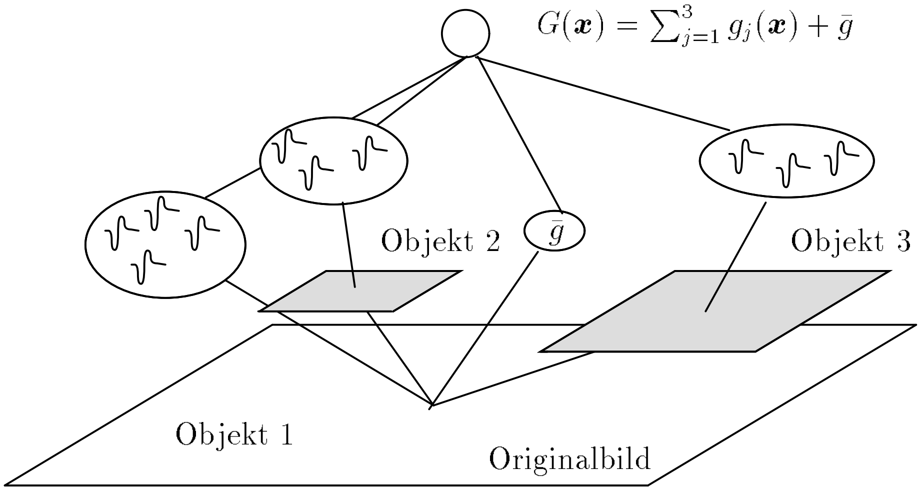



Abbildung 4.5: Die übergeordnete Netzstruktur entsprechend (4.7). Die rechteckigen Regionen entsprechen den vom Beobachter gemäß einer Aufgabenstellung ausgeschnittenen Bildbereichen, die als Definitionsbereich der Objekte eine Untermenge des Gesamtbildes darstellen. Objekt 1 entspricht in diesem Falle dem Hintergrund.

Rahmen vergrößerten, Objekt wird in jedem Iterationsschritt der lokale Fehler berechnet. Die so neu definierte Struktur unseres Waveletnetzes ist in Abbildung 4.5 noch einmal veranschaulicht. Abbildung 4.6 zeigt in einem ersten Versuch die Initialisierung und Optimierung der Zentren der Wavelets sowie die Rekonstruktion von zwei Objekten.

Zur Reduktion der Waveletzahl wurden die Wavelets der Objektpyramiden außerhalb der Objekte, die der Beobachter elliptisch ausschneidet, abgeschnitten. Innerhalb der Objekte wird wieder mit einem regulären Abtastgitter initialisiert. Zur weiteren Verfeinerung der lokalen Optimierung soll noch ein ganz anderer Ansatz für das Objektmodell verfolgt werden. Es handelt sich hierbei um die Modellierung eines Objektes durch eine Struktur, die eher biologisch motiviert ist, die *log-polare Initialisierung*.

### 4.2.2 Die log-polare Initialisierung

Die Motivation für die Einführung der log-polaren Initialisierung soll mit Bolduc und Levine [2] diskutiert werden. Das biologische visuelle System von Primaten stellt ein weites Gesichtsfeld zur Verfügung, wobei im Zentrum, im Punkt des schärfsten Sehens – der Fovea – eine viel höhere Auflösung vorhanden ist als in der Peripherie. Aus Sicht der Computer Vision bedeutet dies in der Peripherie eine Unterabtastung im Vergleich zur Fovea. Diese Eigenschaft soll in unseren

a) b) c)

Abbildung 4.6: Ein Beispiel für Objektapproximation. (a) Lokalisation der Wavelets nach der Initialisierung (b) Lokalisation der Wavelets nach der Optimierung (c) Rekonstruktion der beiden Objekte.

Waveletnetzen zur Redundanzreduktion ausgenutzt werden, indem wir die Fovea mit den Objektzentren identifizieren und die Peripherie mit der Objektumgebung. Diese Eigenschaft zeichnet aber nicht allein verantwortlich für die immense Datenreduktion, die von der Retina der Primaten vollzogen wird. Die Ausgänge der Zapfen und Stäbchen werden anschließend noch summiert und gemittelt. Sie sind räumlich organisiert in Cluster, sogenannte *Rezeptive Felder* (RF). Die Weiterleitung der Information geschieht durch die Ganglienzellen und ihre Axone zum visuellen Cortex. Die in der sogenannten V1 des visuellen Cortex entstehende Abtastung der Retina kann als log-polar modelliert werden. Dieses Modell der sich überlappenden rezeptiven Felder nutzten Bolduc und Levin bei ihrer Entwicklung einer künstlichen Retina für das visuelle System eines autonomen Roboters (siehe Abbildung 4.7).

Wir wollen die Rezeptiven Felder mit den Wavelets identifizieren, um so eine geeignete Initialisierung der Objekte im Sinne einer Fovea-Peripherie-Beziehung zu erreichen. Der Grad der Überlappung steht an dieser Stelle weniger im Vordergrund. Die Grundlage dieses Modells der RF's bildet die komplex-logarithmische oder auch log-polare Abbildung. Sie wird von K. Daniilidis in [4] wie folgt beschrieben: Verwenden wir als kartesische Koordinaten $(x, y)$ und als Polarkoordinaten $(\rho, \eta)$, so wird ein Punkt in der komplexen Ebene beschrieben durch

$$z = x + jy = \rho e^{j\eta}. \tag{4.8}$$

Die komplex-logarithmische oder auch log-polare Abbildung ist dann definiert als

$$\omega = \ln(z), \qquad z \neq 0, \tag{4.9}$$

wobei $Re(\omega) = \ln \rho$ und $Im(\omega) = \eta + 2\pi k$. Da der Imaginärteil periodisch in $2\pi$ ist, betrachten wir nur das Intervall $[0, 2\pi)$.

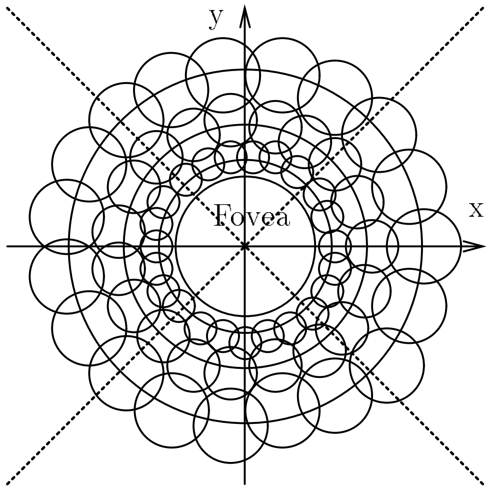



Abbildung 4.7: Das Modell der Rezeptiven Felder mit Überlappung, die Basis der Arbeit von Bolduc und Levine.

In Abbildung 4.8 ist die log-polare Abbildung illustriert, wobei $\xi = \ln \rho$. Der Nullpunkt stellt eine Singularität dar, und da unser kartesisches Gitter durch Pixel gebildet wird, sind hier dem Wertebereich der Abbildung auch Grenzen $\rho_0 \leq \rho \leq \rho_{max}$ gesetzt. Der Radius $\rho_0$ beträgt in der üblichen Auflösung von Grauwertbildern ein Pixel und $\rho_{max}$ die halbe Objektbreite. Die Abbildung der polaren Koordinaten $(\rho, \eta)$ in die log-polaren Koordinaten $(\xi, \gamma)$ wird durch folgende Transformation geleistet:

$$\xi = \log_a(\frac{\rho}{\rho_0}) \qquad \text{und} \qquad \gamma = \frac{N_a}{2\pi}\eta. \tag{4.10}$$

Bezeichnet man $N_a$ als Winkelauflösung und $N_r$ als radiale Auflösung, so erhält man die Basis $a$ aus dem „Fovea-Radius" $\rho_0$, der Objektbegrenzung $\rho_{max}$ und der radialen Auflösung $N_r$

$$a^{N_r} = \frac{\rho_{max}}{\rho_0} \tag{4.11}$$

$$a = e^{\frac{1}{N_r}\ln(\frac{\rho_{max}}{\rho_0})}. \tag{4.12}$$

(4.12) zeigt, daß das Modell der log-polaren Initialisierung durch drei Größen bestimmt wird. Der „Fovea-Radius" $\rho_0$ ist durch die Pixelauflösung von einem

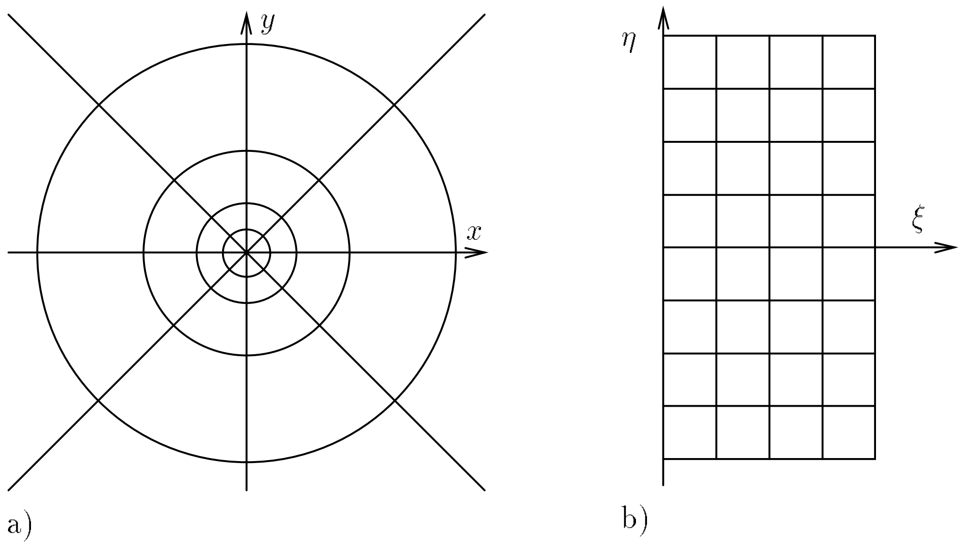



Abbildung 4.8: Bei der log-polaren Abbildung werden radiale Linien und konzentrische Kreise (a) in Linien parallel zu den Koordinatenachsen abgebildet (b).

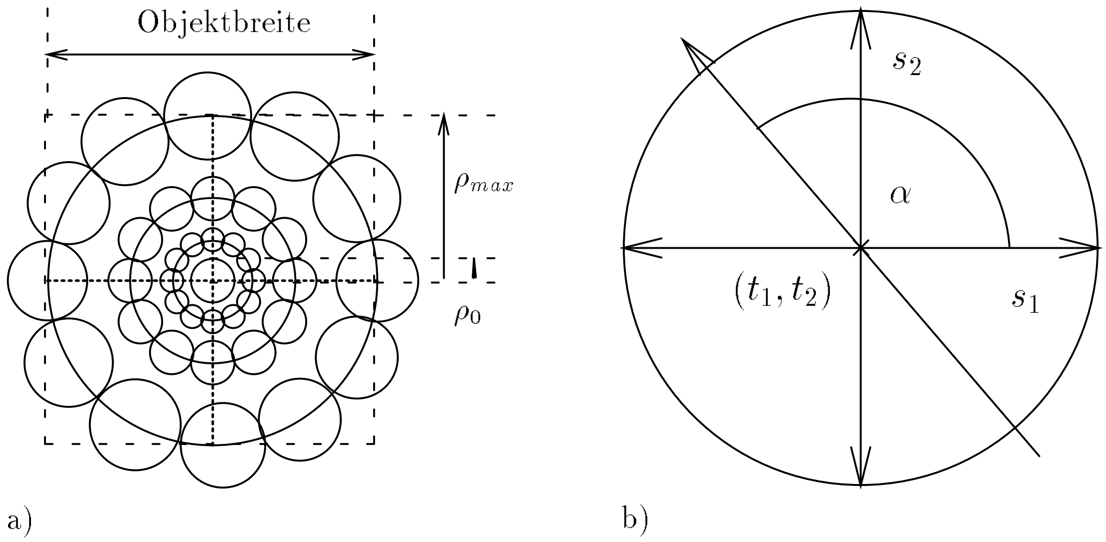



Abbildung 4.9: Die log-polare Initialisierung eines Objektes. (a) Waveletverteilung in den Sektoren, (b) Die Zuordnung der Parameter ($t_1$, $t_2$, $s_1$, $s_2$, $\alpha$). Das Gewicht $\omega$ kann hier nicht veranschaulicht werden.

