

Institut für Visualisierung und Interaktive Systeme

Universität Stuttgart  
Universitätsstraße 38  
D-70569 Stuttgart

Bachelorarbeit Nr. 185

## **Variationsansätze zur Struktur-Textur-Zerlegung**

Stephan Albrecht

**Studiengang:** Informatik  
**Prüfer/in:** Prof. Dr. -Ing. Andrés Bruhn  
**Betreuer/in:** Prof. Dr. -Ing. Andrés Bruhn

**Beginn am:** 17. November 2014

**Beendet am:** 5. Mai 2015

**CR-Nummer:** G.1.2, G.1.4, G.1.6, I.2.10, I.4.3



## Kurzfassung

Das Maschinensehen findet heutzutage in vielen Teilen unseres Alltags Anwendung, sei es in der medizinischen Bildverarbeitung oder in der Robotik. Ein Teilgebiet des Maschinensehens ist die Vor- und Nachbereitung von Bildern. Es existieren je nach Zweck unterschiedliche Methoden, die wiederum unterschiedliche Informationen über das Bild benötigen. Viele Methoden nutzen dabei die Struktur und/oder die Textur eines Bildes. Der Begriff Struktur bezeichnet hierbei den homogenen Farbverlauf sowie die Form eines Objektes. Die periodischen oder quasiperiodischen Muster auf diesen Objekten sind als Textur zu verstehen. In dieser Arbeit wird die Zerlegung in Struktur und Textur einerseits als Optimierungsproblem mit verschiedenen Variationsansätzen gestaltet und andererseits mit Hilfe von Mittelungsfiltren errechnet. Das Optimierungsproblem entspricht hierbei der Minimierung eines Energiefunktional, welches sich aus einem Ähnlichkeitsterm und einem Glätte-terme zusammensetzt. Im Rahmen dieser Arbeit galt es herauszufinden, ob es für die Struktur- und Texturerlegung ein dominierendes Verfahren gibt. Dazu wird eine Auswahl an Variationsansätzen als auch der Rolling-Guidance-Filter vorgestellt und untereinander verglichen, um mögliche Vor- und Nachteile der jeweiligen Methoden aufzuzeigen.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>7</b>
1.1	Aufgabenstellung . . . . .	8
1.2	Verwandte Arbeiten . . . . .	8
1.3	Gliederung . . . . .	8
<b>2</b>	<b>Grundlagen</b>	<b>9</b>
2.1	Struktur und Textur . . . . .	9
2.2	Relevanz . . . . .	10
2.3	Rauscherzeugung . . . . .	11
<b>3</b>	<b>Diskrete variationale Methoden: Quadratischer Ansatz</b>	<b>13</b>
3.1	Der eindimensionale Fall . . . . .	13
3.2	Der zweidimensionale Fall . . . . .	14
3.3	Jacobi-Verfahren . . . . .	17
3.4	Gauß-Seidel-Verfahren . . . . .	17
3.5	SOR-Verfahren . . . . .	19
<b>4</b>	<b>Diskrete Variationale Methoden: Subquadratischer Ansatz</b>	<b>23</b>
4.1	Der eindimensionale Fall . . . . .	23
4.2	Der zweidimensionale Fall . . . . .	25
4.3	Lagged-Nonlinearity-Verfahren . . . . .	25
<b>5</b>	<b>Rolling-Guidance-Filter</b>	<b>33</b>
5.1	Bilaterale Filterung . . . . .	33
5.2	Rolling-Guidance-Filter . . . . .	34
<b>6</b>	<b>Evaluation</b>	<b>37</b>
6.1	Entfernen von Rauschen . . . . .	38
6.2	Struktur- und Texturerlegung bei ähnlicher Struktur und Textur . . . . .	45
6.3	Struktur und Texturerlegung bei feiner Textur . . . . .	47
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>49</b>
	<b>Literaturverzeichnis</b>	<b>51</b>



# 1 Einleitung

Im Bereich der Luft- und Satellitenbildauswertung, bis hin zur medizinischen Bildverarbeitung spielt das Maschinensehen eine zunehmende Rolle. Aufgrund dessen gewinnt die Vor- und Nachbereitung von Bildern immer mehr an Relevanz. Hierzu sind Methoden notwendig, die der Entdeckung, Erkennung, Identifikation sowie der Analyse von Objekten und Szenarien dienen. Dabei spielt die Struktur- und Texturzerlegung eine bedeutende Rolle, da die dadurch gewonnenen Informationen für unterschiedliche Methoden des Maschinensehens benötigt werden. Es existieren schon einige bekannte Methoden, wie der bilaterale Filter [TM98] oder der gewichtete Median-Filter [Wei06], die man hinzuziehen könnte, um ein Bild in seinen Struktur- und Texturanteil zerlegen zu können. Begibt man sich in den Bereich der Struktur-Texturzerlegung eines Bildes, tritt zunächst die Frage auf, was Strukturen und Texturen in einem Bild sind und worin deren Unterschiede liegen. In dieser Arbeit bezeichnet der Begriff Struktur den homogenen Farbverlauf sowie die Form eines Objektes. Die periodischen/quasiperiodischen Muster auf diesen Objekten fallen unter den Begriff Textur. Abbildung 1.1 zeigt eine mögliche Zerlegung von einem Bild in dessen Struktur- und Texturanteil.



**Abbildung 1.1:** Struktur- Texturzerlegung eines Bildes. (*oben*): Original. (*unten links*): Strukturanteil. (*unten rechts*): Texturanteil (modifizierte Farbwerte für bessere Sichtbarkeit).

Diese Arbeit soll dem Leser diese Begriffe und ihren Zusammenhang anhand von unterschiedlichen Verfahren zur Struktur-Texturzerlegung eines Grauwertbildes näher bringen. Es handelt sich hierbei um den quadratischen und subquadratischen Variationsansatz wie auch um die Methode des Rolling-

Guidance-Filters, welcher auf dem bilateralen Filter aufbaut. Die Aufgabe vieler Filter beruht auf der Entfernung von ungewollten Details. Dieses Vorgehen kann man auch als Struktur-Texturzerlegung ansehen. Der Strukturanteil ist hierbei gewollt, der Texturanteil beinhaltet die ungewollten Details. Somit ist es von Interesse, wie gut diese Variationsansätze diese Problematik lösen. Letztendlich gilt es herauszufinden, ob es ein dominierendes Verfahren gibt, das stets das beste Ergebnis der Struktur-Texturzerlegung liefert oder, ob je nach Bildtyp, ein bestimmtes Verfahren zu bevorzugen ist.

### 1.1 Aufgabenstellung

Um diese Frage zu beantworten, werden zunächst einige einfache Variationsansätze zur Struktur-Texturzerlegung, als auch der bilaterale Filter und der Rolling-Guidance-Filter implementiert. Im Anschluss daran gilt es, für diese Methoden geeignete Experimente zu entwickeln, damit diese qualitativ und visuell miteinander verglichen werden können.

### 1.2 Verwandte Arbeiten

Schon andere Autoren haben sich mit den Variationsansätzen zur Struktur-Texturzerlegung beschäftigt. Die Arbeit [AGCO06] von Aujol *et al.* fokussiert sich auf unterschiedliche Energiefunktionale für jeweils unterschiedliche Texturen. Die Texturen werden hierbei in drei Klassen eingeteilt: oszillierende Muster (Total Variation(TV)- $L^2$  [ROF92] und  $TV - G$  [ves]), strukturelle Texturen ( $TV - L^1$  [All97, Nik04]) und glatte periodische, vielleicht auch directionale Texturen (TV-Gabor [DH95]). Es wird diskutiert, welche Auswirkungen die verschiedenen Normen auf die Eigenschaften des Energiefunktionals haben und welche Arten von Texturen damit extrahiert werden können.

In [XYXJ12] wird die Extraktion von Struktur aus Textur mit relativer totaler Variation erläutert. Anhand schon bekannter Methoden zur kanten-erhaltenden Glättung, wie bilaterale [DD02] und lokale histogram-basierte Filterung [KS10], wird ein neues Verfahren vorgestellt, welches die Schwächen der anderen Filter kompensiert. Dieses Verfahren findet in der Vektorisierung, der Kantenvereinfachung und Erkennung sowie der komponenten-bewussten Bildskalierung Anwendung.

### 1.3 Gliederung

Die Arbeit ist wie folgt gegliedert:

Das erste Kapitel gibt einen kurzen Einblick in die Thematik und Vorgehensweise dieser Arbeit. Außerdem umfasst es die verwandten Arbeiten und die Gliederung. Das zweite Kapitel beschäftigt sich mit den in dieser Arbeit benötigten Grundlagen, wie auch der Erläuterung von Struktur- und Textur. Im dritten Kapitel wird der quadratische Variationsansatz und im vierten Kapitel der sub-quadratische Variationsansatz vorgestellt. Das fünfte Kapitel erläutert den bilateralen Filter und den darauf aufbauenden Rolling-Guidance-Filter. Die Evaluation mit Vergleich bilden das sechste Kapitel. Abschließend folgt das siebte Kapitel mit der Zusammenfassung und einem Ausblick.

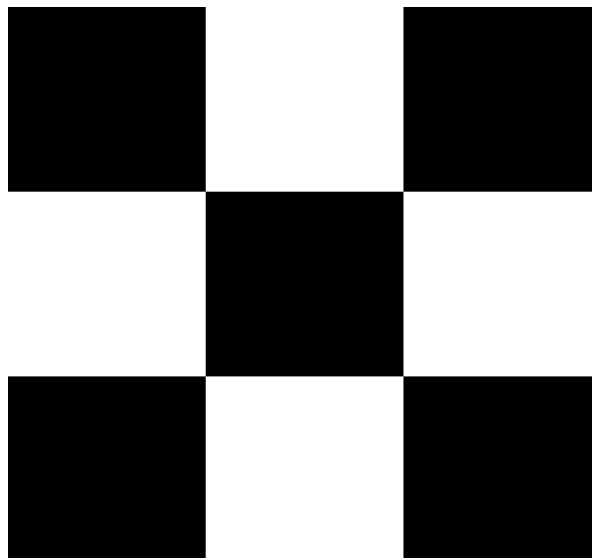


## 2 Grundlagen

In diesem Kapitel werden die für diese Arbeit relevanten Grundlagen der Bildverarbeitung vorgestellt. Die Begriffe Struktur und Textur werden definiert und spezifiziert. Im Anschluss daran erfolgt die Erläuterung der Relevanz der Struktur-Texturerlegung in anderen Bereichen der Bildverarbeitung. Abschließend wird die verwendete Rauscherzeugung erklärt.

### 2.1 Struktur und Textur

Eine Struktur ist grundsätzlich eine Form, die aus einem homogenen Farbverlauf und der Reflexion des Lichts besteht. Im Gegenteil dazu stellt die Textur ein Muster dar, welches sowohl regel- als auch unregelmäßig auftreten kann. Die Begriffe Struktur und Textur unterliegen jedoch im Allgemeinen einem großen Interpretationsspielraum. Dieser beruht einerseits auf dem Blickwinkel des Betrachters und zum Anderen auf der Größe der betrachteten Objekte. Betrachtet man Abbildung 2.1, so ist unentscheidbar, ob fünf schwarze Strukturelemente auf weißem Hintergrund vorliegen oder eine Schachbretttextur vorliegt.



**Abbildung 2.1:** Schachbrettmuster. Hierbei ist unklar, ob es sich um Textur oder Struktur handelt.

Aufgrund dieses Interpretationsspielraumes und der daraus resultierenden Problematik wird versucht, die Definitionen dieser Begriffe hinsichtlich der Verwendung in der Bildverarbeitung einzugrenzen. Hierzu werden die Begrifflichkeiten anhand eines Beispielbildes erläutert. Betrachtet man das folgende Bild 2.2, so sind verschiedene Objekte erkennbar. Das Bild zeigt unter anderem einen Tisch mit einer karierten Tischdecke, auf der ein hölzerner Gegenstand liegt. Wäre die Tischdecke nun nicht kariert, sondern einfarbig, so würde der Mensch trotzdem die Tischdecke erkennen. Dies liegt daran, dass dieses Objekt eine gewisse Form aufweist und zusätzlich auf einem anderen Objekt liegt, welches die Form eines Tisches hat. Solche Formen, die aus einem homogenen Farbverlauf bestehen, werden als Struktur bezeichnet. Der Farbanteil setzt sich hierbei aus der Farbe des Objektes und der Reflexion des Lichts zusammen. Die komplexen Muster in einem Bild hingegen werden in der Bildverarbeitung



**Abbildung 2.2:** Testbild Barbara. Es sind zahlreiche Texturen zu erkennen.

als Textur bezeichnet. Texturen sind lokale Variationen der Farbwerte innerhalb eines Objektes und sind meist periodisch beziehungsweise quasiperiodisch. Bei Vorliegen gewisser Regelmäßigkeiten handelt es sich um periodische Texturen. Ein Beispiel dafür ist das Streifenmuster der Hose der Frau in Abbildung 2.2. Quasiperiodische Texturen sind zum Beispiel Beschriftungen, welche als Verzierung dienen. Da sich ein komplexes Bild aus einem Textur- und Strukturanteil zusammensetzt, lässt sich der Texturanteil durch die Differenz zwischen Originalbild und Strukturteil bestimmen. Dies gilt auch im Umkehrschluss für den Strukturanteil. In anderen Worten: Ein Bild ist in seine Struktur und Textur zerlegbar.

## 2.2 Relevanz

Was nutzt nun die Zerlegung eines Bildes in seine Struktur und Textur? Texturen werden beispielsweise zur Lösung von Korrespondenzproblemen bei der Berechnung des optischen Flusses verwendet.

Das Korrespondenzproblem stellt hierbei die Zuordnung der Pixelbewegung bei der robusten Bewegungsanalyse digitaler Bildfolgen dar [WS99]. Dieser Bereich ist ein zentrales Forschungsthema des Maschinensehens. Ferner wird die Textur, aufgrund ihrer Verformung, zur Bestimmung von Formen hinzugezogen (englisch: shape from texture). Strukturelemente sind bei solchen Verfahren ungewollt und müssen deshalb entfernt werden. Dies findet bei der Herstellung von dreidimensionalen Oberflächen, die in einem Bild dargestellt sind [Alo88], Anwendung. Ein weiteres Anwendungsgebiet ist die Szenenrekonstruktion. Die Szenen werden hierbei anhand der Textur von Objekten rekonstruiert [PT11].

Die Struktur hingegen wird vor allem in dem Forschungsgebiet der Beleuchtung, beispielsweise bei der Detektion von Oberflächen oder Objektkanten, benötigt [PF06, ZK83] (englisch: shape from shading). Texturen sind in diesem Fall ungewollt, da diese die Beleuchtung verfälschen. Objektkanten werden ebenfalls zur Bewegungsanalyse hinzugezogen. Zur Erreichung möglichst guter Resultate in diesen Anwendungsbereichen ist es vor allem wichtig, die optimale Zerlegung eines Bildes errechnen zu können.

## 2.3 Rauscherzeugung

Zum Einstieg in die Texturfilterung liegt der Fokus zunächst auf dem Entfernen von Rauschen in einem Bild. In diesem Fall führt die Mittelung von Pixeln zu einer Verminderung des Rauschens. Um eine Qualitätsmessung durchführen zu können, werden die Bilder in dieser Arbeit zunächst verrauscht und im Anschluss daran wieder entrauscht. Ziel hiervon ist es, das Ergebnis mit dem Originalbild, welches auch als Grundwahrheit bezeichnet wird, zu vergleichen. Das sogenannte Bildrauschen ist die Verschlechterung eines digital aufgenommenen Bildes [Bov10]. Diese Verschlechterung ist auf Störungen in Form von Farb- und Helligkeitsabweichungen zurückzuführen, die aber oft keinen Bezug zum eigentlichen Bildinhalt haben. Im Zuge dieser Arbeit werden verschiedene Struktur- und Texturzerlegungsmethoden verglichen. Das Rauschen wird in den Eingangsbildern simuliert. Dies erfolgt mit Hilfe des additiven Gaußrauschens, welches folgender Verteilung folgt:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) , \quad (2.1)$$

wobei  $\sigma$  die Standardabweichung des Rauschens ist. Mit dem Algorithmus 2.1 kann additives gaußverteiltes Rauschen auf einem Grauwertbild  $f$  simuliert werden. Der Algorithmus berechnet zunächst die zwei gleich-verteilten Variablen  $U$  und  $V$ , deren Werte zwischen 0 und 1 liegen.  $N$  ist die normalverteilte Variable, die mit der Standardabweichung  $\sigma$  multipliziert und auf den Bildpunkt addiert wird.

Abbildung 2.3 zeigt das Ergebnis des implementierten Algorithmus. Da die Grundwahrheit vorliegt, können anhand der so erstellten Bilder Entrauschungs-Methoden getestet und verglichen werden. Je näher das entrauschte Bild an der Grundwahrheit liegt, desto effizienter ist die Methode. Um dies zu messen, betrachtet man den mittleren quadratischen Fehler (Mean Square Error, MSE) sowie den Spitzen-Signal-Rausch-Abstand (Peak Signal to Noise Ratio, PSNR). Ein Beispiel für ein Testbild mit

**Algorithmus 2.1** Erzeugung von gauß-verteilterm Rauschen

---

```

procedure GAUSSIAN_NOISE(out:  $f$ , in:  $\sigma$ , in:  $g$ )
  for all Pixels do
     $U = (\text{float})\text{rand}() / \text{RAND\_MAX};$ 
     $V = (\text{float})\text{rand}() / \text{RAND\_MAX};$ 

     $N = \sqrt{-2 \cdot \ln(U)} \cdot \sin(2 \cdot \pi \cdot V);$ 
     $M = \sqrt{-2 \cdot \ln(U)} \cdot \cos(2 \cdot \pi \cdot V);$ 

     $f[i][j] = g[i][j] + \sigma \cdot N;$ 

  end for
end procedure

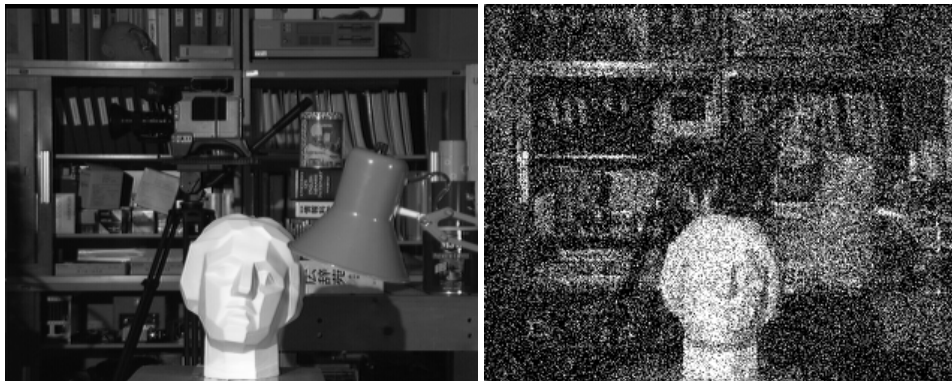
```

---

und ohne Rauschen ist in Abbildung 2.3 gegeben. In der Formel 2.3 stellt  $f$  die verrauschte Version von  $g$ , mit  $g = g_{i,j}; i \in (1 \dots N)$  und  $j \in (1 \dots M)$  dar. Die beiden Fehlermaße lauten:

$$MSE(f, g) = \frac{1}{MN} \sum_{i=1}^N \sum_{j=1}^M (f_{i,j} - g_{i,j})^2, \quad (2.2)$$

$$PSNR(f, g) = 10 \log_{10} \left( \frac{255^2}{MSE(f, g)} \right). \quad (2.3)$$



**Abbildung 2.3:** Links : Original. Rechts : Verrauschtes Bild mit gauß-verteilterm Rauschen, Standardabweichung  $\sigma = 64$ .

### 3 Diskrete variationale Methoden: Quadratischer Ansatz

Bei den folgenden variationalen Methoden handelt es sich um globale Ansätze. Es wird versucht, ein globales Optimalitätskriterium zu erfüllen - in diesem Fall ein zu minimierendes Energiefunktional  $E$ . Hierzu werden Glattheitstransformationen verwendet. Als Eingabe wird ein Eingangssignal sowie ein Regularisierungsparameter benötigt. Das Energiefunktional besteht aus zwei Komponenten, einem Ähnlichkeitsterm und einem Glattheitsterm, d.h.:

$$E = \text{Ähnlichkeit} + \text{Glattheit} . \quad (3.1)$$

Zur Minimierung des Energiefunktionals müssen notwendige Bedingungen diskretisiert werden. Aus dieser Diskretisierung ergibt sich ein Gleichungssystem. Im weiteren Verlauf dieses Kapitels werden Verfahren vorgestellt und verglichen, die dieses Gleichungssystem lösen. Diese sind das Jacobi-Verfahren, das Gauß-Seidel Verfahren und das Überrelaxationsverfahren (Successiv Overrelaxation, SOR-Verfahren). Zur Veranschaulichung wird zunächst der eindimensionale Fall und anschließend der zweidimensionale Fall vorgestellt.

#### 3.1 Der eindimensionale Fall

Um ein besseres Verständnis zu erhalten, betrachte man zunächst die eindimensionalen Signale  $f$ , mit  $f = (f_1, \dots, f_n)$ , und  $u = (u_1, \dots, u_n)$ . Dabei sei  $f$  ein verrauschtes Eingangssignal und  $u$  die gefilterte Version von  $f$ . Das Energiefunktional  $E_f(u)$  besteht dann zu einem Teil aus der Differenz von  $u$  und  $f$  (Ähnlichkeitsterm), zum anderen Teil aus den Differenzen jedes betrachteten Punktes und dessen Nachbarn (Glattheitsterm). Daraus ergibt sich folgende Gleichung für das im vorherigen Abschnitt erläuterte Energiefunktional:

$$E_f(u) = \underbrace{\frac{1}{2} \sum_{k=1}^N (u_k - f_k)^2}_{\text{Ähnlichkeit}} + \underbrace{\frac{\alpha}{2} \sum_{k=1}^{N-1} (u_{k+1} - u_k)^2 + (u_k - u_{k-1})^2}_{\text{Glattheit}} , \quad (3.2)$$

mit  $u_0 = u_1$  und  $u_{N+1} = u_N$ . Der Ähnlichkeitsterm und der Glattheitsterm werden quadratisch bestraft. Der Regularisierungsparameter  $\alpha$  dient als einzustellendes Gewicht für die Glattheit. Je größer der Wert von  $\alpha$ , desto glatter wird das gefilterte Signal. Aufgrund der Konvexität des Energiefunktionals existiert höchstens ein Extremwert, der in diesem Fall ein globales Minimum darstellt.

Dieser Extremwert kann durch Ableiten nach  $u_i$  und durch anschließendes Lösen des resultierenden linearen Gleichungssystems bestimmt werden. Dieses Gleichungssystem ist gegeben durch:

$$\begin{aligned}
 0 &= \frac{\delta E_f}{\delta u_1} = u_1 - f_1 + \alpha(u_1 - u_2) & (i = 1) \\
 0 &= \frac{\delta E_f}{\delta u_i} = u_i - f_i + \alpha(-u_{i+1} + 2u_i - u_{i-1}) & (i = 2 \dots N - 1) \\
 0 &= \frac{\delta E_f}{\delta u_N} = u_N - f_N + \alpha(u_N - u_{N-1}) & (i = N) .
 \end{aligned} \tag{3.3}$$

Die Gleichungen lassen sich in ein Tridiagonalsystem umformen, wie in Abbildung 3.4 dargestellt. Um folgendes System zu erhalten, wurden die Gleichungen nach  $f$  aufgelöst:

$$\begin{pmatrix}
 1 + \alpha & -\alpha & & & & \\
 -\alpha & 1 + 2\alpha & -\alpha & & & \\
 & \ddots & \ddots & \ddots & & \\
 & & & -\alpha & 1 + 2\alpha & -\alpha \\
 & & & & -\alpha & 1 + \alpha
 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N-1} \\ f_N \end{pmatrix} . \tag{3.4}$$

Mithilfe des Thomas-Algorithmus [SK11] kann das aufgeführte System effizient gelöst werden.

## 3.2 Der zweidimensionale Fall

Im zweidimensionalen Fall geht man analog vor, indem man im Glattheitsterm über alle Nachbarn  $\mathcal{N}(i, j)$  des betrachteten Pixels  $(i, j)$  summiert. Dies ist in Gleichung 3.5 dargestellt. Es ist zu beachten, dass bei den Nachbarschaftsbeziehungen sowohl die Hin- als auch die Rückrichtung zu betrachten ist.

$$E_f(u) = \frac{1}{2} \left( \sum_{i=1}^N \sum_{j=1}^M (u_{i,j} - f_{i,j})^2 + \frac{\alpha}{2} \sum_{(k,l) \in \mathcal{N}(i,j)} (u_{i,j} - u_{k,l})^2 \right) . \tag{3.5}$$

Im Gegenzug zum eindimensionalen Signal mit drei Fällen, erhält man beim zweidimensionalen Signal neun Fälle. Es gibt jeweils vier Fälle für die Ecken und Kanten des Bildes und einen für die Mittelpunkte. Zur Verdeutlichung wird in Gleichung 3.6 die Herleitung der Ableitung für den Fall der oberen linken Ecke des Bildes  $u_{1,1}$  bzw.  $f_{1,1}$  gezeigt. Man betrachte dazu die Terme des Energiefunktionals, die den Punkt  $u_{1,1}$  enthalten. Diese lauten:

$$\frac{1}{2} \left( (u_{1,1} - f_{1,1})^2 + \frac{\alpha}{2} \sum_{(k,l) \in \mathcal{N}(1,1)} (u_{1,1} - u_{k,l})^2 \right) . \tag{3.6}$$

Zur Bestimmung des Minimums wird nun die Ableitung gleich 0 gesetzt.

$$\begin{aligned} 0 &= \frac{\delta E_f}{\delta u_{1,1}} = \frac{\delta}{\delta u_{1,1}} \left( \frac{1}{2} \left( (u_{1,1} - f_{1,1})^2 + \frac{\alpha}{2} \sum_{l \in \mathcal{N}(u_{1,1})} (u_l - u_{1,1})^2 \right) \right) \\ &= \frac{\delta}{\delta u_{1,1}} \left( \frac{1}{2} (u_{1,1} - f_{1,1})^2 + \frac{\alpha}{4} \sum_{l \in \mathcal{N}(1,1)} (u_{1,1} - u_{k,l})^2 \right) . \end{aligned}$$

Es erfolgt die Ausschreibung der relevanten Nachbarschaftsbeziehungen und die Vereinfachung der Gleichung.

$$\begin{aligned} &\frac{\delta}{\delta u_{1,1}} \left( \frac{1}{2} (u_{1,1} - f_{1,1})^2 + \frac{\alpha}{4} \sum_{l \in \mathcal{N}(1,1)} (u_{1,1} - u_{k,l})^2 \right) \\ &= \frac{\delta}{\delta u_{1,1}} \left( \frac{1}{2} (u_{1,1} - f_{1,1})^2 + \frac{\alpha}{4} \left( (u_{2,1} - u_{1,1})^2 + (u_{1,1} - u_{2,1})^2 + (u_{1,2} - u_{1,1})^2 + (u_{1,1} - u_{1,2})^2 \right) \right) \\ &= (u_{1,1} - f_{1,1}) + \frac{\alpha}{4} \left( 2(u_{2,1} - u_{1,1}) \cdot (-1) + 2(u_{1,1} - u_{2,1}) + 2(u_{1,2} - u_{1,1}) \cdot (-1) + 2(u_{1,1} - u_{1,2}) \right) \\ &= (u_{1,1} - f_{1,1}) + \frac{\alpha}{2} \left( (u_{1,1} - u_{2,1}) + (u_{1,1} - u_{2,1}) + (u_{1,1} - u_{1,2}) + (u_{1,1} - u_{1,2}) \right) \\ &= (u_{1,1} - f_{1,1}) + \frac{\alpha}{2} \left( -2u_{2,1} + 4u_{1,1} - 2u_{1,2} + \right) \\ &= (u_{1,1} - f_{1,1}) + \alpha(-u_{2,1} + 2u_{1,1} - u_{1,2}) . \end{aligned}$$

Die Gleichungen für die anderen Ecken, Kanten und für die Mitte werden analog gebildet. Gleichung 3.7 zeigt zusammenfassend alle Fälle der Gleichung für ein Bild mit den Abmaßen  $N \times M$ . Die Gleichungen in dieser Abbildung beginnen mit der oberen linken Ecke, gefolgt von der oberen Kante und der oberen rechten Ecke. Danach folgt der mittlere Teil des Bildes, beginnend mit der linken Kante, über die Mitte zur rechten Kante. Zum Schluss werden die Fälle der letzten Reihe aufgezählt und genau so angeordnet wie die der ersten Reihe.