Pixel gegeben. Die Objektbegrenzung $\rho_{max}$ wird durch die Objektauswahl bestimmt. Somit ist die radiale Auflösung $N_r$ die einzige Größe, die willkürlich gesetzt werden muß. Sie gibt die log-polare (nichtlineare) Abtastdichte im Objekt vor. Nachdem das Modell der log-polaren Initialisierung erläutert wurde, müssen die Waveletparameter mit dem Modell identifiziert werden. Die Initialisierung der Orte ist einfach, sie entsprechen den Schnittpunkten des log-polaren Rasters. Bei der Skaleninitialisierung wollen wir in ersten Versuchen im Gegensatz zu Bolduc und Levin keine Überlappung zulassen. Wir halbieren die Sektoren und ordnen die in Pixel gemessenen Skalen den jeweils außen liegenden Wavelets zu. Fällt die so berechnete Skala unter die Grenze von drei Pixel, so wird sie auf drei gesetzt. Die Initialisierung der Winkel ist wie bisher zufällig, und die Gewichte werden wie bei den Pyramiden projiziert oder willkürlich gesetzt. Die Abbildung 4.9 soll die eben beschriebene Initialisierung des Modells noch einmal veranschaulichen.

Die zuletzt besprochene Waveletnetz-Struktur hat eine kompakte Objektstruktur in dem Sinne, daß alle Bildelemente, ob Hintergrund oder Detail, als Objekte aufgefaßt werden. Die Zuordnung der lokalen Bildinformation zu einem Objekt obliegt genauso dem Beobachter (Aufgabensteller) wie die Auswahl des Objektmodelles und der Reihenfolge der Optimierung (und damit auch der Rekonstruktion). Der letzte Aspekt, die Reihenfolge der Objekte bei der Optimierung, bestimmt die Progressivität des Codier- und damit auch des Rekonstruktionsvorganges. Die Abbildung 4.10 zeigt die Initialisierung von drei Objekten und ihre schrittweise Rekonstruktion, um diesen Aspekt der Methode zu verdeutlichen. Damit stellt die *progressive Wavelet-Approximation* (PWA) ein aufgabenbezogenes Bildkompressionsverfahren dar, welches im nächsten Kapitel an verschiedenen Beispielen und unter verschiedenen Aspekten der Bildkompression ausführlich getestet und ausgewertet werden soll.

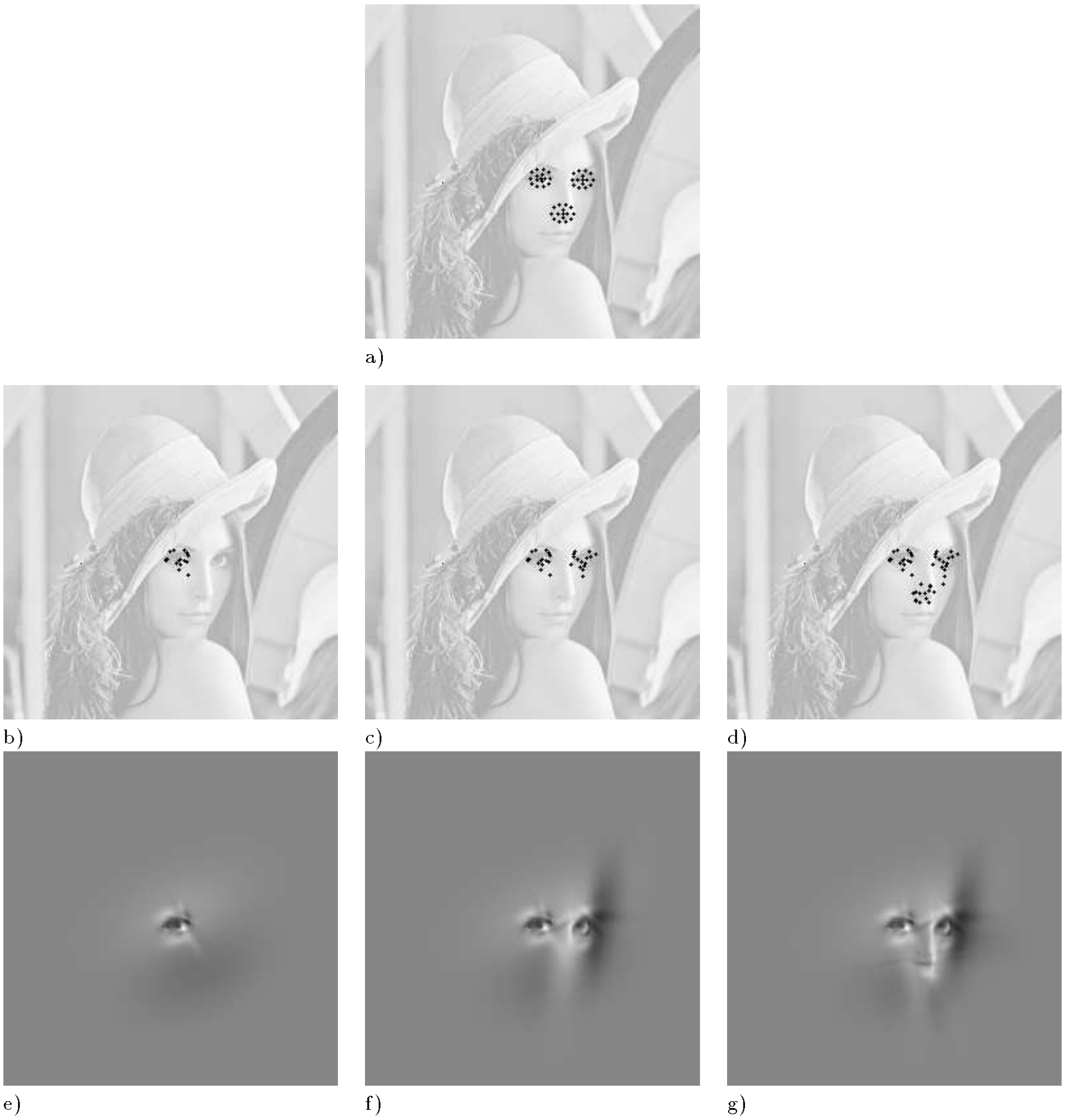



Abbildung 4.10: (a) Die log-polare Initialisierung von drei Objekten. (b) bis (d) Orte nach der Optimierung des jeweiligen Objektes. (e) bis (g) Rekonstruktion des jeweiligen Objektes.

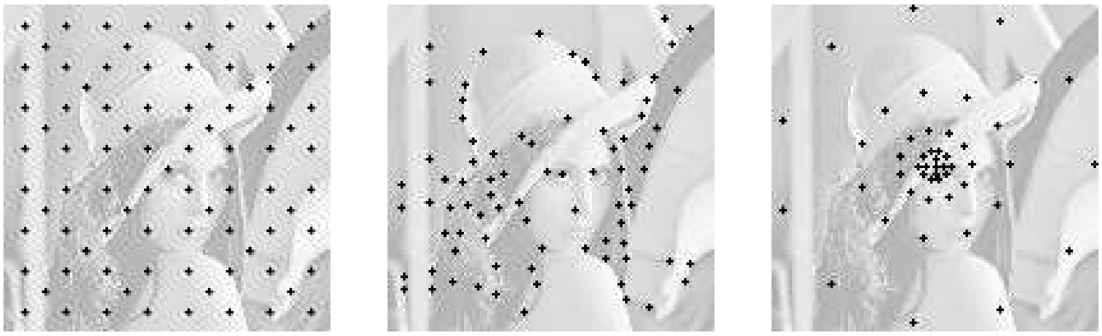
# Kapitel 5

# Auswertung

Nachdem die Modellierung der Waveletnetze mit der Objektinitialisierung abgeschlossen ist, soll das Verfahren der progressiven Wavelet-Approximation (PWA) in diesem Kapitel an Bildern aus verschiedenen Bildklassen getestet werden. Im Vordergrund steht dabei die Güte der Rekonstruktion im Hinblick auf die Wiedererkennung der Objekte. Die für die Auswertung relevanten Größen sind der Signal-Rausch-Abstand ($SNR$) sowie die Coderate ($CR$). Um auch quantitative Aussagen über die Objekte machen zu können, werden die oben genannten Größen für die einzelnen Objekte im Verhältnis zum gesamten Netz betrachtet. Weiterhin soll der Aspekt der Progressivität etwas näher beleuchtet werden, indem ein und dasselbe Netz unter verschiedenen Progressionsmerkmalen rekonstruiert wird. Den Abschluß dieser Auswertung bildet ein Vergleich der PWA mit einer klassischen Variante der Waveletzerlegung, der Subband-Codierung (SBC).

## 5.1 Die attentive Selektion von Objekten

In diesem Abschnitt soll die Wirkung der attentiven Selektion von Objekten auf ihre Wiedererkennung in der Rekonstruktion untersucht werden. Dabei kommt es insbesondere entsprechend einer Aufgabenstellung auf die zweckmäßige Wahl der Objekte und der Objektmodelle an. Die in den letzten Kapiteln vorgestellten drei Objektmodelle sind zur Erinnerung in Abbildung 5.1 noch einmal dargestellt. Auch die Hintergrund-Objekt-Wechselwirkung ist wesentlich für die Approximationsgüte. Deshalb soll auch diese Wechselwirkung genauer untersucht werden. Zunächst werden jedoch die für die Auswertung relevanten Größen bestimmt, der $SNR$ und die $CR$. Der $SNR$ wird für jede Rekonstruktion sowohl für das gesamte Bild als auch für die einzelnen Objekte in Dezibel berechnet. Bezeichnen wir das Objekt mit $O$ und seine Rekonstruktion mit $R$, dann berechnet sich der $SNR$ wie folgt:



a) b) c)

Abbildung 5.1: Die drei zur Verfügung stehenden Objektmodelle (hier für das Beispiel „Lena“) (a) die Pyramide, (b) das Gaußsche Krümmungsmaß, (c) die log-polare Initialisierung.

$$SNR_O = 10 \cdot \log_{10} \frac{\sum_{i=1}^{M \times N} g_i^2}{\sum_{i=1}^{M \times N} (g_i - f_i)^2}, \qquad g_i \in O, f_i \in R. \tag{5.1}$$

Zur Berechnung der Coderate ist es notwendig, die Parameter des Waveletnetzes zu quantisieren. Eine naive Quantisierung geht davon aus, den kleinsten bzw. größten vorkommenden Wert der vorhandenen Parametermenge zu finden und die Differenz geeignet, d.h. den Quantisierungsfehler minimierend, in eine Anzahl Intervalle aufzuteilen. Unter der Annahme, daß die Rekonstruktion die gleiche Auflösung wie das Original besitzt, macht es beispielsweise keinen Sinn, die Orte feiner zu quantisieren als ein Pixel, genausowenig wie die Quantisierung der Rotation feiner als ein Grad. Nachdem jeder Parameter einem Intervall zugewiesen wurde, wird die Intervallnummer bitweise ausgelesen und das gesamte Netz als Bytestream im Sinne einer progressiven Übertragung abgelegt. Bytestream bedeutet hier, daß jeweils ein Byte aufgefüllt und dann ausgelesen wird.

### 5.1.1 Objekt und Objektmodell

Soll ein Bild aufgabenbezogen in einem Waveletnetz repräsentiert werden, so ist es notwendig, die Aufgabenstellung in eine Objektauswahl umzusetzen. Die in Abbildung 5.1 dargestellten Objektmodelle müssen zudem zweckmäßig ausgewählt werden. Deshalb soll an dieser Stelle noch einmal der direkte Vergleich der drei Modelle am Bild „Lettern“ (Abbildung 5.2 (a)) gemacht werden. Die Bildinformation besteht hier aus einem schwarzen Schriftzug auf hellem Grund. Bei der Initialisierung des Netzes durch eine Pyramide ist zu erwarten, daß mit steigender Stufenzahl der $SNR$ der Rekonstruktion immer höher wird. Bei einer attentiven Initialisierung über das Krümmungsmaß ist nur entlang der Lettern eine Krümmung zu erwarten und dementsprechend eine bessere lokale Anpassung an diesen Stellen, was sich nicht unbedingt im $SNR$ widerspiegeln muß, aber in einer besseren Wiedererkennung des Textes. Eine log-polare Initialisierung über das gesamte Bild sollte zu einer sehr guten Anpassung der fovealen Bildregion und einer weniger guten Anpassung der Peripheriebereiche führen, was gleichzusetzen ist mit einem hohen $SNR$ in der Fovea und entsprechend niedrigerem $SNR$ in der Peripherie. Die Abbildung 5.2 zeigt die Rekonstruktionen der entsprechenden optimierten Netze.

Die Rekonstruktion der vierstufigen Pyramide (5.2 (b2)) stellt eine relativ gute globale Approximation des Schriftzuges dar. Die einzelnen Worte sind als schwarze Balken gut zu unterscheiden. An einigen Punkten, an denen lokale Information – einzelne Buchstaben – und Waveletlokalisationen zufällig zusammentreffen, ist auch eine lokale Anpassung andeutungsweise vorhanden. Es ist aber unmöglich, den Schriftzug zu lesen. Die attentive Initialisierung über das Gaußsche Krümmungsmaß bringt hier mehr Vorteile. Die Initialisierung der Wavelets findet lokal statt, was sich am Ende auch global auswirkt. Der Schriftzug wird – wenn auch schwer – lesbar. Die Rekonstruktion des log-polar initialisierten Netzes macht die Bedeutung der attentiven Initialisierung am besten sichtbar. Dort werden globale und lokale Anpassung des Netzes an die Bildinformation verschmolzen. In der Peripherie sind die einzelnen Worte wieder als eigene Objekte zu unterscheiden, aber unlesbar. Im fovealen Bereich dagegen sind die einzelnen Lettern deutlich lesbar approximiert. Die Versuchsergebnisse bestätigen also unsere Vorüberlegungen. Im folgenden soll an verschiedenen Bildern die Auswahl der Objekte und auch der Modelle im Hinblick auf die Erkennung in der Rekonstruktion untersucht werden. Es bleibt auch zu klären, wie aussagekräftig der $SNR$ als Maß der Approximationsgüte ist. Zu diesem Zweck müssen die Objekte in irgendeiner Art und Weise klassifiziert werden. Eine, wenn auch willkürliche, Einteilung, die dies leistet, ist die Einteilung in natürliche Objekte (Gesichter, Augen, Früchte, ...), künstliche Objekte (Gegenstände, Gebäude, Fahrzeuge, ...) und Texturobjekte. Diese Klassifizierung nimmt nicht in Anspruch, universell gültig zu sein, sie reicht aber für unsere Zwecke zur Untersuchung der ebenso willkürlichen Modelle aus. Die nun folgende Testserie, bei der unsere drei Objektmodelle auf je zwei Vertreter der drei Objektklassen angewendet werden, soll die Vor- und Nachteile der jeweiligen Modellierung aufzeigen. Maßstab aller Dinge sind dabei der $SNR$, die $CR$ und nicht zuletzt der visuelle Vergleich von Bild und Rekonstruktion. Tabelle 5.1 enthält die Ergebnisse der Versuchsreihe. Die zugehörigen Rekonstruktionen entnehme man Abbildung 5.3.

Zur Vereinfachung der Auswertung bezeichnen wir die drei Objektmodelle (Abbildung 5.1) im folgenden mit „Pyramide“ – für die Objektinitialisierung über eine Pyramide – mit „Krümmung“ für die Objektinitialisierung über das Gaußsche Krümmungsmaß – und mit „Retina“ – für die log-polare Initialisierung. Es werden sechs Versuchsserien durchgeführt, die in Tabelle 5.2 zusammengestellt sind.

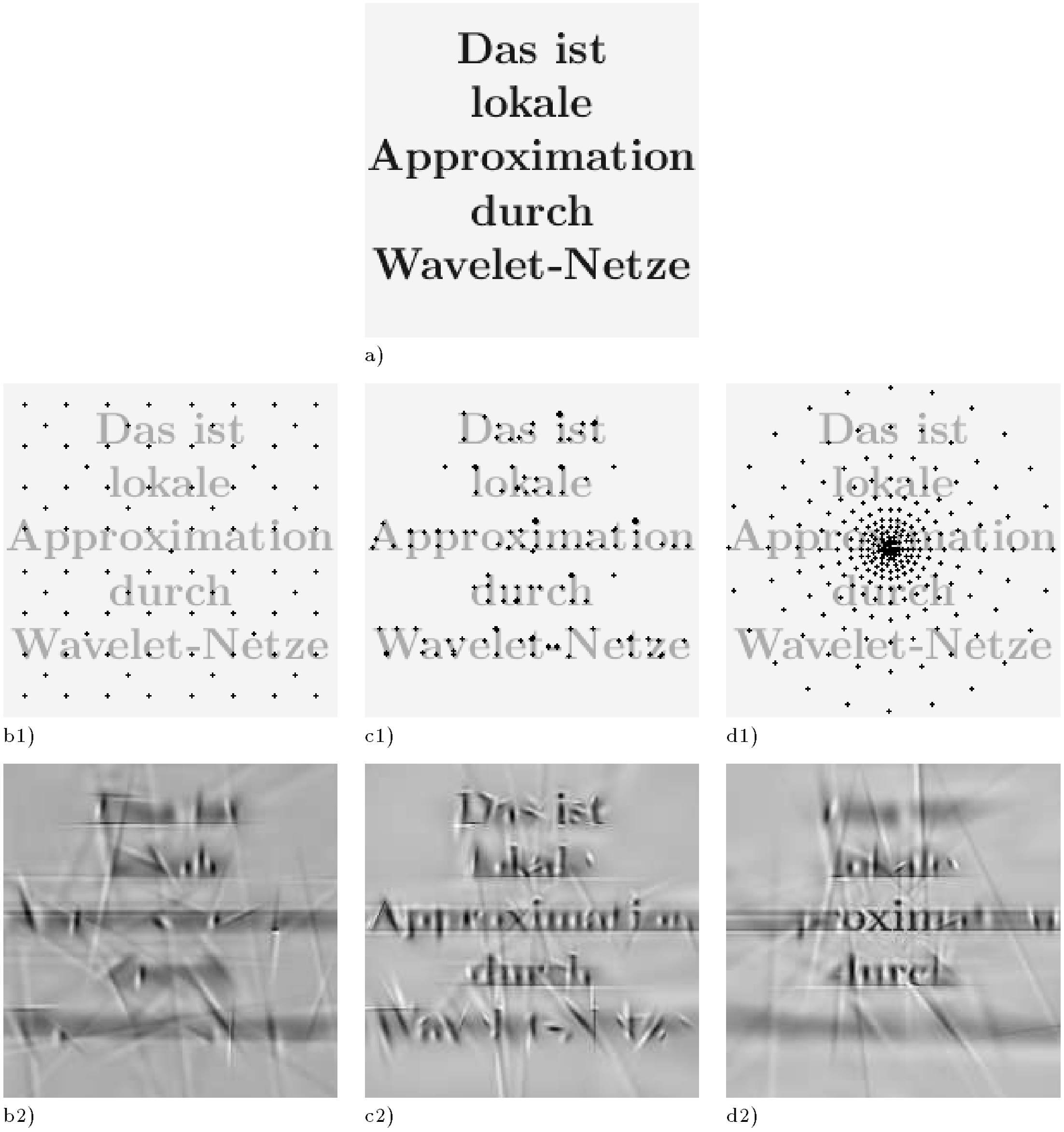



Abbildung 5.2: Die Wirkung der drei Arten der Initialisierung. (a) Das Originalbild „Lettern“, (b1) Initialisierung mit einer vierstufigen Pyramide, (b2) deren Rekonstruktion, globale Approximation, Fehlerminimierung über das gesamte Bild, (c1) attentive Initialisierung über das Gaußsche Krümmungsmaß, (c2) gute lokale Approximation der einzelnen Lettern, (d1) logpolare Initialisierung, (d2) der Fovea-Effekt bei der Approximation wird sehr deutlich.

| Objekt | Modell | Anzahl Wavelets | Code [Byte] | Coderate [Bit/Pxl] | $SNR$ [dB] | Rekon-struktion |
|---|---|---|---|---|---|---|
| | Retina | 36 | 180 | 0.400 | 21.50 | (a3) |
| | Pyramide | 42 | 195 | 0.433 | 23.48 | (a5) |
| Auge | Krümmung | 35 | 184 | 0.409 | 21.97 | (a7) |
| | Retina | 49 | 245 | 0.544 | 28.27 | (b3) |
| | Pyramide | 42 | 205 | 0.456 | 29.13 | (b5) |
| Nase | Krümmung | 43 | 215 | 0.478 | 30.02 | (b7) |
| | Retina | 78 | 429 | 0.953 | 12.10 | (c3) |
| | Pyramide | 85 | 468 | 1.040 | 13.06 | (c5) |
| Leiter | Krümmung | 86 | 473 | 1.051 | 13.40 | (c7) |
| | Retina | 79 | 435 | 0.967 | 11.38 | (d3) |
| | Pyramide | 85 | 457 | 1.016 | 12.59 | (d5) |
| Fahrzeug | Krümmung | 80 | 440 | 0.978 | 11.58 | (d7) |
| | Retina | 80 | 460 | 1.022 | 6.13 | (e3) |
| | Pyramide | 85 | 488 | 1.086 | 5.21 | (e5) |
| Textur fein | Krümmung | 80 | 490 | 1.089 | 6.43 | (e7) |
| | Retina | 80 | 420 | 0.933 | 9.21 | (f3) |
| | Pyramide | 85 | 479 | 1.064 | 9.51 | (f5) |
| Textur grob | Krümmung | 61 | 328 | 0.729 | 7.97 | (f7) |

Tabelle 5.1: Die Ergebnisse der Versuchsreihen A bis F entsprechend Tabelle 5.2 zur Objektmodellierung. Es wurden je zwei Vertreter unserer willkürlichen Objektklassen mit den drei Objektmodellen initialisiert. In der letzten Spalte sind die Referenzen der entsprechenden Rekonstruktionen aus Abbildung 5.3 enthalten.