$$\begin{aligned}
 0 &= \frac{\delta E_f}{\delta u_{1,1}} = u_{1,1} - f_{1,1} + \alpha(-u_{2,1} + 2u_{1,1} - u_{1,2}) & (i = 1, j = 1) \\
 0 &= \frac{\delta E_f}{\delta u_{i,1}} = u_{i,1} - f_{i,1} + \alpha(-u_{i-1,1} + 3u_{i,1} - u_{i+1,1} - u_{i,2}) & (i \in [2, N-1], j = 1) \\
 0 &= \frac{\delta E_f}{\delta u_{N,1}} = u_{N,1} - f_{N,1} + \alpha(-u_{N-1,1} + 2u_{N,1} - u_{N,2}) & (i = N, j = 1) \\
 0 &= \frac{\delta E_f}{\delta u_{1,j}} = u_{1,j} - f_{1,j} + \alpha(-u_{1,j-1} + 3u_{1,j} - u_{2,j} - u_{1,j+1}) & (i = 1, j \in [2, M-1]) \\
 0 &= \frac{\delta E_f}{\delta u_{i,j}} = u_{i,j} - f_{i,j} - \alpha(u_{i,j-1} + u_{i-1,j} - 4u_{i,j} + u_{i+1,j} + u_{i,j+1}) & (i \in [2, N-1], j \in [2, M-1]) \\
 0 &= \frac{\delta E_f}{\delta u_{N,j}} = u_{N,j} - f_{N,j} + \alpha(-u_{N,j-1} - u_{N-1,j} + 3u_{N,j} - u_{N,j+1}) & (i = N, j \in [2, M-1]) \\
 0 &= \frac{\delta E_f}{\delta u_{1,M}} = u_{1,M} - f_{1,M} + \alpha(-u_{1,M-1} + 2u_{1,M} - u_{2,M}) & (i = 1, j = M) \\
 0 &= \frac{\delta E_f}{\delta u_{i,M}} = u_{i,M} - f_{i,M} + \alpha(-u_{i,M-1} - u_{i-1,M} + 3u_{i,M} - u_{i+1,M}) & (i \in [2, N-1], j = M) \\
 0 &= \frac{\delta E_f}{\delta u_{N,M}} = u_{N,M} - f_{N,M} + \alpha(-u_{N,M-1} + 2u_{N,M} - u_{N-1,M}) & (i = N, j = M)
 \end{aligned} \tag{3.7}$$

Formt man diese Gleichungen in eine Matrix um, so ist die Struktur der resultierenden Matrix folgendermaßen aufgebaut. Die Matrix ist dünn besetzt, pentadiagonal und symmetrisch:

$$\left( I - \alpha \begin{pmatrix} F_{oben} & I & & & & & \\ & I & F_{mitte} & I & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & I & F_{mitte} & I & \\ & & & & I & F_{unten} & \end{pmatrix} \right) \cdot \begin{pmatrix} u \\ \end{pmatrix} = \begin{pmatrix} f \\ \end{pmatrix} .$$

Es gibt drei unterschiedliche Blöcke. Aufgeteilt, wie in Gleichung 3.8, entspricht die Anzahl der Zeilen und Spalten in einem Block der Anzahl der Pixel in  $x$ -Richtung (Blockgröße:  $N \times N$ ). Insgesamt gibt es  $M \times M$  Blöcke. Die Blöcke für die erste und letzte Pixelreihe ( $F_{oben}$  und  $F_{unten}$ ) entsprechen dem Schema 3.9:

$$F_{oben} = F_{unten} = \begin{pmatrix} -2 & 1 & & & \\ 1 & -3 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -3 & 1 \\ & & & 1 & -2 \end{pmatrix} . \tag{3.9}$$





Sobald beim Gauß-Seidel-Verfahren eine Gleichung gelöst wurde, findet dieser Wert umgehend Verwendung in den nächsten Gleichungen. Das in Kapitel 3.3 vorgestellte Jacobi-Verfahren nutzt lediglich die errechneten Werte des letzten Iterationsschrittes und ist dadurch zeitaufwendiger als das Gauß-Seidel-Verfahren. Da die Lösung einer Gleichung im Gauß-Seidel-Verfahren von den vorherigen Gleichungen abhängt, ist dieses im Gegenzug zum Jacobi-Verfahren nicht gut parallelisierbar. Das iterative Schema des Gauß-Seidel-Verfahrens ist in Gleichung 3.13 dargestellt, wobei  $\mathcal{N}^-(i, j)$  die Menge der Nachbarpixel von  $(i, j)$  beschreibt, welche bereits einen neuen Wert besitzen, und  $\mathcal{N}^+(i, j)$  diejenigen Nachbarpixel, für die noch kein neuer Wert berechnet wurde.

$$u_{i,j}^{(m+1)} = \frac{f_{i,j} + \alpha \sum_{(k,l) \in \mathcal{N}^-(i,j)} u_{k,l}^{(m+1)} + \alpha \sum_{(k,l) \in \mathcal{N}^+(i,j)} u_{k,l}^{(m)}}{1 + \alpha \sum_{(k,l) \in \mathcal{N}(i,j)} 1} . \quad (3.13)$$

Das gleiche Konvergenzkriterium wie für das Jacobi-Verfahren (die Matrix muss symmetrisch und positiv definit sein) gilt für das Gauß-Seidel-Verfahren. Eine bildliche Veranschaulichung der beiden Verfahren zeigt Abbildung 3.1. Von links nach rechts betrachtet wird ein Bild zunächst verrauscht und danach entweder mit dem Gauß-Seidel- oder dem Jacobi-Verfahren bearbeitet. Diese liefern nach einer gewissen Anzahl von Iterationen eine geglättete Version des verrauschten Bildes. Die MSE und PSNR Werte werden dann zwischen dem Original und dem geglätteten Bild berechnet.

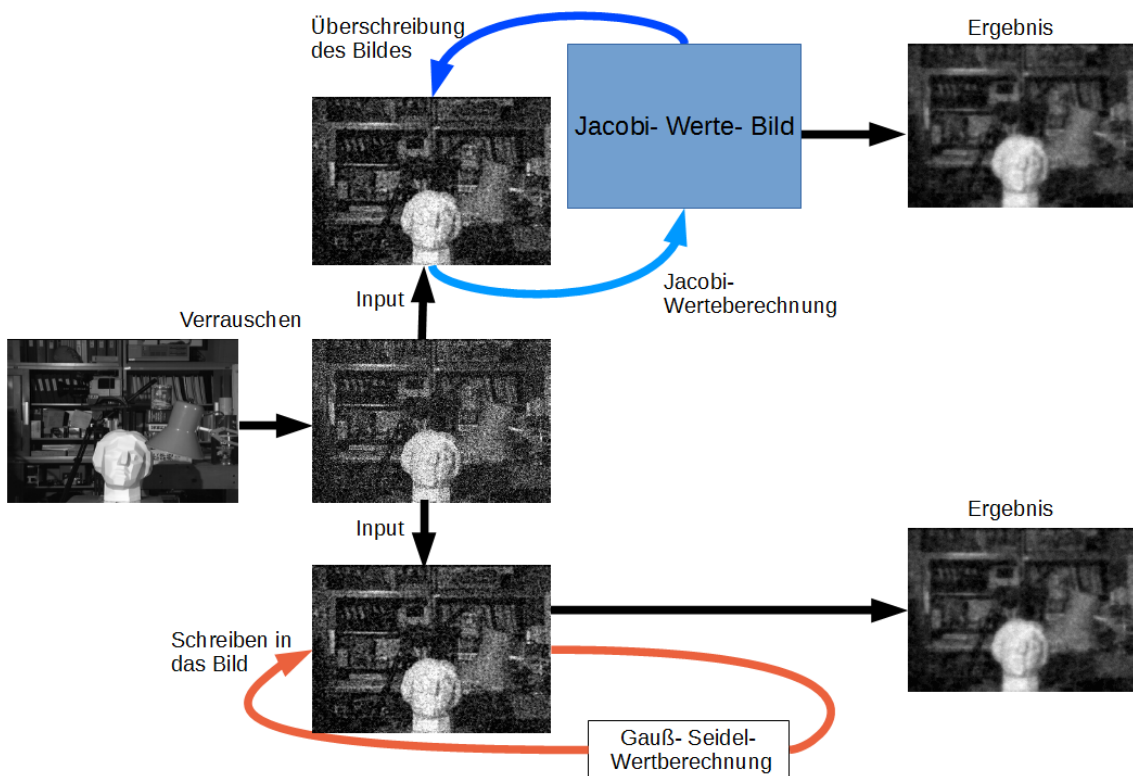


Abbildung 3.1: Anwendung der Verfahren Gauß-Seidel und Jacobi.

Um den zeitlichen Vorteil des Gauß-Seidel-Verfahrens zu verdeutlichen, zeigen die Diagramme 3.2 und 3.3 das Verhältnis zwischen der Iterationszahl und den in den Grundlagen, siehe Kapitel 2, vorgestellten MSE- und PSNR- Werten. Für die Messung wurde der Parameter  $\alpha$  auf 4.216 gesetzt. Bei der nullten Iteration beträgt der MSE-Wert für diese Einstellungen 4091.236 und der PSNR-Wert 12.012 für das berechnete Bild.

Das Diagramm 3.2 zeigt besonders gut, wie sich die näherungsweise Lösungen zur tatsächlichen Lösung einpendeln. Das Pendelverhalten des Jacobi-Verfahrens ist stärker ausgeprägt als des Gauß-Seidel-Verfahrens. Daraus resultiert, dass das Gauß-Seidel-Verfahren schneller den MSE-Wert von 334.917, welcher der MSE-Wert der tatsächlichen Lösung für diesen Versuchsaufbau ist, erreicht. Um diesen Wert zu erzielen, benötigte das Jacobi-Verfahren 133 Iterationen, das Gauß-Seidel-Verfahren hingegen nur 64. Dementsprechend verhalten sich die PSNR-Werte, wie in Diagramm 3.3 dargestellt. Hier liegt die tatsächliche Lösung bei 22.881.

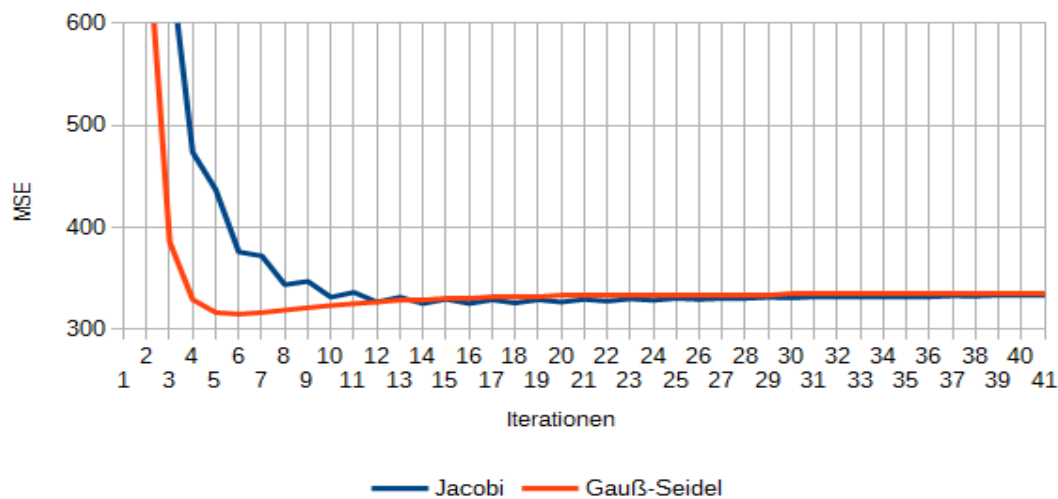


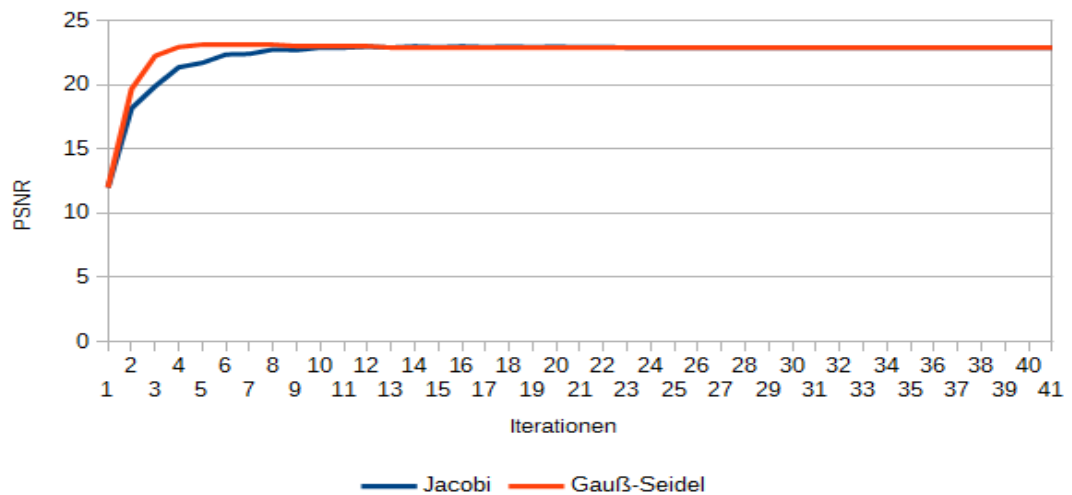
Abbildung 3.2: MSE- Werte des Jacobi- und Gauß-Seidel-Verfahrens mit  $\alpha = 4, 216$ .

### 3.5 SOR-Verfahren

Ein Verfahren, welches noch schneller als das Gauß-Seidel-Verfahren konvergiert, ist das Überrelaxationsverfahren [You14]. Die iterative Formel 3.14 des SOR-Verfahrens besteht punktweise aus den gewichteten Werten der vorherigen Iteration und den gewichteten Gauß-Seidel-Werten. Das Gewicht gibt der Überrelaxationsparameter  $\omega \in [0, 2)$  an. Dieser Parameter sorgt für eine schnellere Konvergenz als bei dem vorher beschriebenen Gauß-Seidel-Verfahren, welches ein Spezialfall dieser Formel mit  $\omega = 1$  ist.