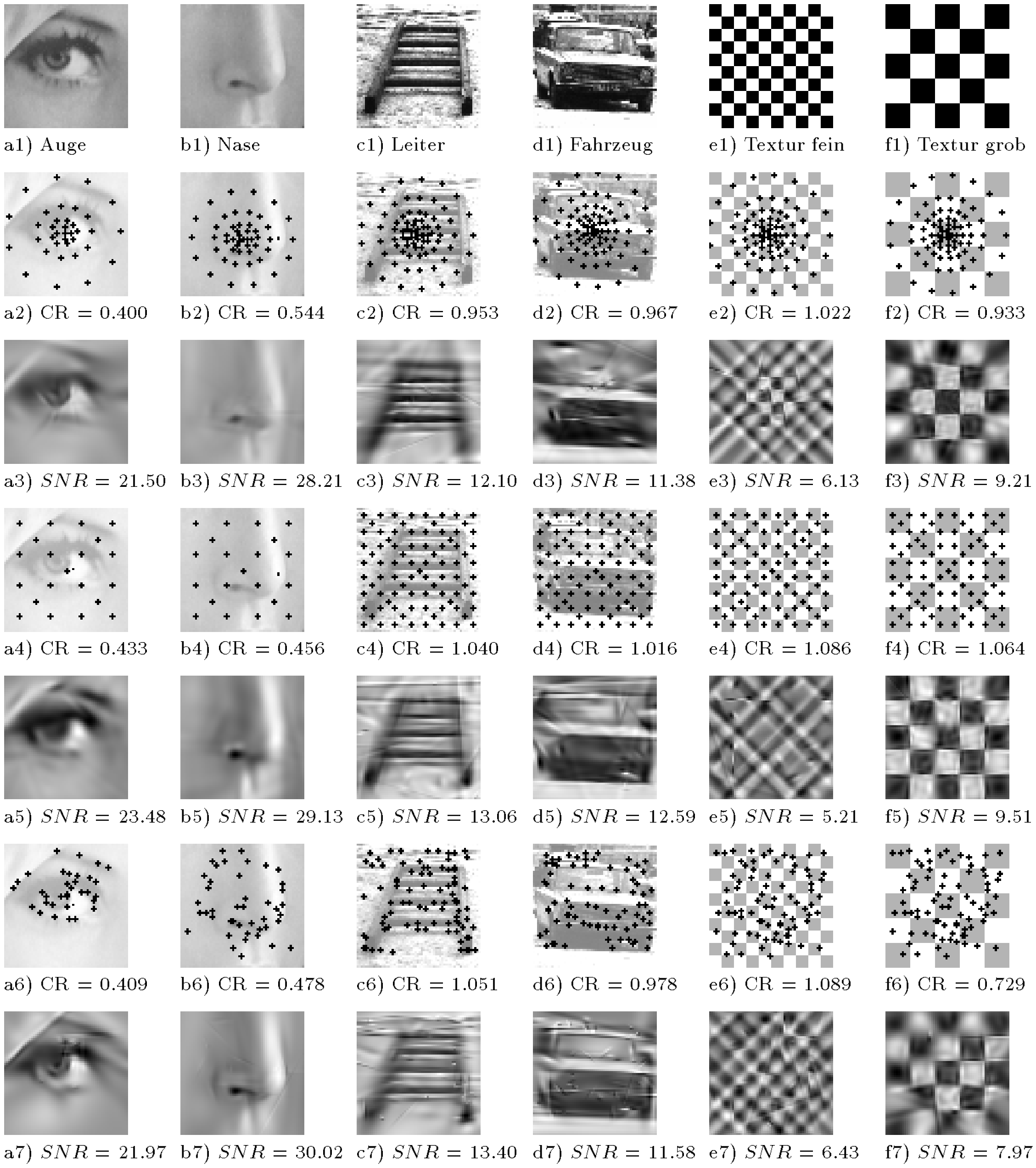



Abbildung 5.3: Die Testserien A bis F entsprechend Tabelle 5.2 zur Modellierung der drei Objektklassen natürliche Objekte (Auge, Nase), künstliche Objekte (Leiter, Fahrzeug) und Textur. Jede Spalte entspricht einem Objekt. Die erste Zeile enthält alle Originalbilder, danach gehören jeweils zwei Zeilen zusammen, die erste gibt die Initialisierung mit der Coderate $CR$ in Bit/Pixel wieder, und die zweite gibt die Rekonstruktion mit dem Signal-Rausch-Verhältnis $SNR$ in dB wieder.

| Serie | Objektklasse | Objekt |
|---|---|---|
| A | natürlich | Auge |
| B | natürlich | Nase |
| C | künstlich | Leiter |
| D | künstlich | Fahrzeug |
| E | Textur (fein) | Karo (fein) |
| F | Textur (grob) | Karo (grob) |

Tabelle 5.2: Die Zusammenstellung der Versuchsserien A bis F zur Untersuchung der Modellierung verschiedener Objektklassen.

### 1. Modellierung natürlicher Objekte

Als Vertreter der Klasse der natürlichen Objekte sollen uns hier das rechte Auge (Serie A, Abb. 5.3 (a1)) und die Nase (Serie B, Abb. 5.3 (b1)) unseres altbekannten Bildes „Lena“ dienen. In Serie A fällt bei etwa gleicher Coderate $CR \sim 0.4$ der $SNR$ bei der Rekonstruktion der Pyramide mit $SNR = 23.48$ am höchsten aus, gefolgt von der Krümmung mit $SNR = 21.97$. Die Retina zeigt mit $SNR = 21.50$ das schlechteste Ergebnis. In Serie B verhält es sich etwas anders. Hier bringt mit $SNR = 30.02$ die Krümmung die beste Rekonstruktion, gefolgt von der Pyramide mit $SNR = 29.13$. Die Retina hat wieder die schlechteste Rekonstruktion mit $SNR = 28.21$. Vergleicht man jedoch alle drei Rekonstruktionen mit den Originalen, so schneidet visuell die Retina sowohl in Serie A als auch in Serie B besser ab. Der $SNR$ ist bei unwesentlich höherer Coderate für Serie B größer. Dies hat seine Ursache in der Detailarmut dieses Objektes.

### 2. Modellierung künstlicher Objekte

Auch hier wurden die verschiedenen Objektmodelle bei etwa gleicher Coderate $CR \sim 1.0$ zur Initialisierung gebracht. In Serie C ist der $SNR$ bei der Krümmung $SNR = 13.40$ am größten, gefolgt von der Pyramide mit $SNR = 13.06$, eine unwesentliche Differenz. Die Retina liefert hier mit $SNR = 12.10$ das schlechteste Ergebnis, was sich schließlich auch im visuellen Vergleich widerspiegelt. In der Serie D ist die Situation etwas anders. Die Rekonstruktion der Pyramide liegt mit $SNR = 12.59$ ein dB höher als bei der Krümmung mit $SNR = 11.58$. Der $SNR$ der Retina ist mit $SNR = 11.38$ am geringsten. Dies entspricht auch dem visuellen Vergleich der Rekonstruktionen, denn die Retina hat nur die Vorderfront des Fahrzeugs lokal gut approximiert, der Rest des Objektes ist kaum wiederzuerkennen. Aber auch hier fällt der visuelle Vergleich der Rekonstruktionen der Krümmung und der Pyramide zugunsten des geringeren $SNR$ aus.

**3. Modellierung von Textur**

Die letzten beiden Objekte sind einfache Beispiele für feine (Abb. 5.3 (e1)) und grobe (Abb. 5.3 (f1)) regelmäßige Textur. Es handelt sich um das einfache Testmuster eines Schachbretts. In Serie E ist die Rekonstruktion bei der Krümmung mit $SNR = 6.43$ am besten, gefolgt von der Retina mit $SNR = 6.13$. Die Pyramide erbringt in diesem Beispiel mit $SNR = 5.21$ die schlechteste Rekonstruktion. Dieses Ergebnis läßt sich auch durch den visuellen Vergleich bestätigen. Auffällig bei dieser Serie ist die Beobachtung, daß das Waveletnetz offenbar in allen drei Versuchen die Diagonalen des Musters besser approximiert als die horizontale und vertikale Musterung. Dies ist möglicherweise bedingt durch die Wahl des Filters (Gabor), soll aber an dieser Stelle nicht weiter untersucht werden. In Serie F ist die Periode des Schachbrettmusters verdoppelt. Die Ergebnisse fallen in dieser Serie etwas anders aus. Die Pyramide zeigt mit $SNR = 9.51$ sowohl für den $SNR$ als auch im visuellen Vergleich deutlich das beste Resultat. Die Retina erzielt mit $SNR = 9.21$ das zweitbeste Ergebnis, dies bestätigt sich auch visuell. Mit $SNR = 7.97$ ist die Rekonstruktion der Krümmung die schlechteste.

Diese Ergebnisse bestätigen zum einen die fovealen Qualitäten der Retina, die besonders bei natürlichen Objekten von Vorteil sind, und zum anderen die Schwierigkeit, allein mit dem $SNR$ eine Aussage über die Approximationsgüte zu machen. Bei künstlichen Objekten scheint die Krümmung das bessere Modell zu sein, da lokale Approximation mehr zur visuellen Wiedererkennung beiträgt als die globale Minimierung des Gesamtfehlers, wie es bei der Pyramide der Fall ist. Diese letzte Eigenschaft bringt aber einige Vorteile bei der Approximation von grober Textur, wie man sie in vielen Bildern als Hintergrund vorfindet. Deshalb soll sich der nächste Abschnitt der Wechselwirkung zwischen Hintergrund- und Objektinitialisierung widmen.

### 5.1.2 Hintergrund- und Objektinitialisierung

Im letzten Abschnitt wurde die Modellierung dreier willkürlich gewählter Objektklassen untersucht. Dabei wurde der Hintergrund, der in jedem Bild vorhanden ist, völlig vernachlässigt, was die Ergebnisse etwas verfälscht. Deshalb soll an dieser Stelle die Rolle des Hintergrundes bei der Approximation der Bildinformation genauer untersucht werden. Für die weiteren Experimente soll das Bild „Child“ (siehe Abbildung 5.5 (a1)) dienen. Es zeichnet sich durch deutliche natürliche Objekte im Vordergrund (Gesicht) und durch einen detailreichen Hintergrund (texturiertes Kleidungsstück) aus. Wir wollen an diesem Bild drei Serien zur Untersuchung der Wechselwirkung zwischen der Objekt- und der Hintergrundinitialisierung durchführen. Als Objekte sollen, wie in Abbildung 5.5 (a2) zu sehen, die Augen, die Nase und der Mund als Vordergrundobjekte log-polar initialisiert werden. In Serie A wird der Hintergrund über eine Pyramide initialisiert, wobei

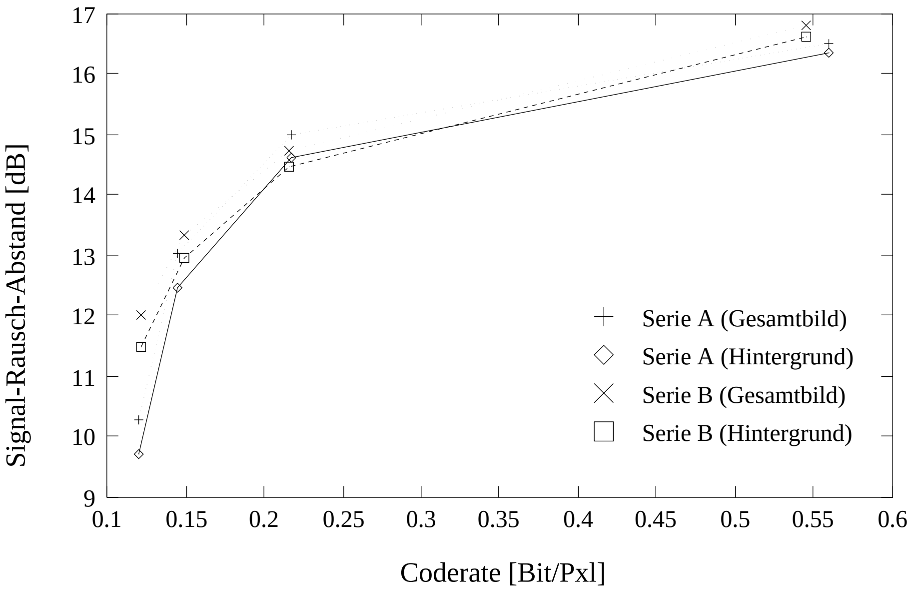



Abbildung 5.4: Die Entwicklung des $SNR$ für das Gesamtbild und für den Hintergrund allein bei den Versuchsserien A (Hintergrundinitialisierung über die Krümmung) und B (Hintergrundinitialisierung über die Pyramide).

von Experiment zu Experiment die Anzahl der Stufen erhöht wird. In Serie B wird in ähnlicher Weise der Hintergrund über das Gaußsche Krümmungsmaß initialisiert, wobei die Anzahl der lokalen Krümmungspunkte so beschränkt wurde, daß sie in etwa mit der Anzahl der Wavelets aus Serie A vergleichbar ist. Für beide Serien wurde bei etwa gleichen Coderaten der $SNR$ für alle Objekte (einschließlich Hintergrund) und für das Gesamtbild berechnet. Die Daten werden durch die Tabellen 5.3 und 5.4 wiedergegeben.

Diagramm in Abb. 5.4 zeigt die Entwicklung des $SNR$ beider Serien sowohl für den Hintergrund allein als auch für das Gesamtbild. Hier zeigt Serie B einen leicht höheren $SNR$ als Serie A.

Auch im visuellen Vergleich (siehe Abbildung 5.5 (a) und (b)) zeigt Serie B eine bessere Approximation des Hintergrundes, vor allem in den höheren Stufen.

Offensichtlich hat die sehr gute Approximation der Objekte im Vordergrund keinen sehr großen Effekt auf den $SNR$ des Gesamtbildes. Dieses Phänomen soll in einer weiteren Versuchsserie C etwas genauer untersucht werden. Wir beschränken uns der besseren Handhabbarkeit wegen auf die Initialisierung sowohl des Vordergrundobjektes als auch des Hintergrundes mit Hilfe einer Pyramide. Das Vordergrundobjekt soll in diesem Versuch durch die Augenpartie gegeben sein (siehe Abbildung 5.7 (a)). Für die Berechnung des $SNR$ wird sowohl die Stufenzahl der Hintergrundpyramide als auch die der Objektpyramide von null bis fünf variiert.

| Nr. | Objekt | Anzahl Stufen | Anzahl Wavelets | Code [Byte] | Coderate [Bit/Pxl] | $SNR$ [dB] | Rekon-struktion |
|---|---|---|---|---|---|---|---|
| | Hintergrund | 0 | 0 | 0 | 0 | - | |
| | Auge links | 3 | 25 | 131 | 0.028 | 19.83 | |
| | Auge rechts | 3 | 26 | 136 | 0.030 | 15.35 | |
| | Nase | 3 | 24 | 126 | 0.027 | 15.23 | |
| | Mund | 3 | 21 | 110 | 0.024 | 9.39 | |
| 1 | Gesamt | | 96 | 504 | 0.109 | 7.9 | (c3) |
| | Hintergrund | 2 | 5 | 27 | 0.006 | 9.72 | |
| | Auge links | 3 | 26 | 143 | 0.031 | 15.62 | |
| | Auge rechts | 3 | 26 | 143 | 0.031 | 22.00 | |
| | Nase | 3 | 22 | 121 | 0.026 | 25.49 | |
| | Mund | 3 | 22 | 121 | 0.026 | 22.72 | |
| 2 | Gesamt | | 101 | 556 | 0.121 | 10.29 | (a1) |
| | Hintergrund | 3 | 21 | 123 | 0.027 | 12.48 | |
| | Auge links | 3 | 25 | 146 | 0.032 | 11.29 | |
| | Auge rechts | 3 | 25 | 146 | 0.032 | 17.59 | |
| | Nase | 3 | 22 | 129 | 0.028 | 23.63 | |
| | Mund | 3 | 21 | 123 | 0.027 | 24.60 | |
| 3 | Gesamt | | 114 | 670 | 0.145 | 13.05 | (a2) |
| | Hintergrund | 4 | 85 | 467 | 0.101 | 14.64 | |
| | Auge links | 3 | 25 | 137 | 0.030 | 15.07 | |
| | Auge rechts | 3 | 25 | 137 | 0.030 | 22.11 | |
| | Nase | 3 | 22 | 121 | 0.026 | 22.67 | |
| | Mund | 3 | 25 | 137 | 0.030 | 22.52 | |
| 4 | Gesamt | | 182 | 1001 | 0.217 | 15.01 | (a3) |
| | Hintergrund | 5 | 341 | 2003 | 0.435 | 16.36 | |
| | Auge links | 3 | 25 | 146 | 0.032 | 13.30 | |
| | Auge rechts | 3 | 26 | 152 | 0.033 | 22.94 | |
| | Nase | 3 | 21 | 123 | 0.027 | 20.49 | |
| | Mund | 3 | 25 | 146 | 0.032 | 26.18 | |
| 5 | Gesamt | | 438 | 2574 | 0.559 | 16.52 | (a4) |

Tabelle 5.3: Die Ergebnisse der Versuchsserie A (Hintergrundinitialisierung mit Pyramide) zur Untersuchung der Wechselwirkung zwischen der Hintergrund- und Objektinitialisierung. In jedem Versuch wurde die Anzahl Hintergrundstufen erhöht. Die letzte Spalte enthält die Referenz der zugehörigen Rekonstruktion aus Abbildung 5.5. Die Anzahl Stufen bezieht sich beim Hintergrund auf die Anzahl Pyramidenstufen, bei den Objekten ist dies die radiale Auflösung der log-polaren Initialisierung (siehe Abbildung 5.5 (c2)).

| Nr. | Objekt | Anzahl Stufen | Anzahl Wavelets | Code [Byte] | Coderate [Bit/Pxl] | $SNR$ [dB] | Rekon-struktion |
|---|---|---|---|---|---|---|---|
| | Hintergrund | 2 | 9 | 50 | 0.011 | 11.49 | |
| | Auge links | 3 | 24 | 135 | 0.029 | 14.97 | |
| | Auge rechts | 3 | 25 | 140 | 0.031 | 20.37 | |
| | Nase | 3 | 21 | 118 | 0.026 | 21.96 | |
| | Mund | 3 | 21 | 118 | 0.026 | 22.69 | |
| 1 | Gesamt | | 100 | 563 | 0.122 | 12.00 | (b1) |
| | Hintergrund | 3 | 24 | 135 | 0.029 | 12.95 | |
| | Auge links | 3 | 25 | 140 | 0.031 | 14.39 | |
| | Auge rechts | 3 | 25 | 140 | 0.031 | 20.10 | |
| | Nase | 3 | 24 | 135 | 0.029 | 24.05 | |
| | Mund | 3 | 24 | 135 | 0.029 | 24.26 | |
| 2 | Gesamt | | 122 | 687 | 0.149 | 13.34 | (b2) |
| | Hintergrund | 4 | 81 | 455 | 0.099 | 14.46 | |
| | Auge links | 3 | 25 | 140 | 0.031 | 14.57 | |
| | Auge rechts | 3 | 25 | 140 | 0.031 | 19.79 | |
| | Nase | 3 | 24 | 135 | 0.029 | 22.09 | |
| | Mund | 3 | 22 | 123 | 0.027 | 24.88 | |
| 3 | Gesamt | | 177 | 996 | 0.216 | 14.74 | (b3) |
| | Hintergrund | 5 | 342 | 1966 | 0.427 | 16.63 | |
| | Auge links | 3 | 25 | 143 | 0.031 | 15.14 | |
| | Auge rechts | 3 | 24 | 138 | 0.030 | 19.40 | |
| | Nase | 3 | 25 | 143 | 0.031 | 25.53 | |
| | Mund | 3 | 21 | 120 | 0.026 | 26.68 | |
| 4 | Gesamt | | 437 | 2513 | 0.545 | 16.80 | (b4) |

Tabelle 5.4: Die Ergebnisse der Versuchsserie B (Hintergrundinitialisierung über Krümmung) zur Untersuchung der Wechselwirkung zwischen der Hintergrund- und Objektinitialisierung. In jedem Versuch wurde die Anzahl Hintergrundstufen erhöht, wobei der Versuch ohne Hintergrund mit dem Versuch 1 aus Tabelle 5.3 identisch ist. Die letzte Spalte enthält die Referenz der zugehörigen Rekonstruktion aus Abbildung 5.5. Die Anzahl Stufen bezieht sich beim Hintergrund auf die Anzahl Skalen, für die die Krümmung berechnet wurde, bei den Objekten ist dies die radiale Auflösung der log-polaren Initialisierung (siehe Abbildung 5.5 (c2)).


Abbildung 5.5: Versuchsserien A und B zur Untersuchung der Wechselwirkung zwischen Hintergrund- und Objektinitialisierung. (a) Rekonstruktion mit regulärer (Pyramide) Hintergrundinitialisierung (Serie A), (b) Rekonstruktion mit irregulärer (Krümmung) Hintergrundinitialisierung (Serie B), (c1) das Original, (c2) die log-polare Initialisierung der Objekte, (c3) die Rekonstruktion der Objekte (ohne Hintergrund), $s$ gibt die Anzahl der Stufen wieder.

| Coderate des Hintergrundes [Bit/Pxl] | Coderate des Objektes [Bit/Pxl] | | | | | |
|---|---|---|---|---|---|---|
| | 0 | 0.001 | 0.005 | 0.024 | 0.099 | 0.407 |
| 0 | - | 7.94 | 8.09 | 8.86 | 8.85 | 9.47 |
| 0.001 | 8.47 | 8.78 | 9.80 | 10.06 | 10.46 | 10.16 |
| 0.005 | 9.49 | 9.89 | 10.22 | 10.54 | 10.45 | 10.87 |
| 0.024 | 12.43 | 12.45 | 12.69 | 12.85 | 13.02 | 12.76 |
| 0.099 | 14.21 | 14.25 | 14.30 | 14.57 | 14.47 | 14.74 |
| 0.407 | 15.41 | 15.41 | 15.43 | 15.44 | 15.56 | 15.67 |

Tabelle 5.5: Die Ergebnisse der Versuchsserie C zur Untersuchung der Wechselwirkung zwischen Hintergrund- und Objektinitialisierung. Es wurden sowohl die Stufen der Hintergrundpyramide als auch die der Objektpyramide von null bis fünf erhöht (dies ist an der Coderate in der ersten Zeile bzw. Spalte zu erkennen). Da es sich um Pyramiden handelt, vervierfacht sich die Rate etwa von Stufe zu Stufe.

Das damit berechnete Feld von 36 Werten wird durch Tabelle 5.5 wiedergegeben.

Das Diagramm 5.6 veranschaulicht dieses Experiment noch einmal graphisch. Die Erhöhung der Coderate für das Objekt bei gleichbleibendem Hintergrund hat kaum einen Effekt auf den Gesamt-$SNR$. Wir verzeichnen bei Abwesenheit des Hintergrundes eine Erhöhung des Gesamt-$SNR$ von der ersten bis zur fünften Stufe der Objektpyramide (1. Zeile in Tab. 5.5) um gerade $1,5$ dB. In der Spalte, in der das Objekt abwesend ist, läßt sich eine Steigerung des Gesamt-$SNR$ um etwa 7 dB erkennen. Diese Ergebnisse sind damit zu erklären, daß das Objekt nur einen geringen Teil des Bildes ausmacht und somit auch nur diesen geringen Teil zum Gesamt-$SNR$ beiträgt.

Greift man sich jedoch zwei Punkte aus diesem Feld heraus, etwa wie im Diagramm Abb. 5.6 markiert, und betrachtet die Rekonstruktionen an diesen Stellen (siehe Abbildung 5.7), so wird deutlich, wie wenig der Gesamt-$SNR$ über den Informationsgehalt eines Bildes aussagt, wenn man das Wiedererkennen von Objekten als Maßstab ansetzt.

## 5.2 Die progressive Rekonstruktion

Nachdem die Bedeutung der attentiven Initialisierung und auch der Hintergrund-Objekt-Beziehung untersucht worden ist, soll noch ein weiterer wichtiger Aspekt der PWA untersucht werden, die progressive Übertragung. Unter progressiver Übertragung haben wir bisher verstanden, daß der Approximationsfehler in jedem Rekonstruktions- bzw. Übertragungsschritt minimal ist. Dies ist eine Folge

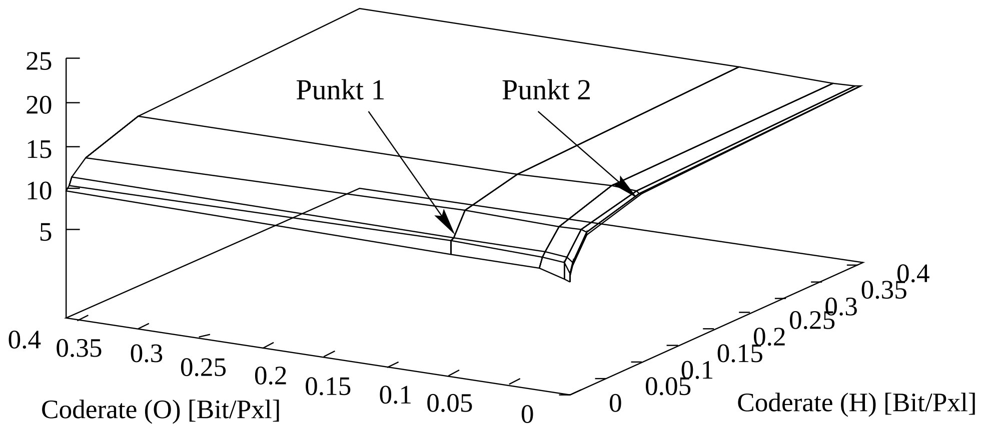



Abbildung 5.6: Die Versuchsserie C zur Untersuchung der Entwicklung des $SNR$ für das Gesamtbild bei einer Variation der Coderate sowohl für den Hintergrund (H) als auch für das Objekt (O). An den Punkten 1 und 2 wurden die Bilder in Abbildung 5.7 rekonstruiert.