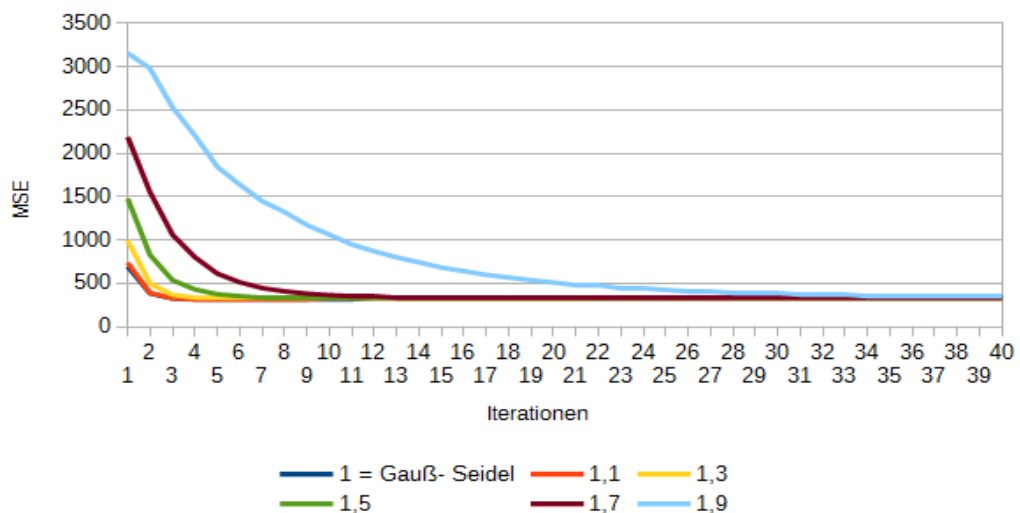
$$u_{i,j}^{(m+1)} = (1 - \omega) \cdot u_{i,j}^{(m)} + \omega \cdot \frac{f_{i,j} + \alpha \sum_{(k,l) \in \mathcal{N}^-(i,j)} u_{k,l}^{(m+1)} + \alpha \sum_{(k,l) \in \mathcal{N}^+(i,j)} u_{k,l}^{(m)}}{1 + \alpha \sum_{(k,l) \in \mathcal{N}(i,j)} 1} \quad (3.14)$$

### 3 Diskrete variationale Methoden: Quadratischer Ansatz



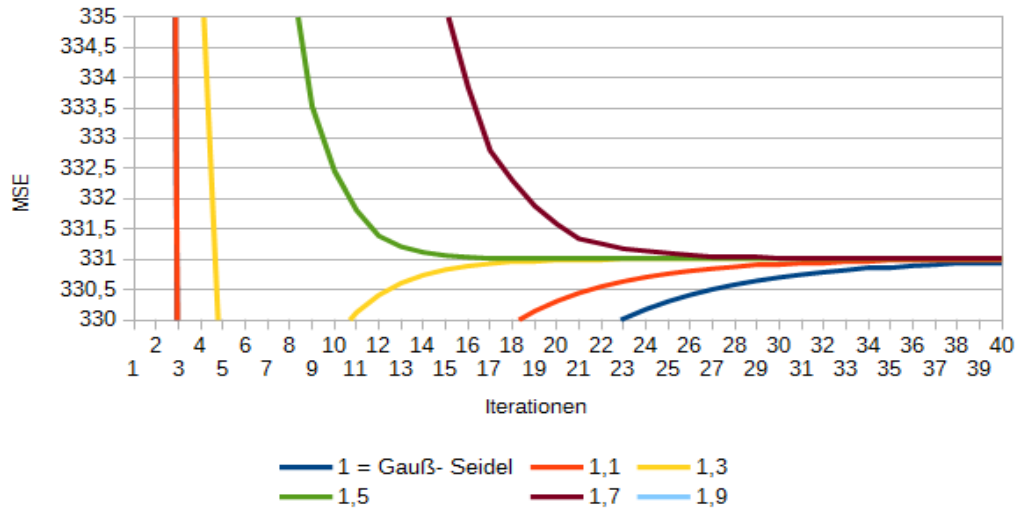
**Abbildung 3.3:** PSNR-Werte des Jacobi- und Gauß-Seidel-Verfahrens mit  $\alpha = 4,216$ .

Die folgenden Diagramme 3.4 und 3.5 sollen verdeutlichen, welche Auswirkungen  $\omega$  auf die Anzahl der Iterationen hat. Auch diesmal wurde für  $\alpha$  der Wert 4,216 gewählt. Um einen möglichst guten und interessanten Überblick über das Intervall  $[0, 2)$  zu bekommen, wurde  $\omega$  auf 1, 1.1, 1.3, 1.5, 1.7 und 1.9 gesetzt. Das Diagramm 3.4 zeigt vor allem, wie sich die  $\omega$ -Werte am Anfang des SOR-Verfahrens verhalten. Es ist zu erkennen, dass je höher der Wert von  $\omega$  gesetzt wird, desto stärker weicht die Näherung von dem tatsächlichen Ergebnis ab. Für  $\omega$  zwischen 1 und 1.7 ist zu sehen, dass die Kurven für die MSE-Werte im Vergleich zum Abstand von den Kurven 1.7 und 1.9 nah beieinander liegen.



**Abbildung 3.4:** MSE-Werte für verschiedene  $\omega$ -Werte des SOR-Verfahrens mit  $\alpha = 4,216$ .

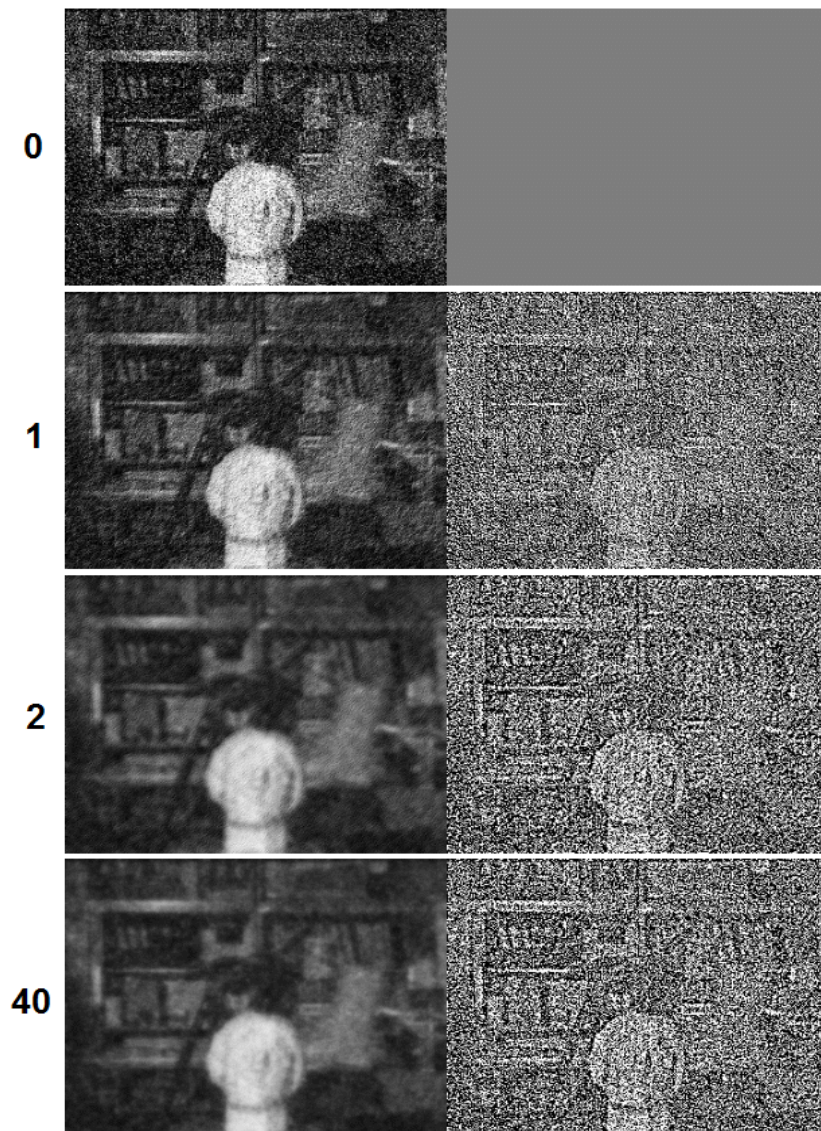
Es stellt sich nun die Frage nach dem optimalen Wert für  $\omega$ . Um dies zu beantworten, zeigt die Grafik 3.5 einen vergrößerten Ausschnitt des Diagramms 3.4 im Bereich des tatsächlichen Ergebnisses 331.015. Es ist zu erkennen, dass im Bereich für  $\omega \in [1.3, 1.5]$  die wenigsten Iterationen benötigt werden. Diese Werte liefern somit die bessere Zeiteffizienz für das SOR-Verfahren für den gewählten Wert von  $\alpha = 4.216$ .



**Abbildung 3.5:** Vergrößerung des Diagramms 3.4 im Bereich des Endergebnisses (330 -335).

Abbildung 3.6 zeigt den Verlauf der Annäherung. In der Grafik sind die ersten drei Iterationen zu sehen, beginnend mit der nullten Iteration, als auch das Ergebnis der vierzigsten Iteration. Zu jeder Iteration ist jeweils ein Bild des Ergebnisses sowie ein Bild der Differenz zwischen Anfangs- und Ergebnisbild erstellt worden. Das Ergebnisbild ist hierbei links, das Differenzbild rechts angeordnet. Bei dem Differenzbild wurde der Nullpunkt auf 127,5 gesetzt, damit auch negative Differenzen gut erkennbar sind. Der Glattheitsparameter  $\alpha$  wurde erneut auf 4,216 gesetzt.

In diesen Bildern ist nicht nur gut zu beobachten, wie das Rauschen sondern ebenfalls Teile der Struktur entfernt werden. Dies ist daran zu erkennen, dass je weiter sich das Bild der Lösung annähert, desto erkennbarer werden beispielsweise die Konturen des Kopfes in dem Differenzbild. Im Gegensatz zur ersten Iteration, bei der die Umrisse des Regals im Differenzbild nur zu erahnen sind, sind diese im Differenzbild der vierzigsten Iteration deutlicher zu erkennen. Betrachtet man nun die Ergebnissseite, verliert das Bild pro Iteration an Schärfe und an Rauschen, bis letztendlich nur ein paar Rauschüberreste zu sehen sind. Diese sind vor allem in den dunkleren Bereichen des Bildes zu erkennen.



**Abbildung 3.6:** Strukturbild (*links*) und Texturbild (*rechts*) der Anwendung des SOR-Verfahrens mit den Iterationsschritten 0,1,2,40 und  $\alpha = 4, 216$ .

## 4 Diskrete Variationale Methoden: Subquadratischer Ansatz

Im quadratischen Ansatz sind keine Ausreißer erlaubt, da diese sehr stark quadratisch bestraft werden. Dies bedeutet, man vertraut den Daten (auch bei Fehlmessungen) und man erlaubt keine Sprünge in der Glattheit (Kanten). Damit sich die Robustheit des Energiefunktionals verbessert, führt man zwei subquadratische Bestrafungsfunktionen ein. Die Bestrafungsfunktionen werden als  $\psi_d$  und  $\psi_g$  bezeichnet. Die Funktion  $\psi_d$  wird auf den Ähnlichkeitsterm angewendet,  $\psi_g$  auf den Glattheitsterm. Die Bestrafungsfunktionen besitzen in dieser Arbeit folgende Form:

$$\psi_d(t^2) = 2\lambda_d^2 \sqrt{1 + \frac{t^2}{\lambda_d^2}} - 2\lambda_d^2 \quad \text{und} \quad \psi_g(t^2) = 2\lambda_g^2 \sqrt{1 + \frac{t^2}{\lambda_g^2}} - 2\lambda_g^2 . \quad (4.1)$$

Durch die Anwendung dieser Funktionen ist das daraus resultierende Gleichungssystem der notwendigen Bedingungen nicht mehr linear. Um diesen Ansatz zu erläutern, wird dieser zunächst beim eindimensionalen Signal eingeführt. Im Anschluss daran wird der zweidimensionale Fall für den subquadratischen Ansatz dargestellt, gefolgt von dem Lagged-Nonlinearity-Verfahrens zur Lösung von nicht linearen Gleichungssystemen.

### 4.1 Der eindimensionale Fall

Zur Veranschaulichung wird erneut das Energiefunktional für ein eindimensionales Signal mit der Erweiterung der subquadratischen Bestrafungsfunktion in Betracht gezogen. Die Formel 4.2 zeigt, an welcher Stelle die oben erwähnten Funktionen in das Energiefunktional integriert werden können.

$$E_f(u) = \frac{1}{2} \sum_{k=1}^N \psi_d((u_k - f_k)^2) + \alpha \sum_{k=1}^{N-1} \psi_g\left(\frac{1}{2}(u_{k+1} - u_k)^2 + \frac{1}{2}(u_k - u_{k-1})^2\right) , \quad (4.2)$$

wobei  $u_0 = u_1$  und  $u_{N+1} = u_N$  ist. Genau so wie für den quadratischen Ansatz müssen auch hier die Ableitungen bestimmt und gelöst werden, um das Minimum des Energiefunktionals zu finden. Um zu verdeutlichen, welche Auswirkungen die Robustheitsfunktionen auf die Ableitungen haben, wird in Gleichung 4.5 die Ableitung für den ersten Punkt berechnet.

$$\begin{aligned}
 0 &= \frac{\delta}{\delta u_1} \left( \frac{1}{2} \psi_d((u_1 - f_1)^2) \right) + \frac{\delta}{\delta u_1} \left( \left( \frac{1}{2} \alpha \psi_g((u_1 - u_2)^2) \right) + \left( \frac{1}{2} \alpha \psi_g \left( \frac{1}{2} (u_1 - u_2)^2 + \frac{1}{2} (u_3 - u_2)^2 \right) \right) \right) \\
 &= \psi'_{d,1}(u_1 - f_1) + \frac{(\psi'_{g,1} + \psi'_{g,2})}{2}(u_1 - u_2)
 \end{aligned} \tag{4.3}$$

mit

$$\begin{aligned}
 \psi'_{d,i} &= \psi'_d((u_i - f_i)^2) \\
 \psi'_{g,i} &= \psi'_g \left( \frac{1}{2} (u_{i+1} - u_i)^2 + \frac{1}{2} (u_i - u_{i-1})^2 \right)
 \end{aligned} \tag{4.4}$$

wobei

$$\begin{aligned}
 \psi'_d(t^2) &= \frac{1}{\sqrt{1 + \frac{t^2}{\lambda_d^2}}} \\
 \psi'_g(t^2) &= \frac{1}{\sqrt{1 + \frac{t^2}{\lambda_g^2}}} .
 \end{aligned} \tag{4.5}$$

Leitet man die anderen Fälle ab, so ergeben sich die in 4.6 aufgezeigten Gleichungen.

$$\begin{aligned}
 0 &= \frac{\delta}{\delta u_1} = \psi'_{d,1}(u_1 - f_1) + \alpha \frac{(\psi'_{g,1} + \psi'_{g,2})}{2}(u_1 - u_2) && \text{für } i = 1 \\
 0 &= \frac{\delta}{\delta u_i} = \psi'_{d,i}(u_i - f_i) + \alpha \frac{(\psi'_{g,i} + \psi'_{g,i-1})}{2}(u_i - u_{i-1}) + \alpha \frac{(\psi'_{g,i} + \psi'_{g,i+1})}{2}(u_i - u_{i+1}) && \text{für } i \in [2, N-1] \\
 0 &= \frac{\delta}{\delta u_N} = \psi'_{d,N}(u_N - f_N) + \alpha \frac{(\psi'_{g,N} + \psi'_{g,N-1})}{2}(u_N - u_{N-1}) && \text{für } i = N
 \end{aligned} \tag{4.6}$$

Im Allgemeinen mit  $u_0 = u_1$  und  $u_{N+1} = u_N$  gilt :

$$0 = \psi'_{d,i} \cdot (u_i - f_i) + \alpha \left( \frac{\psi'_{g,i-1} + \psi'_{g,i}}{2} \right) \cdot (u_i - u_{i-1}) + \alpha \left( \frac{\psi'_{g,i+1} + \psi'_{g,i}}{2} \right) \cdot (u_i - u_{i+1}) . \tag{4.7}$$

Es findet damit eine Mittlung der  $\psi'_g$ -Werte des betrachteten Punktes und seinen Nachbarn statt. Die Anwendung der subquadratischen Bestrafungsfunktionen hat zur Folge, dass kein lineares Gleichungssystem mehr entsteht, da die Gewichte  $\psi'_d$  und  $\psi'_g$  nichtlinear in  $u$  sind.