Abbildung 5.7: Die Initialisierung des Objektes (hier durch eine dreistufige Pyramide im Hintergrund und eine vierstufige für das Objekt) sowie zwei Rekonstruktionen entsprechend der Graphik aus Abbildung 5.6. (a) am Punkt 1, Initialisierung mit vier Stufen im Hintergrund und einer Stufe für das Objekt, (b) am Punkt 2, Initialisierung mit zwei Stufen im Hintergrund und vier Stufen für das Objekt.

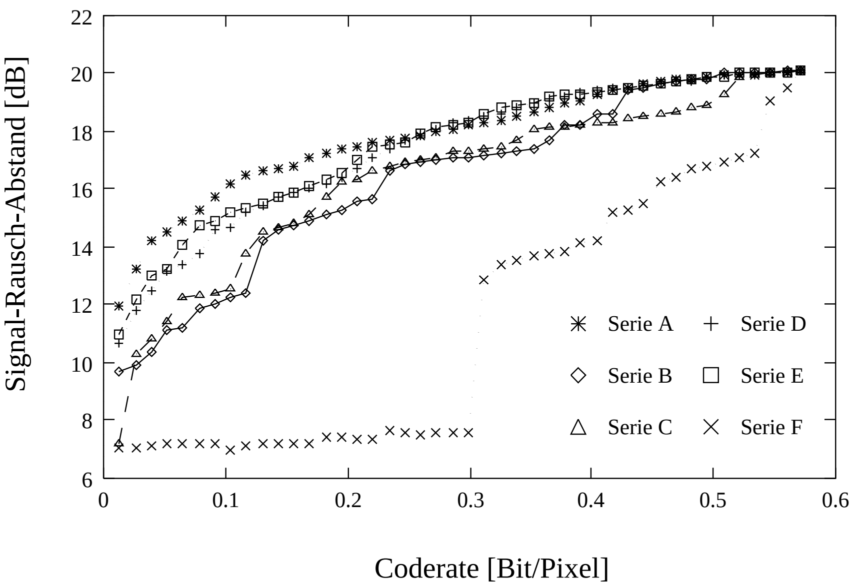



Abbildung 5.8: Die Entwicklung des $SNR$ bei der progressiven Rekonstruktion eines Netzes mit Hintergrund- und Objektinitialisierung unter verschiedenen Progressionsmerkmalen (A - optimal, B - Gewicht, C - regulär, D - $\bar{g}_{s_1 s_2}$, E - $\bar{a}_{s_1 s_2}$, F - zufällig).

der progressiven Optimierung (siehe Kapitel 3). Unsere Netzstruktur ist jedoch so angelegt, daß wir die einzelnen Wavelets beliebig nach einem bestimmten Merkmal, wie z.B. Gewicht oder Skala, sortieren können. In den folgenden sechs Versuchsserien soll die Bedeutung der einzelnen Waveletparameter im Hinblick auf die progressive Übertragung untersucht werden. Es werden jeweils 40 Rekonstruktionsschritte in jeder Serie bei der gleichen Coderate betrachtet und der $SNR$ berechnet. Die erhaltenen Daten sind im Diagramm 5.8 über der Coderate aufgetragen. Jeweils fünf der 40 Rekonstruktionsschritte sind in den Abbildungen 5.9 und 5.10 als Bildsequenz einem visuellen Vergleich zugänglich. Die Serie A gilt bei aller Auswertung als Referenzserie, denn dort wurde das Netz in derselben Reihenfolge rekonstruiert wie optimiert, d.h. diese Versuchsserie entspricht nach unserer Auffassung dem geringsten Approximationsfehler während der Rekonstruktion. Ein Blick in die Graphik bestätigt diese Überlegungen zumindest über weite Strecken der Rekonstruktion. Daß dies im Bereich einer Coderate von $0,25$ [Bit/Pxl] bis $0,4$ [Bit/Pxl] nicht so ist, liegt an dem Kompromiß, den wir mit der Regularisierung und der damit verbundenen Einschränkung des Wertebereichs der Waveletparameter eingehen.

In Serie B wurden die Wavelets des gesamten Netzes unabhängig von ihrer Objektzugehörigkeit absteigend nach dem Betrag der Gewichte sortiert. Wenn man wie bei der Wavelettransformation davon ausgeht, daß die hohen Koeffizienten (Gewichte) hauptsächlich die Träger der Bildinformation sind, ist diese Art der

Übertragung sehr plausibel. In der Graphik erscheint diese Serie über weite Teile der Rekonstruktion an vorletzter Stelle. Betrachtet man jedoch die Bildsequenz in der Abbildung 5.9 (b), so bestätigen sich die eben gemachten Überlegungen bezüglich des Informationsgehaltes hoher Gewichte. Die Augen, die Gesichtsumrandung sowie die Textur im Hintergrund werden als erstes rekonstruiert. Die Serie C zeichnet sich durch eine reguläre Sortierung der Wavelets aus. Unter regulärer zeitlicher Anordnung der Wavelets ist in diesem Zusammenhang folgendes zu verstehen. Wir berechnen zur Abschätzung der Größe eines Wavelets das geometrische Mittel der beiden Skalen.

$$\bar{g}_{s_1 s_2} = \sqrt{s_1 s_2} \tag{5.2}$$

Nun nehmen wir eine Pyramide an, wie sie in Kapitel 2 beschrieben wird und füllen die Stufen dem geometrischen Mittel (5.2) nach mit den Wavelets des Netzes auf. Anschließend werden die Wavelets einer jeden Stufe zeilenweise sortiert. Damit haben wir unserem Netz eine Pyramidenstruktur aufgeprägt, die wir mit regulär bezeichnen wollen. Interessant ist, daß das Netz in unserem Beispiel neun Stufen der angenommenen Pyramide besetzt. Dies bedeutet offenbar, daß die Initialisierung einer Pyramide mit weniger als neun Stufen in diesem Beispiel nicht optimal ist. Diese Serie C rangiert in der Graphik bis zu einer Coderate von etwa $0,38$ [Bit/Pxl] auf dem vierten Platz. In der Bildsequenz 5.9 (c) zeichnet sich diese Serie durch eine zeitige Rekonstruktion des Hintergrundes aus, die Objektdetails im Vordergrund werden relativ spät rekonstruiert. In den Versuchsserien D und E werden die Wavelets nach dem geometrischen Mittel (5.2) (Serie D) bzw. dem arithmetischen Mittel (5.3) (Serie E) sortiert.

$$\bar{a}_{s_1 s_2} = \frac{s_1 + s_2}{2} \tag{5.3}$$

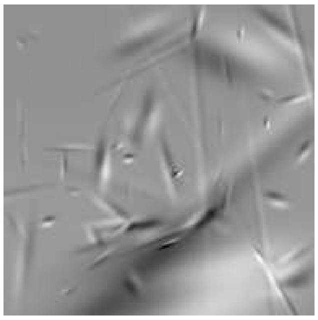
Die Graphen beider Serien liegen dicht beieinander hinter der Serie A. Nur im Bereich einer Coderate von $0,25$ [Bit/Pxl] bis $0,4$ [Bit/Pxl] ist der $SNR$ höher als bei der „optimalen“ Serie A, was seine Ursache in dem oben genannten Regularisierungskompromiß hat. Beim Vergleich der beiden Bildsequenzen in Abbildung 5.10 (a) und (b) zeigt Serie E (arithmetisches Skalenmittel) eine bessere Qualität. Zum Abschluß dieser Experimente wird in Serie F das Waveletnetz zufällig durcheinander gewürfelt, so daß statistisch keines der Merkmale bevorzugt wird. Das Ergebnis dieser Serie ist sehr interessant, denn nachdem sich der $SNR$ über die Hälfte der Rekonstruktion kaum ändert, zeigt der Graph bei einer Coderate von $0,3$ [Bit/Pxl] einen großen Sprung. Dieser Verlauf ist auch der Bildsequenz zu entnehmen. Hier werden die Wavelets völlig unkorreliert rekonstruiert. Diese letzte Serie unterstreicht noch einmal eindrucksvoll, wie wichtig die Wahl eines geeigneten Progressionsmerkmales für die Approximationsgüte ist. Der visuelle

Vergleich der Bildsequenzen der Serien A und B macht darüberhinaus noch einmal deutlich, wie wenig der $SNR$ über die Güte einer Rekonstruktion im Hinblick auf die visuelle Erkennung von Objekten Auskunft gibt.


Abbildung 5.9: Schrittweise Rekonstruktion nach bestimmten Progressionsmerkmalen (a) Serie A - optimal (b) Serie B - Gewicht (c) Serie C - regulär.


a1) CR = 0.115


b1) CR = 0.115


c1) CR = 0.115


a2) CR = 0.229


b2) CR = 0.229


c2) CR = 0.229


a3) CR = 0.344


b3) CR = 0.344


c3) CR = 0.344


a4) CR = 0.458


b4) CR = 0.458


c4) CR = 0.458


a5) CR = 0.572


b5) CR = 0.572


c5) CR = 0.572

Abbildung 5.10: Schrittweise Rekonstruktion nach bestimmten Progressionsmerkmalen (a) Serie D - $\bar{g}_{s_1 s_2}$ (b) Serie E - $\bar{a}_{s_1 s_2}$ (c) Serie F - zufällig.

## 5.3 Das Waveletnetz im Vergleich

Damit ein fairer Vergleich der PWA mit der klassischen Wavelettransformation möglich ist, muß ein Weg gefunden werden, sowohl eine vergleichbare Coderate als auch verschiedene Rekonstruktionszustände und damit den $SNR$ für beide Methoden zu erzeugen. Bei unserer Methode der PWA haben wir die Coderate erzeugt, indem wir die quantisierten Waveletparameter bitweise ausgelesen und als Bytestream abgespeichert haben. Das gesamte Waveletnetz liegt dann als Bytestream vor, dessen Größe die Grundlage für die Berechnung der Coderate liefert. Die einzelnen Zustände der Rekonstruktion erhalten wir durch sukzessive Rekonstruktion der Wavelets. Wählt man die Subband-Codierung (SBC) als Vergleichsmethode, so ist eine ähnliche Vorgehensweise möglich.

### 5.3.1 Die progressive Rekonstruktion bei der Subband-Codierung

Das Wesentliche bei der SBC ist die pyramidenartige Filterung des Bildes mit einem Filterpaar aus Tiefpaß und Hochpaß. Es handelt sich hierbei um $1D$-Filter, deren separable Anwendung in beiden Dimensionen ein erstes Band mit vier Subbändern liefert (siehe Abbildung 5.11). Jedes dieser vier Subbänder trägt die entsprechend gefilterte Bildinformation. Die Anwendung dieser Filter auf das LL-Subband (Tiefpaß in beide Richtungen) liefert ein weiteres Band mit ebenfalls vier Subbändern.