## 4.2 Der zweidimensionale Fall

Analog wie zum eindimensionalen Fall wird die allgemeine Gleichung erweitert zu:

$$E_f(u) = \frac{1}{2} \sum_{i=1, j=1}^{N, M} \psi_d((u_{i,j} - f_{i,j})^2) + \frac{\alpha}{2} \psi_g \left( \sum_{(k,l) \in \mathcal{N}(i,j)} \frac{1}{2} (u_{k,l} - u_{i,j})^2 \right) \quad (4.8)$$

mit der allgemeinen Ableitung

$$0 = \psi'_{d,i} \cdot (u_{i,j} - f_{i,j}) + \alpha \sum_{(k,l) \in \mathcal{N}(i,j)} \left( \frac{(\psi'_{g,i,j} + \psi'_{g,k,l})}{2} \cdot (u_{i,j} - u_{k,l}) \right) \quad (4.9)$$

für  $i \in [1, N]$  und  $j \in [1, M]$ . Würde man für die  $\psi$ 's die Funktion  $\psi(t^2) = t^2$  verwenden, bei der die Ableitung  $\psi'(t^2) = 1$  ist, so wäre das resultierende Gleichungssystem linear. Verwendet man nun die vorher vorgestellten Funktionen für die  $\psi$ 's, so erhält man ein nichtlineares Gleichungssystem, da die Ableitungen der  $\psi$ 's nicht linear in  $u$  sind.

## 4.3 Lagged-Nonlinearity-Verfahren

Zur Lösung von nichtlinearen Gleichungssystem ziehen wir uns auf den linearen Fall zurück. Ein Verfahren, welches nichtlineare Gleichungen mit Hilfe von linearen Lösern löst, ist das Lagged-Nonlinearity-Verfahren [KNPS68, Net89, CM99]. Es wird immer dann angewendet, wenn aufgrund der Nichtlinearität weder das Gauß-Seidel noch das SOR-Verfahren direkt angewendet werden können. Das Verfahren führt eine Fixpunktiteration durch, indem es das nichtlineare Gleichungssystem in eine Serie von linearen Gleichungssystemen für feste Werte von  $\psi'_d$  und  $\psi'_s$ , unterteilt:

$$0 = \psi'^m_{d,i,j} (u^{m+1}_{i,j} - f_{i,j}) + \alpha \sum_{(k,l) \in \mathcal{N}(i,j)} \left( \frac{(\psi'^m_{g,i,j} + \psi'^m_{g,k,l})}{2} \cdot (u^{m+1}_{i,j} - u^{m+1}_{k,l}) \right) . \quad (4.10)$$

Die so erstellten Gleichungen können dann mit dem SOR-Verfahren gelöst werden, da die Matrizen symmetrisch und positiv definit sind. Daraus resultiert der folgende Algorithmus 4.1 für den subquadratischen Ansatz. Durch die Variable *Iter* erhält die Prozedur die Anzahl der durchzuführenden Iterationen. Des weiteren erhält sie die Einstellungsvariablen  $\lambda_d$  und  $\lambda_g$  sowie das Eingangsbild  $f$  und das zu berechnende Bild  $u$ .

Die folgenden Bilderserien zeigen die Auswirkungen der verschiedenen Parameter  $\alpha$ ,  $\lambda_d$  und  $\lambda_g$ . Durch die Umstellung der einzelnen Parameter lassen sich unterschiedlich starke Ausprägungen des Texturbildes bzw. des Strukturbildes einstellen. Der Parameter  $\lambda_g$  reguliert die Kantenerhaltung. Je höher dieser Wert, desto mehr Kanten werden durch die Glättung beeinflusst. Ab einem hohen Wert, von beispielsweise 100, verhält sich der Glattheitsterm wie der des quadratischen Ansatzes.  $\lambda_d$  beeinflusst die Bestrafung der Ähnlichkeit von  $f$  und  $u$ . Je höher  $\lambda_d$  gewählt wird, desto ähnlicher müssen  $f$  und  $u$  sein. Um einen Vergleich zwischen den Werten zu erzielen, wurde für die Bilder

### Algorithmus 4.1 Subquadratischer Ansatz

---

```
procedure SUBQUADRATISCHER_ANSATZ(in:  $f$ , out:  $u$ , in: Iter, in:  $\lambda_d$ , in:  $\lambda_g$ )  
  for  $m = 0; k < \text{Iter}; m ++$  do  
    Compute_Psi_d_prime( $f, u, \text{Psid\_prime}, \lambda_d$ );  
    Compute_Psi_g_prime( $f, u, \text{Psig\_prime}, \lambda_g$ );  
  
    Solve_SOR( $f, u, \text{Psig\_prime}, \text{Psid\_prime}$ )  
  end for  
end procedure
```

---

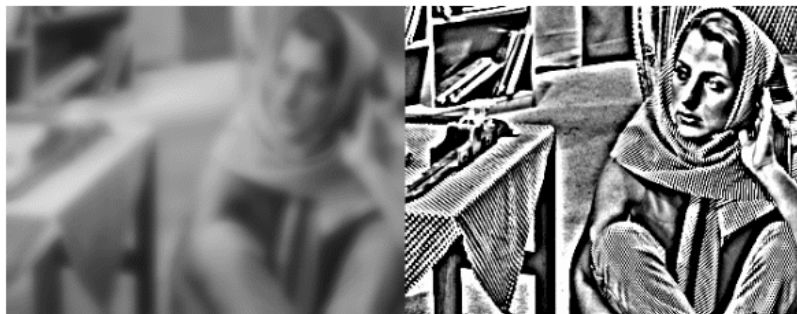
ein fester  $\omega$ -Wert von 1.5 festgelegt. In allen Bilderserien befinden sich auf der linken Seite jeweils die Strukturbilder und auf der rechten Seite die resultierenden Texturbilder. Die Bilderserien 4.1 und 4.2 zeigen das Bildverhalten bei verschiedenen Einstellungen des  $\alpha$ -Wertes bei großem  $\lambda_g$  Wert und kleinem  $\lambda_d$ -Wert. Wie unschwer zu sehen ist, werden mit höheren  $\alpha$ -Werten sehr starke Texturbilder erzeugt. Durch einen  $\alpha$ -Wert von 23 wird das Strukturbild schon so stark geglättet, dass Objekte ihre Form verlieren oder so verzerrt werden, dass sie nicht mehr erkennbar bzw. verloren sind. Die Bilderserien 4.3 und 4.4 zeigen die Auswirkungen von einem kleinen Wert für  $\lambda_g$ . Deutlich ist zu erkennen, dass die Strukturbilder schärfer sind als die Strukturbilder der Ergebnisbildserien 4.1 und 4.2. Dies liegt daran, dass ein Großteil der Kanten trotz der Glättung erhalten werden. Die Bilderserie 4.5 zeigt den Einfluss des  $\lambda_d$ -Wertes. Da dieser den Ähnlichkeitsterm beeinflusst, werden die Strukturbilder durch dessen Erhöhung wieder schärfer, wogegen die Texturbilder schwächer werden. Der Term  $\lambda_d$  wirkt den Werten  $\alpha$  und  $\lambda_s$  entgegen.



(a)

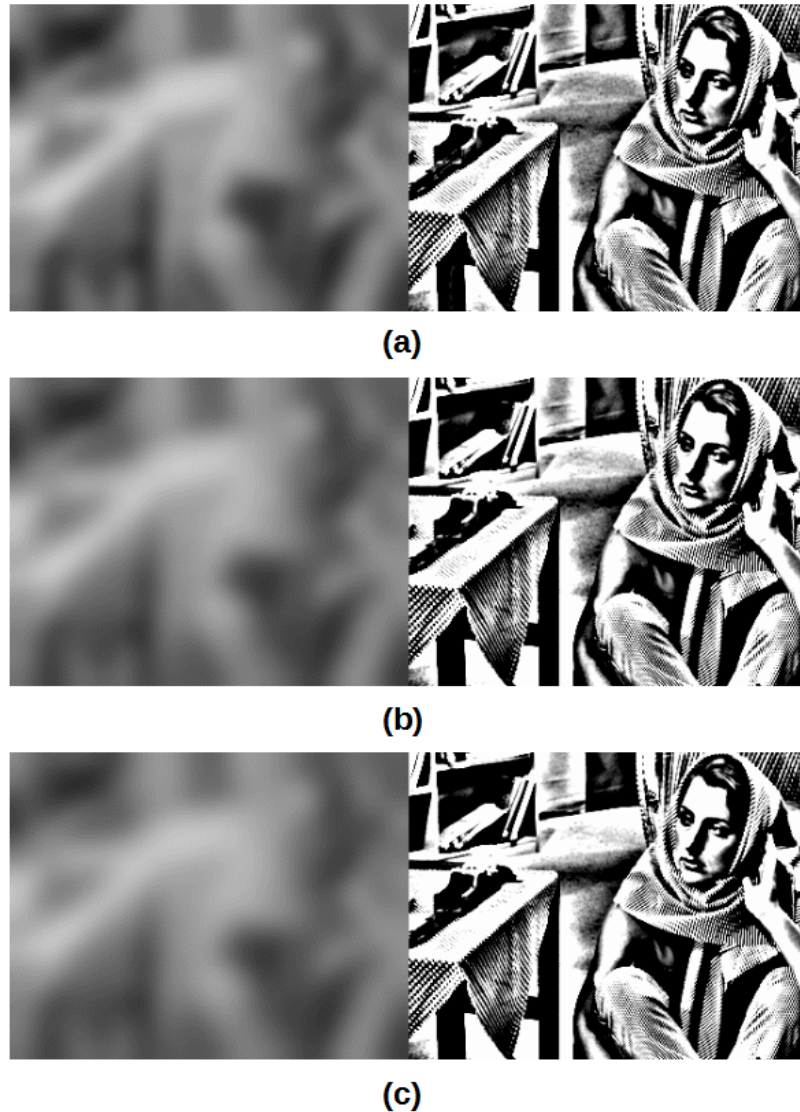


(b)



(c)

**Abbildung 4.1:** Strukturbild (*links*) und Texturbild (*rechts*) des subquadratischen Variationsansatzes mit einer Iteration und hohem  $\lambda_g$ -Wert für verschiedene  $\alpha$ -Werte. Modellparameter: (a)  $\alpha = 1$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . (b)  $\alpha = 4$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . (c)  $\alpha = 23$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . Numerische Parameter:  $\omega = 1.5$ , Iter = 1, SOR-Iter = 10.



**Abbildung 4.2:** Strukturbild (*links*) und Texturbild (*rechts*) des subquadratischen Variationsansatzes mit fünf Iterationen und hohem  $\lambda_g$ -Wert für verschiedene  $\alpha$ -Werte. Modellparameter: (a)  $\alpha = 1$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . (b)  $\alpha = 4$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . (c)  $\alpha = 23$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 100$ . Numerische Parameter:  $\omega = 1.5$ , Iter = 5, SOR-Iter = 10.



(a)



(b)



(c)

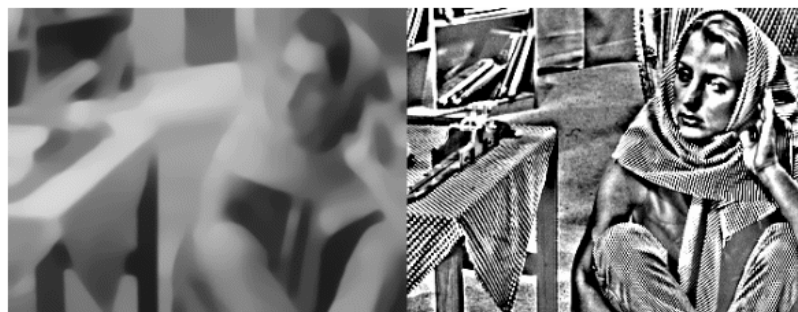
**Abbildung 4.3:** Strukturbild (*links*) und Texturbild (*rechts*) des subquadratischen Variationsansatzes mit einer Iteration und kleinem  $\lambda_g$ -Wert für verschiedene  $\alpha$ -Werte. Modellparameter: (a)  $\alpha = 1$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . (b)  $\alpha = 4$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . (c)  $\alpha = 23$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . Numerische Parameter:  $\omega = 1.5$ , Iter = 1, SOR-Iter = 10.



(a)



(b)



(c)

**Abbildung 4.4:** Strukturbild (*links*) und Texturbild (*rechts*) des subquadratischen Variationsansatzes mit fünf Iterationen und kleinem  $\lambda_g$ -Wert für verschiedene  $\alpha$ -Werte. Modellparameter: (a)  $\alpha = 1$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . (b)  $\alpha = 4$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . (c)  $\alpha = 23$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . Numerische Parameter:  $\omega = 1.5$ , Iter = 5, SOR-Iter = 10.



(a)



(b)



(c)

**Abbildung 4.5:** Strukturbild (*links*) und Texturbild (*rechts*) des subquadratischen Variationsansatzes für verschiedene  $\lambda_d$ -Werte. Modellparameter: (a)  $\alpha = 4$ ,  $\lambda_d = 0.1$ ,  $\lambda_g = 0.1$ . (b)  $\alpha = 4$ ,  $\lambda_d = 0.5$ ,  $\lambda_g = 0.1$ . (c)  $\alpha = 4$ ,  $\lambda_d = 0.9$ ,  $\lambda_g = 0.1$ . Numerische Parameter:  $\omega = 1.5$ , Iter = 5, SOR-Iter = 10.





# 5 Rolling-Guidance-Filter

## 5.1 Bilaterale Filterung

Um den Rolling-Guidance-Filter zu verstehen, soll zuerst ein Blick auf die zugrunde liegende Idee geworfen werden. Dies ist der bilaterale Filter [TM98]. Der bilaterale Filter gehört zu den nicht linearen Filtern und dient hauptsächlich der Weichzeichnung von Bildern unter Beibehaltung der Kantenschärfe mittels gewichteter Mittelung von Pixeln in der Nachbarschaft. Um dies zu ermöglichen, werden zwei Gewichtsmasken benötigt. Die erste Gewichtsmaske wird benötigt, um den Farbwert anhand des Pixelabstands zu gewichten. Dies wird durch die folgende Funktion erreicht:  $w(|x_i - x_j|^2)$ . Die zweite Gewichtsmaske wird zur Berechnung der Farbwerte anhand der Farbdifferenz benötigt und durch die folgende Funktion dargestellt:  $g(|f_i - f_j|^2)$ . Häufig wird für  $w$  und  $g$  die Gauß-Funktion

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad (5.1)$$

mit der Standardabweichung  $\sigma_s$  und  $\sigma_r$  verwendet. Im weiteren Verlauf dieses Kapitels bezeichnet  $(k, l) \in \mathcal{B}(i, j)$  alle Pixel  $(k, l)$ , die in dem Bereich der Gewichtsmaske mit dem Mittelpunkt  $(i, j)$  liegen. Der bilaterale Filter eignet sich sehr gut, da er unter anderem auch das Maximum-Minimum-Kriterium erfüllt. Das bedeutet, dass ein errechneter Wert nicht größer als der maximale Wert oder kleiner als der minimale Wert ist. Die Iterationsvorschriften unterschiedlicher bilateraler Filter werden in den Gleichungen 5.2 bis 5.4 dargestellt.

**Bilateraler Filter:**

$$J_{i,j}^{new} = \frac{1}{K_{i,j}} \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i, j)^T - (k, l)^T|) w(|J_{i,j}^{old} - J_{k,l}^{old}|) J_{k,l}^{old} \quad (5.2)$$

$$\text{mit : } K_{i,j} = \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i, j)^T - (k, l)^T|) w(|J_{i,j}^{old} - J_{k,l}^{old}|)$$

Wendet man diesen Filter iterativ auf sich selbst an, so erhält man den iterativen bilateralen Filter:

**Iterativer Bilateraler Filter:**

$$J_{i,j}^{t+1} = \frac{1}{K_{i,j}} \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i, j)^T - (k, l)^T|) w(|J_{i,j}^t - J_{k,l}^t|) J_{k,l}^t \quad (5.3)$$

$$\text{mit : } K_{i,j} = \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i, j)^T - (k, l)^T|) w(|J_{i,j}^t - J_{k,l}^t|)$$

Nutzt man nun ein anderes Bild, ein sogenanntes Guidance-Bild  $G$ , zur Berechnung der Farbgewichte für die Filterung eines Eingangsbildes  $I$ , so erhält man den gemeinsamen bilateralen Filter.

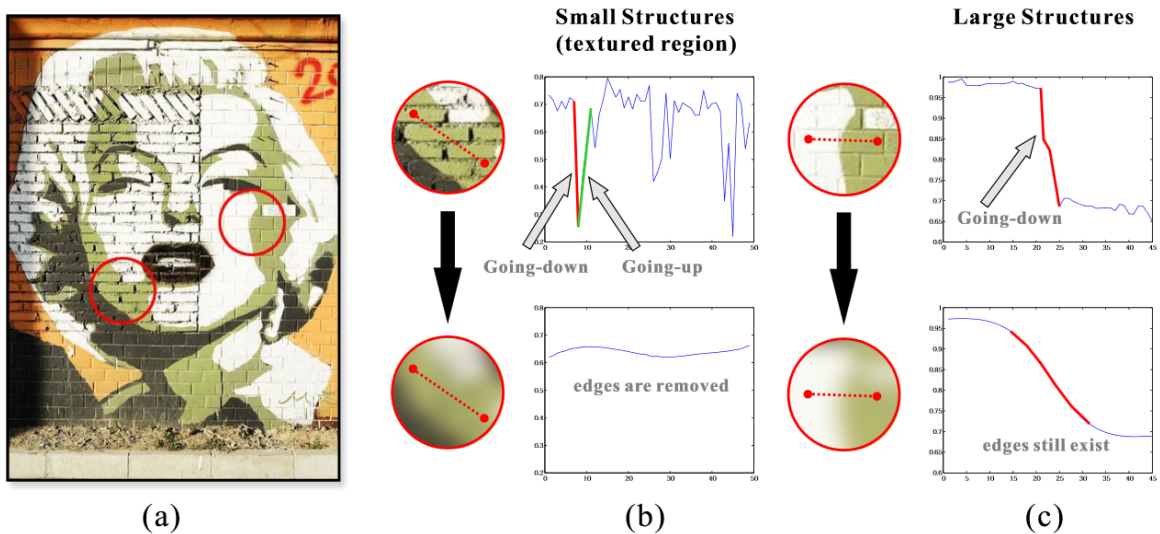
**Gemeinsamer Bilateraler Filter:**

$$J_{i,j}^{new} = \frac{1}{K_{i,j}} \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i,j)^T - (k,l)^T|) w(|G_{i,j} - G_{k,l}|) I_{k,l} \quad (5.4)$$

mit :  $K_{i,j} = \sum_{(k,l) \in \mathcal{B}(i,j)} g(|(i,j)^T - (k,l)^T|) w(|G_{i,j} - G_{k,l}|)$

**5.2 Rolling-Guidance-Filter**

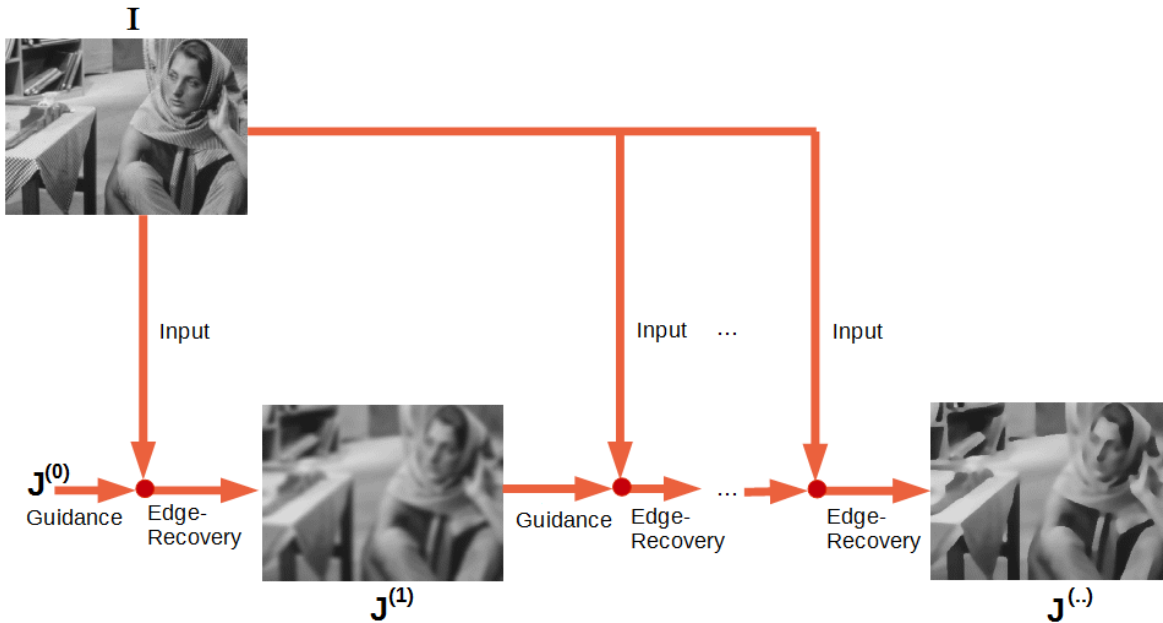
Der Rolling-Guidance-Filter [ZSXJ14] basiert auf der Idee, dass Kanten bei der Glättung erhalten bleiben, wie bei der bilateralen Filterung, je nach Größe der gewählten Textur. Der Filter arbeitet mittels der Gauß-Glättung und bilateralen Filtern und verknüpft diese, um das resultierende Bild schnell und effizient zu berechnen. Die in dieser Arbeit implementierte Methode ist die direkte Implementation der Formel 5.6. Der Grundgedanke des Rolling-Guidance-Filters beruht auf der Texturanalyse mittels Funktionen.



**Abbildung 5.1:** Vergleich der Ergebnisse von kleinen und großen Strukturen nach einer Gaußfilterung. (a): Eingangsbild. (b)-(c): 1D-Signale von Pixelwerten von zwei Geraden im Bild. Die oberen sind die der Eingangssignale, die unteren die gaußgefilterten Signale.[ZSXJ14]

Die resultierenden Graphen zeigen gewisse Ausschläge, welche hierbei als Texturunterschiede dargestellt werden. Glatte Kurvenverläufe deuten auf geringe Farbverlaufänderungen hin. Kleine oszillierende Kurvenverläufe werden herausgemittelt 5.1 (b). Große Sprünge wie in 5.1 (c) deuten auf eine Kante zwischen zwei großen Strukturen. Diese Verläufe werden genutzt, um zu zeigen, dass die

Glättung der Kanten trotz allem die Kanten erhält. Diese Erkenntnis wird verwendet, um nach der Glättung die Kanten wieder herzustellen. Das folgende Flowchartbild des Rolling-Guidance-Filters beschreibt schematisch das Vorgehen. Zu Beginn wird die kleinere Struktur mittels eines Gauß-Filters



**Abbildung 5.2:** Ablaufdiagramm des Rolling-Guidance-Filters mit dem mit Nullen initialisiertem Bild  $J^{(0)}$ .

entfernt. Hierbei wird die Standardabweichung  $\sigma_s$  verwendet, um die Strukturen zu entfernen, welche kleiner als  $\sigma_s$  sind. Nach diesem Schritt werden die Kanten iterativ im Guidance-Bild mittels des Eingangsbildes  $I$  wiederhergestellt. Hierzu werden beide Bilder (Eingangsbild und Guidance-Bild) in einem gemeinsamen bilateralen Filter zusammengefügt:

$$G_{i,j}^{m+1} = \frac{1}{K_{i,j}} \sum_{(k,l) \in \mathcal{B}(i,j)} g(\|(i,j)^T - (k,l)^T\|) w(\|G_{i,j}^m - G_{k,l}^m\|) I_{k,l} \quad (5.5)$$

mit :  $K_{i,j} = \sum_{(k,l) \in \mathcal{B}(i,j)} g(\|(i,j)^T - (k,l)^T\|) w(\|G_{i,j}^m - G_{k,l}^m\|)$

Jetzt ist man allerdings an dem resultierenden Guidance-Bild interessiert. Aus diesem Grund wird zum Schluss das neu berechnete Bild auf das Guidance-Bild gesetzt.

$$J^{new} = G^m \quad (5.6)$$

Wie im Algorithmus 5.1 zu sehen ist, wird im ersten Schritt das Eingangsbild mit einem Guidance-Bild, welches die Werte 0 hat, gefiltert. Dies entspricht der Gauß-Filterung. In den folgenden Iterationen wird das Guidance-Bild (und somit auch das finale Ergebnis) iterativ verfeinert.

---

**Algorithmus 5.1** Rolling-Guidance-Filter

---

```
procedure RG(out:  $J^{new}$ , in:  $I$ , in: Iter, in:  $\sigma_s$ , in:  $\sigma_r$ )  
  Initialize  $G^0$  with Zeros;  
  for  $m = 0; m < \text{Iter}; m++$  do  
     $G^{m+1} \leftarrow \text{JointBilateral}(I, G^m, \sigma_s, \sigma_r)$ ;  
  end for  
   $J^{new} \leftarrow G^{\text{Iter}}$ ;  
end procedure
```

---



**Abbildung 5.3:** Ergebnis des Rolling-Guidance-Filters mit acht Iterationen,  $\sigma_s = 3$  und  $\sigma_r = 0.1$ .

## 6 Evaluation

Im folgenden Kapitel werden die vorgestellten Ansätze zur Struktur-Textur-Zerlegung eines Bildes getestet und ausgewertet. Anschließend werden die Ergebnisse des quadratischen Variationsansatzes, des subquadratischen Variationsansatzes, des Rolling-Guidance-Filters und des bilateralen Filters miteinander verglichen. Für die Evaluation dieser verschiedenen Ansätze werden drei Experimente durchgeführt und deren Resultate betrachtet. Die Experimente umfassen folgende Aspekte:

1. Eignung für das Entfernen von Rauschen
2. Struktur und Texturerlegung bei ähnlicher Struktur und Textur
3. Struktur und Texturerlegung bei feiner Textur

Die Bilder die für diese Experimente hinzugezogen wurden sind in Abbildung 6.1 abgebildet.



**Abbildung 6.1:** Eingangsbilder für die Experimente: Frequenzbild/Ringbild (*oben links*). Bergbild mit einer großen und vielen kleinen Texturen (*oben rechts*). Realbild (barbara) mit Zahlreichen Texturen (*unten links*). Szene mit vielen Objekten (*unten rechts*).

## 6.1 Entfernen von Rauschen

Das folgende Experiment untersucht die Effektivität jedes Ansatzes bei der Entfernung von Rauschen aus einem Bild. Die Bewertung der Ergebnisbilder beruht dabei auf folgenden Kriterien:

1. Die Struktur des Bildes soll gut erkennbar sein.
2. Das Rauschen soll nur noch im Texturbild zu sehen sein.
3. Der MSE Wert soll möglichst gering sein.

Das optimale Ergebnis ist erreicht, wenn das Strukturbild mit dem unverrauschten Bild übereinstimmt und das Texturbild das Rauschen enthält. Vorbereitend auf dieses Experiment wurde das *Realbild* (aus Abbildung 6.1) mit additiven Gauß-Rauschen versehen (siehe Abbildung 6.2).



**Abbildung 6.2:** Links: Original; Rechts: Verrauschte Version mit einem MSE-Wert von 4216

Dieses Experiment wurde zweimal ausgeführt. Bei der ersten Ausführung lag der Fokus auf dem Entfernen des Rauschens, sodass die Struktur aber noch erkennbar ist. Beim zweiten Durchlauf lag der Schwerpunkt auf der Minimierung des MSE-Wertes. Abbildung 6.3 zeigt die Resultate des ersten und Abbildung 6.4 die des zweiten Durchlaufes.

### Struktur-Textur-Zerlegung :

Für ein möglichst gutes Ergebnis des ersten Durchlaufes sind die Parameter so gewählt, dass im Strukturbild möglichst wenig Rauschen zu sehen ist und der MSE-Wert möglichst gering ist. Die Parameter hierzu wurden empirisch festgelegt :

1. Quadratischer Variationsansatz:  $\alpha = 3.162$ ,  $\omega = 1.5$ , Iter = 40
2. Subquadratischer Variationsansatz:  $\alpha = 0.562$ ,  $\omega = 1.5$ ,  $\lambda_s = 1.05$ ,  $\lambda_d = 1$ , Iter = 10
3. Rolling-Guidance-Filter:  $\sigma_s = 2.2$ ,  $\sigma_r = 0.1$ , Iter = 8
4. Bilateraler-Filter:  $\sigma_s = 2.8$ ,  $\sigma_r = 1.1$

Die so gefilterten Bilder liefern folgende Ergebnisse (Abbildung 6.3 (Die Farbwerte der Texturbilder wurden für die Sichtbarkeit modifiziert: Multiplikation mit Faktor 10 und auf den mittleren Grauwert gebracht(+127.5))): Bei Verwendung des quadratischen Variationsansatzes (Abbildung 6.3 (a)) existiert noch ein leichtes Rauschen im Strukturbild. Die Konturen der einzelnen Objekte sind jedoch glatt und noch gut erkennbar. Dies ist beispielsweise in der Region der rechten Schulter der Frau ersichtlich. Der Übergang zwischen Kopftuch und Fußboden ist dort klar erkennbar. Ebenfalls sind kleinere Details, wie die vier Bücher im Hintergrund, eindeutig identifizierbar. Bei den anderen Ansätzen gehen diese kleineren Details verloren. Das Texturbild hingegen weist eine sehr schlechte Strukturentfernung auf, da die Frau immer noch ansatzweise zu erahnen ist. In diesem Fall wurde ein MSE-Wert von 344 erreicht, welcher als Indikator für gute Rauschentfernung gesehen werden kann.