Dieses Verfahren läßt sich bei einer Bildgröße von $N \times N$ $\log N$-mal durchführen. Die anschließende unterschiedliche Quantisierung der Koeffizienten der einzelnen Subbänder führt schließlich zur Irrelevanzreduktion und damit zur Kompression des Bildes. Die Koeffizienten bilden die neue Signalrepräsentation, das Filterpaar ihre Basis. Es ist nun naheliegend, zum Vergleich mit der PWA diese Koeffizienten in gleicher Weise bitweise als Bytestream auszulesen. Damit ist aber genausowenig wie bisher bei der PWA auf die Statistik der Koeffizienten eingegangen worden. Unser Ziel war es, eine möglichst gute Abschätzung der Coderate beider Methoden zu erhalten, damit ein fairer Vergleich möglich ist. Dieses Ziel sollte aber erreicht sein, wenn die Bytestreams beider Methoden mit dem selben Datenkompressionsverfahren, wie z.B. „gzip“, komprimiert werden. Dann nämlich unterliegen beide Codes demselben statistischen Maß, und ein Vergleich der daraus berechneten Coderaten sollte möglich sein. Zur Untersuchung der Entwicklung des $SNR$ während der progressiven Rekonstruktion ist es aber noch notwendig, den $SNR$ zu verschiedenen Rekonstruktionszuständen zu berechnen. Dies haben wir beim WNC durch das Auslesen des Parametervektors eines jeden Wavelets und seiner Rekonstruktion erreicht. Damit ist die Auflösung bei der PWA auf ein Wavelet nach unten begrenzt. Die Methode der SBC läßt es zu, nach dem

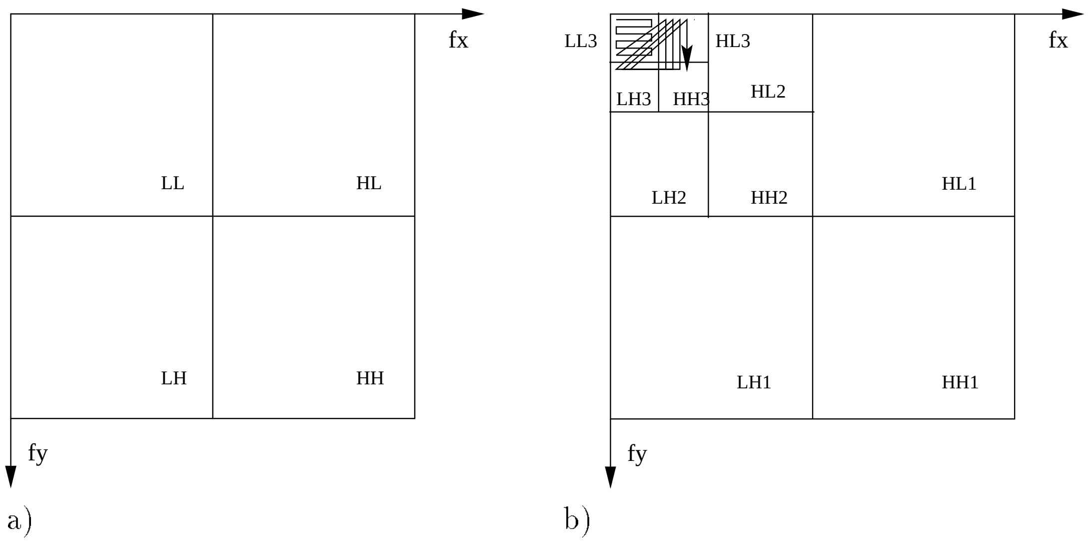



Abbildung 5.11: (a) Einstufige Waveletzerlegung des Bildes liefert vier Subbänder im ersten Band. (b) Dreistufige Waveletzerlegung resultiert in zehn Subbändern, eingezeichnet ist die Rekonstruktionsstrategie der Koeffizienten, die keine Orientierung bevorzugt.

Auslesen eines jeden Koeffizienten zu rekonstruieren, zu klären ist nur in welcher Reihenfolge. An dieser Stelle muß wieder mit etwas Heuristik gearbeitet werden. Die Übertragung muß bandweise, beginnend mit dem höchsten Band, geschehen. Dieses Band besitzt vier Subbänder (LL, HL, LH, HH), jedes weitere drei (HL, LH, HH). Wir beginnen also damit, das LL-Band zeilenweise zu rekonstruieren. Damit desweiteren keine der Orientierungen (HL, LH, HH) bevorzugt wird, lesen wir anschließend jeweils einen Koeffizienten aus jedem dieser Subbänder. Die Abbildung 5.11 veranschaulicht diese Strategie der Rekonstruktion noch einmal. Damit ist ein Weg gefunden, die beiden Methoden direkt am Beispiel miteinander zu vergleichen. Die Coderate des $n$-ten Rekonstruktionsschrittes schätzen wir nach

$$CR_n = n\frac{CR_{total}}{N} \tag{5.4}$$

ab. Dabei ist $CR_{total}$ die Coderate der gesamten Rekonstruktion und $N$ die Anzahl der Rekonstruktionsschritte. Nach diesen Vorüberlegungen soll im nächsten Abschnitt der direkte Vergleich der beiden Methoden vollzogen werden.

| Binzahl | 17 | 15 | 15 | 9 | 9 | 9 | 5 | 3 | 3 | 3 |
|---|---|---|---|---|---|---|---|---|---|---|
| Subband | LL3 | HL3 | LH3 | HH3 | HL2 | LH2 | HH2 | HL1 | LH1 | HH1 |

Tabelle 5.6: Die Quantisierung der einzelnen Subbänder bei der SBC für den Vergleich mit der PWA.

### 5.3.2 Progressive Wavelet-Approximation versus Subband-Codierung

Das schon bei der Untersuchung der progressiven Rekonstruktion verwendete Netz soll auch an dieser Stelle wieder zum Einsatz kommen. Damit werden auch Vergleiche mit früheren Experimenten möglich. gzip verwendet die *Lempel-Ziv-Codierung* (LZ 77) zur Kompression von Daten. Seine Anwendung reduziert die Größe des Netzes noch einmal um etwas mehr als 50 Prozent. Um bei der SBC auf eine etwa gleiche Rate zu kommen, müssen wir uns bei der Quantisierung der einzelnen Subbänder schrittweise an diese Rate herantasten. Wir führen eine Waveletzerlegung auf drei Stufen mit den Filtern $h_0 = \frac{1}{\sqrt{2}}(-1, 1)$, $h_1 = \frac{1}{\sqrt{2}}(1, -1)$, $g_0 = \frac{1}{\sqrt{2}}(1, 1)$, und $g_1 = \frac{1}{\sqrt{2}}(1, 1)$ durch und erhalten auf diese Art und Weise eine geeignete Quantisierung gemäß Tabelle 5.6.

Für den Vergleich unserer Methode mit der SBC führen wir vier Versuchsreihen mit je 40 Rekonstruktionsschritten durch. Es wird wie gehabt der $SNR$ über der Coderate aufgetragen. Das Diagramm in Abb. 5.12 enthält die zugehörige Graphik. In Serie A wird das Bild „Child“ wie im letzten Abschnitt beschrieben über die SBC komprimiert und rekonstruiert. Der zugehörige Graph läßt die Rekonstruktion der drei Bänder deutlich an den Sprüngen erkennen, die eine Folge der unterschiedlichen Quantisierung der drei Bänder sind. Diese Serie beginnt mit dem höchsten $SNR$ und erreicht bereits nach einem Viertel des Rekonstruktionsweges fast das Maximum des $SNR$. Dies hat seine Ursache darin, daß nach elf Schritten die ersten beiden Bänder bereits vollständig rekonstruiert sind. Das letzte Band trägt offenbar kaum noch zur Verbesserung des $SNR$ bei. Die Serien B, C und D entsprechen den Rekonstruktionen unseres Waveletnetzes nach den Progressionsmerkmalen Skala (arithmetisches Mittel), regulär und Gewicht. Den Verlauf dieser Serien kennen wir bereits. Nach dem elften Rekonstruktionsschritt bei einer Coderate von etwa $0,06$ [Bit/Pxl] hat die Serie B mit etwa vier dB den geringsten Abstand zur SBC. Diese Tatsache ist nicht sehr überraschend, denn die SBC ist in diesem Zusammenhang optimal. Da aber, wie wir schon mehrfach erkannt haben, der $SNR$ nicht das alleinige Maß der Approximationsgüte ist, wird der visuelle Vergleich der Rekonstruktionen ein viel interessanteres Unterfangen. Die Abbildungen 5.13 und 5.14 dokumentieren die Rekonstruktionen der Serien A (SBC) und D (PWA mit Progressionsmerkmal Gewicht) zu je 15

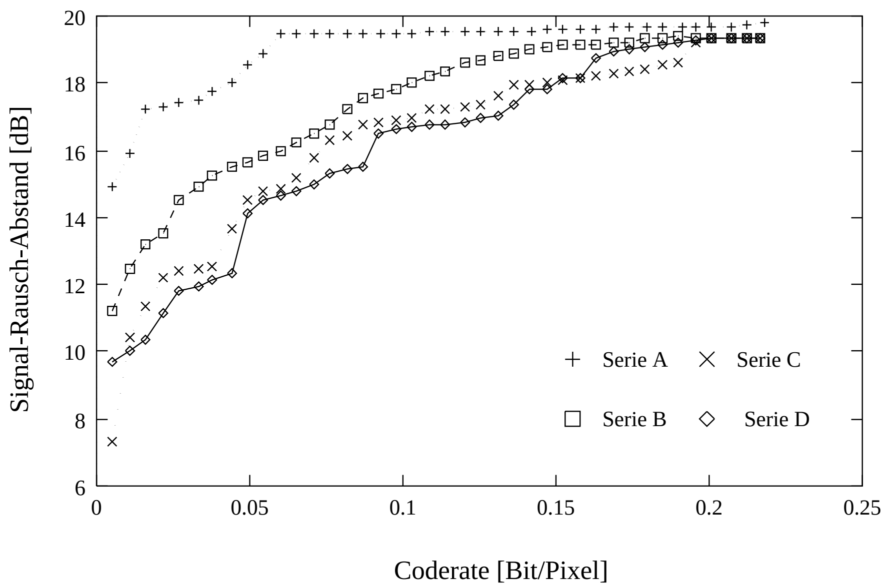



Abbildung 5.12: Der Vergleich der Entwicklung des $SNR$ bei der progressiven Rekonstruktion eines Netzes mit Hintergrund- und Objektinitialisierung bei der klassischen Rekonstruktion des subband-codierten Bildes (Serie A) und bei unterschiedlichen Progressionsmerkmalen der PWA, (Serie B - arithmetisches Skalenmittel, Serie C - regulär und Serie D - Gewicht).

Rekonstruktionsschritten. Die Entwicklung des $SNR$, wie sie aus dem Diagramm in Abb. 5.12 hervorgeht, wird hier bestätigt. Bei der SBC sind bereits nach vier Schritten sowohl der Hintergrund als auch die vermeintlichen Vordergrundobjekte (Augen, Nase, Mund) rekonstruiert. Allerdings ändert sich auch an der Qualität der Rekonstruktion nicht mehr viel. Die Rekonstruktion in Abbildung 5.14 beginnt relativ schlecht, zeigt aber eine ganz andere Qualität. Die typischen Quantisierungsfehler der SBC, hervorgerufen durch die reguläre Abtastung treten hier nicht auf, ebensowenig wie der Effekt der zeilenweisen Rekonstruktion der Bildinformation.


Abbildung 5.13: Schrittweise Rekonstruktion des subband-codierten Bildes. *CR* gibt die Rate in Bit/Pxl an.


Abbildung 5.14: Schrittweise Rekonstruktion des Waveletnetzes über das Progressionsmerkmal Gewicht, $CR$ gibt die Rate in Bit/Pxl an.