Bei der Anwendung des subquadratischen Variationsansatzes (Abbildung 6.3 (b)) verschwimmt das Strukturbild. Diese Unschärfen beeinflussen hauptsächlich die Struktur der kleineren Objekte im Bild und sind im Bereich des Kopfes und der Hand gut zu sehen. Der im vorigen Abschnitt betrachtete Übergang zwischen Kopftuch und Boden ist optisch weicher als im Ergebnisbild des quadratischen Ansatzes. Im Texturbild kann man, wie im quadratischen Ansatz, leichte Züge einer Struktur erkennen, jedoch wurden mehr Kanten erhalten. Der hier erreichte MSE-Wert lag bei 346 und liegt somit über dem Wert des quadratischen Ansatzes.

Unter Anwendung des Rolling Guidance-Filters (Abbildung 6.3 (c)) kommt es ebenfalls zu Unschärfen im Strukturbild. Diese sind im Vergleich zum quadratischen Variationsansatz stärker und führen zu einer Art Fleckenbildung im Strukturbild. Auf Grund dieser verstärkten Verzerrung des Strukturbildes sind feinere Elemente, wie die Naht zweier aufeinanderliegender Bücher, nicht mehr erkennbar. Im Texturbild sind im Gegensatz zum quadratischen und subquadratischen Variationsansatz keine Strukturzüge erkennbar. Der resultierte MSE-Wert liegt bei 371, welcher deutlich über den Werten der beiden anderen Verfahren liegt.

Das Strukturbild des bilateralen Filters (Abbildung 6.3 (d)) ist etwas verschwommen, kleinere Details sind jedoch noch zu erkennen. Im Texturbild sind leichte Kantenansätze wahrnehmbar. Mit einem MSE-Wert von 341 liegt dieser unterhalb der Werte der anderen Methoden. Im Vergleich zu den anderen Verfahren liefert der bilaterale Filter einen guten Kompromiss zwischen Strukturerhaltung und Rauschentfernung.

Alle Ergebnisbilder beinhalten die größten Strukturelemente des Originalbildes, wobei der quadratische Variationsansatz ein optisch besseres Ergebnis hinsichtlich kleinerer Details liefert. Die Kanten erscheinen beim Rolling-Guidance-Filter deutlich schärfer als bei den anderen Ansätzen. In Bezug auf das Texturbild liefert der Rolling-Guidance-Filter das gewünschte Ergebnis bei der Rauschentfernung. Das Rauschen ist nur noch im Texturbild erkennbar. Aufgrund der visuellen Auswertung des ersten Tests lässt sich momentan vermuten, dass der bilaterale Filter im Gegensatz zu den anderen bessere Ergebnisse liefert. Dies belegt auch die Differenz zwischen dem errechneten Bild und dem Originalbild, die beim bilateralen Filter am kleinsten ist.

### **Entrauschen:**

Zur Erreichung eines guten Ergebnisses im zweiten Durchgang des Experiments sind die Parameter so gewählt, dass der MSE-Wert sein Minimum erreicht. Die Parameter wurden ebenfalls empirisch ermittelt:

1. Quadratischer Variationsansatz:  $\alpha = 1.77$ ,  $\omega = 1.5$ , Iter = 40

2. Subquadratischer Variationsansatz:  $\alpha = 0.749$ ,  $\omega = 1.5$ ,  $\lambda_s = 1.18$ ,  $\lambda_d = 1.29$ , Iter = 10
3. Rolling-Guidance:  $\sigma_s = 1.3$ ,  $\sigma_r = 0.7$ , Iter = 10
4. Bilateraler-Filter:  $\sigma_s = 1.8$ ,  $\sigma_r = 0.4$

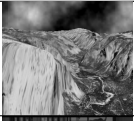



Der Ausgangs-MSE-Wert dieses Durchgangs liegt bei 1594. Nach Durchführung erreichte der quadratische Ansatz einen MSE-Wert von 248, der subquadratische Ansatz einen Wert von 247, der Rolling-Guidance-Filter 246 und der bilaterale Filter 243. Alle Ansätze beinhalten in ihren Texturbildern Kantenelemente, wobei diese im Texturbild des quadratischen, subquadratischen Variationsansatzes und des bilateralen Filters stärker ausgeprägt sind als beim Rolling-Guidance-Texturbild.

Der bilaterale Filter erwies sich somit im zweiten Durchgang mit dem *Realbild* (aus Abbildungfig 6.1) als bestes Verfahren. Es stellt sich die Frage, inwiefern speziell das *Realbild* verantwortlich für diese Schlussfolgerung ist und ob sich unter Hinzunahme anderer Beispielbilder dieses Ergebnis bestätigt. Um dies zu beantworten, wurde dieses Experiment nochmals mit den anderen Bildern aus Abbildung 6.1 und unterschiedlichen Standardabweichungen des Gauß-Rauschens durchgeführt. Es ergaben sich folgende Ergebnisse:

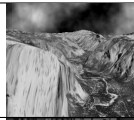



Anhand der Tabellen 6.1 bis 6.4 ist zu erkennen, dass bei der Rauschentfernung der bilaterale Filter als auch der Rolling-Guidance-Filter die besseren Ergebnisse liefern. Deutliche Unterschiede zwischen dem Rolling-Guidance und dem bilateralen Filter sind vor allem im vierten Bild, dem Ringbild, zu erkennen. Je ausgeprägter der Rauschanteil in diesem Bild ist, desto besser erweist sich der Rolling-Guidance Filter. Bei den ersten beiden Bildern, die mit einer Standardabweichung von 20 verrauscht wurden, liefert der Rolling-Guidance Filter nur minimal bessere Ergebnisse als der bilaterale Filter.

Hieraus geht hervor, dass der Rolling-Guidance-Filter und der bilaterale Filter sich hinsichtlich des Entfernens von Rauschen als bessere Alternativen erweisen. Je nach Strukturgröße ist es jedoch sinnvoll, zwischen diesen beiden Methoden zu alternieren, um ein bestmögliches Ergebnis zu erhalten.

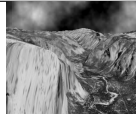





Bilder	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$
	<b>0</b>	56.282	124.132	225.263
	<b>0</b>	50.018	109.312	200.841
	<b>0</b>	57.813	142.436	248.528
	<b>0</b>	58.884	161.451	404.482

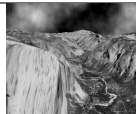



**Tabelle 6.1:** MSE-Wertetabelle des quadratischen Variationsansatzes. Die jeweils besten Werte im Vergleich zu den restlichen Verfahren sind schwarz hervorgehoben.

Bilder	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$
	<b>0</b>	57.628	120.238	223.405
	<b>0</b>	47.980	98.318	187.849
	<b>0</b>	62.751	117.170	210.233
	<b>0</b>	75.470	181.505	404.232

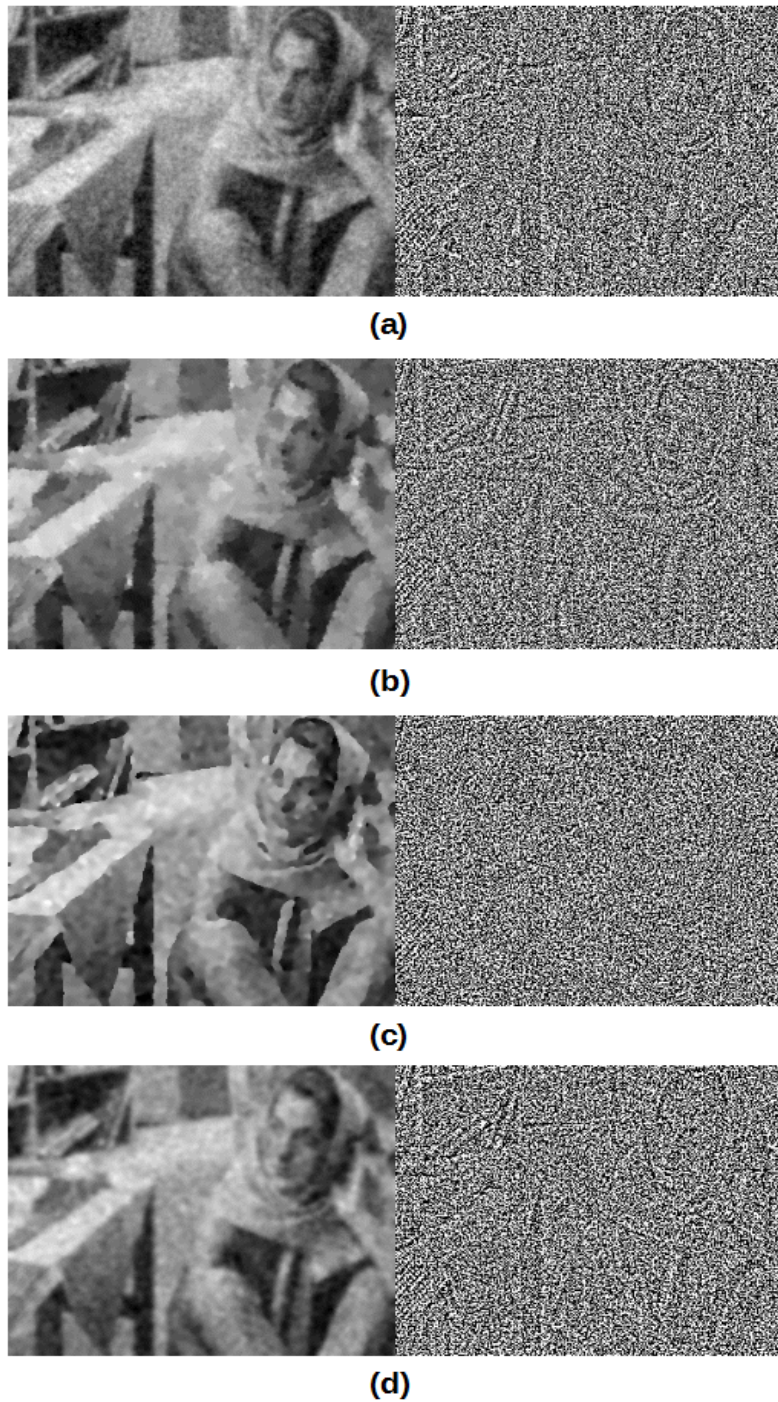
**Tabelle 6.2:** MSE-Wertetabelle des subquadratischen Variationsansatzes. Die jeweils besten Werte im Vergleich zu den restlichen Verfahren sind schwarz hervorgehoben.

Bilder	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$
	<b>0</b>	46.774	<b>106.06</b>	217.74
	<b>0</b>	<b>33.824</b>	<b>82.076</b>	<b>170.213</b>
	<b>0</b>	51.125	114.765	208.015
	<b>0</b>	47.59	<b>97.309</b>	<b>235.22</b>

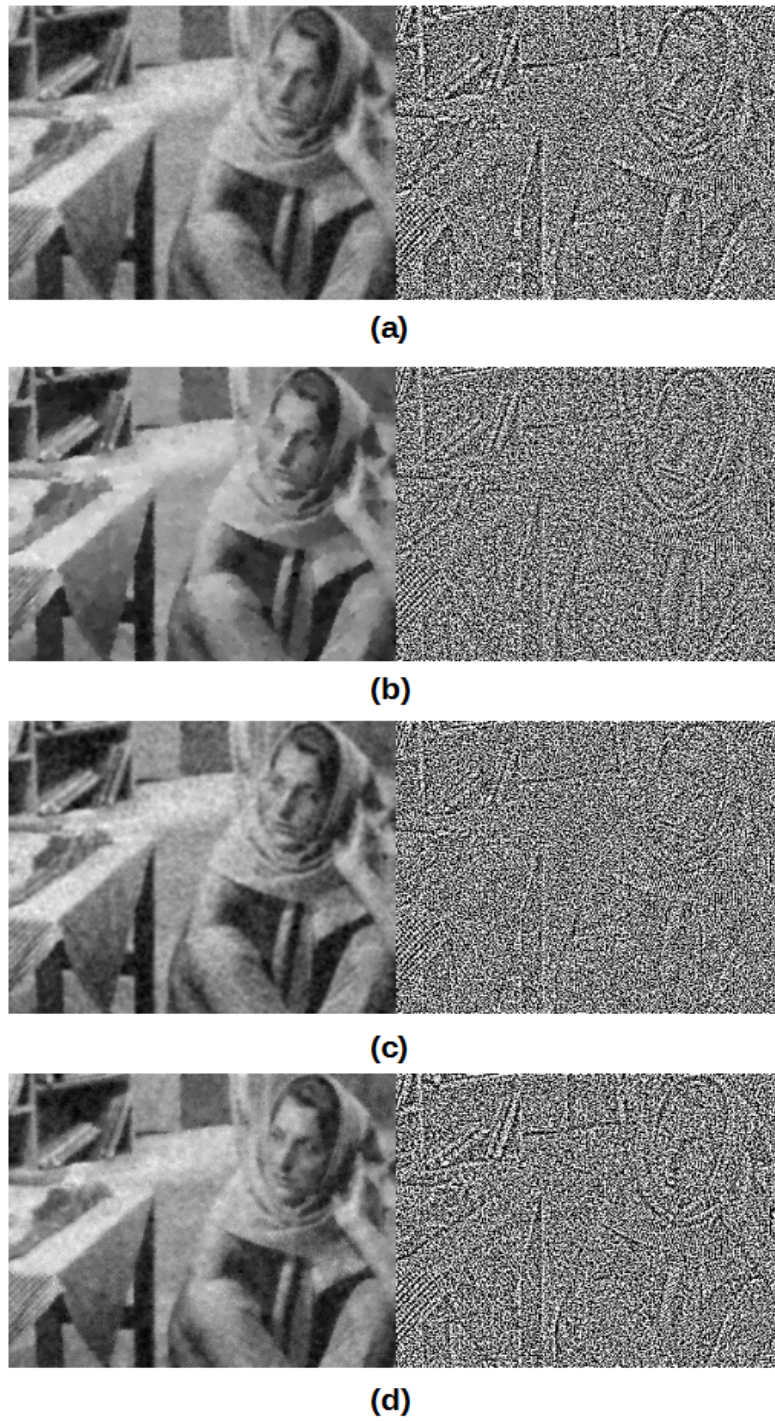
**Tabelle 6.3:** MSE-Wertetabelle des Rolling-Guidance-Filters. Die jeweils besten Werte im Vergleich zu den restlichen Verfahren sind schwarz hervorgehoben.