# Kapitel 6

# Diskussion und Ausblick

## 6.1 Resumee

Ziel der Arbeit war es, ausgehend von der Grauwertdarstellung, eine aufgabenbezogen progressive Repräsentation von Bildern zu finden. Dabei sollte die in der klassischen Bildkompression vorhandene Zweiteilung der Irrelevanzreduktion in Transformation/Unterabtastung und Quantisierung durch ein Approximationsverfahren mit Waveletnetzen überwunden werden. Dieses Verfahren sollte die die lokal-spektrale Analyseeigenschaft der Wavelettransformation mit der universellen Approximationseigenschaft Neuronaler Netze verbinden, um die freien Parameter der Skala, der Orientierung und des Ortes durch eine geeignete Regularisierung dem Einfluß einer Aufgabenstellung zugänglich zu machen. Durch die Wahl des Optimierungsverfahrens sollte zudem in jedem Rekonstruktionsschritt durch die Minimierung des Approximationsfehlers ein optimaler Informationsgewinn im Sinne einer progressiven Übertragung dieser neuen Signalrepräsentation gewährleistet werden.

Zunächst war es also notwendig, die Signalrepräsentation Waveletnetz genauer zu spezifizieren. Mit der Schaffung einer Struktur, die der der Neuronalen Netze (RBF-Netze) sehr nahe steht, ist uns auch die Parametrisierung eines Waveletnetzes gelungen. Das in der Bildverarbeitung oft verfolgte Konzept der Auflösungshierarchien und Bildpyramiden ermöglichte es uns, eine erste Strategie für die Initialisierung des Netzes zu finden. Damit war die Abschätzung eines ersten Parametersatzes für die Optimierung, d.h. die Anpassung des Netzes an das Signal, möglich. Mit der Levenberg-Marquardt-Methode haben wir einen Algorithmus der Backpropagation-Klasse gefunden, der als Least-Square-Algorithmus die Vorteile von Gradientenabstieg und der Hesse-Methode miteinander verbindet. Ausgestattet mit diesen Werkzeugen sind uns erste Approximationen von einfachen $2D$ Funktionen gelungen. Bei dem Versuch, diese Waveletnetze an beliebige

Grauwertbilder anzupassen, trat zum ersten Mal die Notwendigkeit einer Regularisierung der Waveletparameter zum Vorschein. Dieses Problem wurde durch die zusätzliche Anwesenheit eines, die Wavelets in bestimmte Grenzen von Ort und Skala zwingenden, Potentials während der Optimierung gelöst.

Durch die Variation des Mother Wavelet (Gaußableitung, Gabor) wurde der Einfluß der Wahl des Filters auf die Approximationsgüte deutlich. Die geforderte progressive Rekonstruktion des Bildes aus einem Waveletnetz wird durch eine progressive Optimierung erreicht, die bis dahin durch eine stufenweise Anpassung der Bildpyramide im Sinne einer Grob-Fein-Strategie gekennzeichnet war. Die lokale Eigenschaft einer Waveletanpassung machte jedoch auch die Optimierung einzelner Wavelets einer Stufe plausibel. Diese Strategie senkte zudem die Komplexität der Optimierung erheblich, verlangte aber auch die Spezifikation der Waveletreihenfolge bei der Optimierung. Durch das Konzept der attentiven Initialisierung wurde auch dieses Problem gelöst. Damit war gleichzeitig eine Möglichkeit gegeben, die Aufgabenstellung, unter der die neue Repräsentation gesucht wird, als zusätzliches Wissen mit in die Initialisierung einfließen zu lassen. Das Gaußsche Maß der Grauwertkrümmung war ein Merkmal zur Bestimmung der initialen Waveletpositionen. Es war ein erster Ansatz in dieser Richtung, der bei dem Vergleich mit der regulären Initialisierung über die Bildpyramide sowohl bei der Approximationsgüte als auch bei der progressiven Rekonstruktion unter dem Aspekt einer Aufgabenstellung seine Überlegenheit bewies. Ein zweiter und viel wirkungsvollerer Ansatz gelang mit der Selektion von unter einer Aufgabenstellung interessanten Bildregionen durch den Aufgabensteller selbst. Das hierbei verfolgte und biologisch motivierte Konzept der retina-ähnlichen log-polaren Initialisierung war ein weiterer Schritt hin zu einer kompakten, auf die Aufgabe reduzierten, Repräsentation des Bildes.

Die umfangreiche Untersuchung des Einflusses der Initialisierung des Netzes machte die Bedeutung einer genauen Spezifikation der Aufgabe für eine erfolgreiche Modellierung deutlich. Wie sich in der Auswertung ebenfalls zeigte, macht es die Struktur eines Waveletnetzes möglich, ein und dasselbe Netz und damit ein und dieselbe Repräsentation des Bildes unter verschiedenen Progressionsmerkmalen zu rekonstruieren. Zahlreiche Versuche zu diesem Aspekt haben die Bedeutung der verschiedenen Waveletparameter für die progressive Rekonstruktion vor Augen geführt. Am Schluß der Auswertung stand die Untersuchung, ob unser Verfahren dem Vergleich mit ausgereiften klassischen Techniken der Bildkompression standhält. Bei der Gegenüberstellung mit der Methode der Subband-Codierung (SBC) wurde die Flexibilität, mit der die Aufgabenstellung in die Repräsentation eingebracht werden kann deutlich. Die in Bezug auf den SNR optimale Methode der Subband-Codierung gewährleistet bei geeigneter Rekonstruktionsstrategie in jedem Rekonstruktionsschritt maximalen SNR. Die hier entwickelte Methode der progressiven Wavelet-Approximation (PWA) minimiert dagegen in jedem Rekonstruktionsschritt den Approximationsfehler entsprechend der durch die Regularisierung manifestierten Aufgabenstellung. Wie sich in vielen Versuchen zeigte, resultiert dies nicht in einen maximalen SNR aber in einen visuell (subjektiv) stärkeren Anstieg der Bildqualität bei gleicher Coderate. Die PWA stellt eine neue Qualität in der Rekonstruktion dar. Es gibt aber auch einige Aspekte, die noch nicht ausführlich oder gar nicht untersucht worden sind, weil sie während der Untersuchungen nur am Rand auftauchten und sich nicht unmittelbar als Schritt nach vorn anboten. Deshalb sollen nun mögliche Perspektiven für zukünftige Untersuchungen, die vielleicht auf dieser Arbeit aufbauen, aufgezeigt werden.

## 6.2 Perspektiven

Beginnen wir mit dem Namensgeber des Verfahrens, dem Wavelet, so haben wir bereits festgestellt, daß eine Variation des Filters (hier unter dem Aspekt der Selektivität: Gaußableitung $\leftrightarrow$ Gabor) erheblichen Einfluß auf die Approximationsgüte hat. Hier sind, abhängig von der Aufgabenstellung, Untersuchungen mit Methoden des Filterdesigns angebracht. Auch die Rolle der lokalen Phase bei der Optimierung wurde noch nicht untersucht.

Lassen wir die Beschränkung auf Skalierung und Rotation fallen, so ist auch der Ansatz einer *affinen Deformation* möglich. Der Parametervektor für ein Wavelet nimmt die Form $\boldsymbol{a} = (a_{11}, a_{12}, a_{21}, a_{22}, t_1, t_2, \omega)$ an, wodurch ein zusätzlicher Freiheitsgrad gewonnen wird. Dann wird aber auch eine geeignete Regularisierung der affinen Parameter notwendig, die zunächst eine Interpretation der Parameter verlangt.

Die Optimierung des Netzes ist ein wesentlicher Bestandteil des Verfahrens. Zu untersuchen sind andere Optimierungsalgorithmen als der Levenberg-Marquardt-Algorithmus. Es muß auch eine Möglichkeit gefunden werden, die Komplexität der Optimierung weiter zu senken, um die Methode einer realen Anwendung zugänglich zu machen. Forschungsbedarf besteht vor allem bei dem Attentionsmechanismus, der der Optimierung vorangestellt wird, denn es ist deutlich geworden, wie wichtig die attentive Initialisierung für den Einfluß der Aufgabenstellung ist. Sie bestimmt darüberhinaus die Konvergenz der Optimierung sowie die Approximationsgüte und damit die Qualität der progressive Rekonstruktion. Der Attentionsmechanismus nimmt also bei diesem Verfahren eine Schlüsselrolle ein, die es weiter zu ergründen gilt, um ihr schließlich in der aufgabenbezogenen Repräsentation von Bildern gerecht zu werden.

# Literaturverzeichnis


[1] E. Barth, T. Caelli, and C. Zetzsche. Image encoding, labeling, and reconstruction from differential geometry. *Graph. Models and Image Processing*, 55:428–446, 1993.

[2] Marc Bolduc and Martin D. Levin. A foveated retina system for robotic vision. In *Vision Interface*, Mc Gill University, Canada H3A 2A7, March 1994. Centre for Intelligent Machines and Department of electrical Engineering.

[3] G. Cybenko. Approximation by superposition of a sigmoidal function. *Mathematics of control, signals and systems*, 2:303–314, 1989.

[4] K. Daniilidis. Attentive visual motion processing: computations in the log–polar plane. *Computing*, 11:1–20, 1995. Supplement on Theoretical Foundations of Computer Vision.

[5] I. Daubechies. The wavelet transform, time-frequency lokalisation and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, 1990.

[6] Andreas Galka. Entwurf und Realisierung Neuronaler Netze mit diskreten analogelektronischen Bauelementen. Diplomarbeit, Institut für Angewandte Physik, Christian Albrechts Universität, Kiel, Feb 1994.

[7] Udo Mahlmeister, Harro Pahl, and Gerald Sommer. Color–orientation indexing. In B. Jähne, P. Geißler, H. Haußecker, and F. Hering, editors, *Mustererkennung 1996, 18. DAGM–Symposium*, pages 3–10. Springer–Verlag, 1996. URL: ftp://ftp.informatik.uni-kiel.de/pub/kiel/publications/-CognitiveSystems/Ps_Z/1996_.ps.Z.

[8] Yves Meyer and Robert D. Ryan. *Wavelets, Algorithms and Applikations.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1993.

[9] T. Poggio and F. Girosi. Networks for approximation and learning. *IEEE Trans. PAMI*, 78(9):1481–1497, 1990.

[10] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical recipies in Pascal.* Cambridge University Press, 1989.

[11] O. Rioul and M. Vetterli. Wavelets and signal processing. *IEEE SP Magazine*, pages 15–38, 1991.

[12] F. Rosenblatt. The perceptron: a probalistic model for information storage and organization in the brain. In J. A. Anderson and E. Rosenfeld, editors, *Neurocomputing, Foundations of Research*, pages 92–114. MIT-P, Cambridge, Massachusetts, 1988.

[13] Gerald Sommer, Kostas Daniilidis, Udo Mahlmeister, and Michael Hansen. Skript zur Vorlesung Computer Vision I, August 1994. Lehrstuhl für Kognitive Systeme, CAU zu Kiel.

[14] Brani Vidakovic and Peter Müller. Wavelets for kids, a tutorail introduction. Technical report, Institute of Statistics and Decision Sciences, Duke University Durham, NC 27708-0251, 1991.

[15] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. Neural Networks*, 3(6):889–898, 1992.

# Danksagung

Für die Ermöglichung dieser interdisziplinären Arbeit möchte ich mich bei Herrn Prof. Dr. G. Sommer und Herrn Prof. Dr. G. Pfister bedanken. Mein späterer Betreuer Udo Mahlmeister weckte mein Interesse für dieses Thema während meiner Hospitantentätigkeit in den Semesterferien. Ich möchte mich bei ihm für seine Bereitschaft und für die hilfreichen Diskussionen bedanken. Ferner gilt mein Dank der gesamten Arbeitsgruppe Kognitive Systeme für die angenehme Arbeitsatmosphäre. Für das Korrekturlesen dieser Arbeit und die vielen wertvollen Hinweise bin ich Britta, Michael und Volker sehr dankbar. Meiner Freundin Bianca möchte ich schließlich für ihre große Unterstützung danken, die sehr zum Gelingen dieser Arbeit beigetragen hat.

## Erklärung:

Hiermit versichere ich, daß ich die vorliegende Arbeit selbständig verfaßt und ausschließlich die angegebenen Hilfsmittel und Quellen verwendet habe.

Kiel, den