Bilder	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$
	<b>0</b>	<b>44.504</b>	108.24	<b>212.8</b>
	<b>0</b>	36.702	88.331	186.986
	<b>0</b>	<b>44.274</b>	<b>106.788</b>	<b>206.736</b>
	<b>0</b>	<b>45.104</b>	105.854	249.33

**Tabelle 6.4:** MSE-Wertetabelle des bilateralen Filters. Die jeweils besten Werte im Vergleich zu den restlichen Verfahren sind schwarz hervorgehoben.



**Abbildung 6.3:** links: Struktur-rechts: Textur-Zerlegung: (a) Quadratischer Variationsansatz. (b) Subquadratischer Variationsansatz. (c) Rolling-Guidance-Filter und (d) Bilateraler Filter.



**Abbildung 6.4:** Entrauschen: Struktur(*links*), Textur(*rechts*): (a) Quadratischer Variationsansatz. (b) Subquadratischer Variationsansatz. (c) Rolling-Guidance und (d) Bilateraler Filter.

## 6.2 Struktur- und Texturerlegung bei ähnlicher Struktur und Textur

Das zweite Experiment bezieht sich auf die Zerlegung des Bildes bei Ähnlichkeit von Struktur und Textur. Grundlage für dieses Experiment ist ein Bild, bei dem die Größe von Struktur und Textur nahezu identisch ist. Es dient dazu, mögliche Effekte sichtbar zu machen, um diese anschließend sinnvoll interpretieren zu können. Dieses Experiment wurde mit dem *Ringbild* (aus Abbildung 6.1) durchgeführt. Nach folgenden Merkmalen soll bewertet werden:

1. Die Struktur des errechneten Bildes soll gut zu erkennen sein.
2. Alle Ringe ab dem 5. Ring, ausgehend von innen, sollen entfernt werden.
3. Die Kanten sollen möglichst sehr scharf und gut erkennbar sein.

Um die definierten Ziele zu erreichen, sind die Parameter so gewählt, dass der 6. Ring nicht mehr im Strukturbild zu erkennen ist und alle Ringe ab dem 5. Ring möglichst scharfe Kanten haben. Zur Erreichung dieser Ziele wurden folgenden Parameter verwendet.

1. Quadratischer Variationsansatz:  $\alpha = 749.89$ ,  $\omega = 1.5$ ,  $\text{Iter} = 40$
2. Subquadratischer Variationsansatz:  $\alpha = 0.999$ ,  $\omega = 1.5$ ,  $\lambda_s = 1.4$ ,  $\lambda_d = 0.1$ ,  $\text{Iter} = 10$
3. Rolling-Guidance-Filter:  $\sigma_s = 5.6$ ,  $\sigma_r = 0.1$ ,  $\text{Iter} = 8$
4. Bilateraler-Filter:  $\sigma_s = 9.1$ ,  $\sigma_r = 4.0$

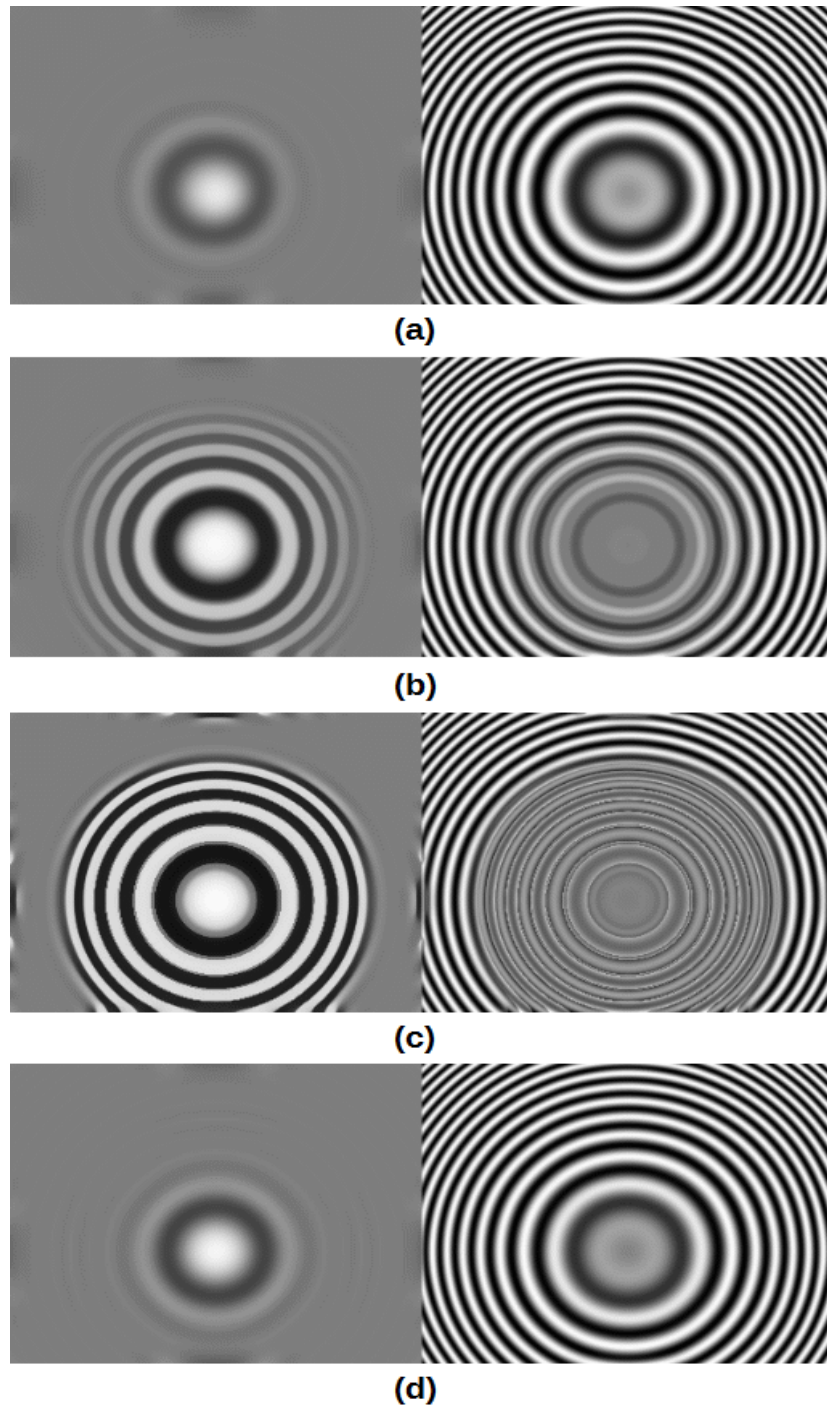
Abbildung 6.5 zeigt die Resultate des Experiments. In diesem Fall erzeugt der quadratische Variationsansatz keine scharfen Kanten im Strukturbild. Alle Ringe sind verschwommen. Dies spricht dafür, dass sich dieser Variationsansatz nicht für die Zerlegung bei ähnlicher Struktur und Textur eignet.

Beim subquadratischen Variationsansatz sind die mittleren Ringe im Strukturbild gut erkennbar. Der 5. Ring ist jedoch nur noch zu erahnen. An den grauen Ringen im Texturbild (*b*) ist gut zu erkennen, dass hier wenig Textur entfernt wurde. Durch diese Eigenschaft erscheint der subquadratische Variationsansatz besser als der quadratische Variationsansatz.

Der bilaterale Filter erzeugt, ähnlich wie der quadratische Variationsansatz, ein sehr verschwommenes Strukturbild. Dies ist jedoch subjektiv betrachtet besser als das des quadratischen Variationsansatzes, da die mittlere weiße Fläche stärker ausgeprägt ist.

Unter Verwendung des Rolling-Guidance-Filters sind die inneren vier Ringe im Strukturbild sehr scharf, wogegen der 5. verschwimmt. Die grauen Kreise im Texturbild bestätigen die Strukturhaltung. An den Rändern des Bildes treten zudem Aliasing-Effekte auf. Außerdem entstehen im Strukturbild ungewollte graue Ringe.

Anhand der Beobachtungen kommt man zu dem Schluss, dass der Rolling-Guidance-Filter sich erneut als die beste Methode zur Struktur- und Texturerlegung erweist.



**Abbildung 6.5:** links: Struktur- rechts: Textur-Zerlegung bei ähnlicher Struktur und Textur: (a) Quadratischer Variationsansatz, (b) Subquadratischer Variationsansatz, (c) Rolling-Guidance-Filter und (d) Bilateraler Filter.

## 6.3 Struktur und Texturerlegung bei feiner Textur

Der dritte Test befasst sich mit der Zerlegung von Struktur und Textur eines Realbildes. Dazu werden folgende Ziele vorgegeben:

1. Die Struktur des Bildes soll gut zu erkennen sein.
2. Es soll keine Strukturkanten im Texturbild geben.
3. Die Textur soll im Texturbild gut erkennbar sein.

Zu Analyse Zwecken wurde erneut das *Realbild* (aus Abbildung 6.1) zur Durchführung des Experiments herangezogen. Dieses Bild eignet sich besonders gut, da es eine Vielfalt an Texturgrößen aufweist. Für die nachfolgende Ausführung liegt das Hauptaugenmerk des Struktur- und des Texturbildes auf dem dargestellten Kopftuch. Die Veränderungen der anderen Bildelemente sind zweitrangig. Ziel ist es nun, die Zerlegung des Bildes so zu gestalten, dass die Textur des Kopftuches komplett entfernt wird. Daraus resultieren folgende Parameter für die vier verschiedenen Ansätze.

1. Quadratischer Variationsansatz:  $\alpha = 3.162$ ,  $\omega = 1.5$ ,  $\text{Iter} = 40$
2. Subquadratischer Variationsansatz:  $\alpha = 0.99$ ,  $\omega = 1.5$ ,  $\lambda_s = 1.05$ ,  $\lambda_d = 1$ ,  $\text{Iter} = 10$
3. Rolling-Guidance-Filter:  $\sigma_s = 1.1$ ,  $\sigma_r = 0.1$ ,  $\text{Iter} = 8$
4. Bilateraler-Filter:  $\sigma_s = 1.0$ ,  $\sigma_r = 0.6$

In Abbildung 6.6 sind die Ergebnisse dargestellt (Die Farbwerte der Texturbilder wurden auch hier für die Sichtbarkeit modifiziert: Multiplikation mit Faktor 10 und auf den mittleren Grauwert gebracht(+127.5)). Es ist gut zu erkennen, dass der quadratische Variationsansatz die Textur entfernt, jedoch ein unscharfes Strukturbild erzeugt. Das Texturbild ist sehr ausgeprägt, da es viele Kanten enthält; dies ist jedoch unerwünscht. Im Falle des subquadratischen Variationsansatzes, erhalten wir ein scharfes Strukturbild. Die Textur wurde wie gewünscht entfernt. Das Texturbild besitzt jedoch zu viele Schattierungen. Dieses Ergebnis liefert zwar noch nicht den gewünschten Effekt, es werden jedoch mehr Kanten erhalten als beim quadratische Ansatz. Der Rolling-Guidance-Filter entfernt die Textur gut und behält die Schattierungen im Strukturbild bei. Das Texturbild besitzt relativ wenig Kanten, weshalb sich dieser Ansatz gegenüber den anderen Verfahren als besser erweist. Der bilaterale Filter zeigt ein subjektiv schlechteres Ergebnis. Dieses Ergebnis war zu erwarten, da der verwendete Rolling-Guidance-Filter eine optimierte Variante des bilateralen Filters hinsichtlich dieser speziellen Anwendung ist. Das Ergebnis dieses Experiments zeigt, dass erneut der Rolling-Guidance-Filter das beste Ergebnis liefert. Insgesamt scheint also der Rolling-Guidance-Filter den anderen Verfahren überlegen zu sein. In einigen Fällen kann zumindest noch der bilaterale Filter mithalten.



(a)



(b)



(c)



(d)

**Abbildung 6.6:** links: Struktur- rechts: Textur-Zerlegung bei feiner Textur: (a) Quadratischer Variationsansatz, (b) Subquadratischer Variationsansatz, (c) Rolling-Guidance-Filter und (d) bilateraler Filter



## 7 Zusammenfassung und Ausblick

Eine gute Zerlegung eines Bildes in seinen Struktur- und Texturanteil ist für viele Methoden ausschlaggebend, die für die Vor- und Nachbereitung von Bildern genutzt werden. Durch die Vielfalt an Verfahren, die hierzu genutzt werden können, ist es für den Benutzer besonders von Interesse, ob es ein dominierendes Verfahren gibt. Dies wurde im Rahmen der Arbeit anhand von vier Verfahren untersucht.

Das Hauptaugenmerk lag hierbei auf der Implementierung und dem anschließenden Vergleich eines quadratischen Variationsansatzes, eines subquadratischen Variationsansatzes, des Rolling-Guidance-Filters und des bilateralen Filters. Aus dem durchgeführten Test zur Entfernung von Rauschen geht hervor, dass sich hierfür der Rolling-Guidance-Filter und der bilaterale-Filter besonders gut eignen. Das Experiment zur Struktur- Texturerlegung bei ähnlicher Struktur und Textur zeigte, dass der Rolling-Guidance-Filter die bessere Alternative im Vergleich zu den anderen Ansätzen ist. Zum selben Schluss kommt man bei der Zerlegung eines Realbildes mit feiner Textur. Die Ergebnisse dieser Untersuchung zeigen, dass sich, je nach Anwendungsfall, der iterative bilaterale Filter oder der Rolling-Guidance-Filter als beste Alternativen erweisen.

### Ausblick

Die aus dieser Arbeit gewonnenen Erkenntnisse können für die Zukunft als Ansporn für die Verbesserung und Weiterentwicklung des bilateralen Filters und des Rolling-Guidance-Filters gesehen werden. Dabei zeigen die durchgeführten Versuche Verbesserungspotenziale auf, die in weiterführenden Arbeiten betrachtet werden sollten. Es wäre zu überlegen, ob man die Vorteile des Rolling-Guidance-Filters und des bilateralen Filters mit den Variationsansätzen verknüpft. Beispielsweise können nicht-lokale Datenterme mit räumlichen und farbspezifischen Gewichten in einen Variationsansatz integriert werden. Zusätzlich könnten weitere texturspezifische Gewichte zu den Gewichten im bilateralen Filter und Rolling-Guidance-Filter hinzugefügt werden, wie bei den Variationsansätzen mit speziell angepasstem Daten und Glattheitsterm bei spezieller Textur [AGCO06].



# Literaturverzeichnis

- [AGCO06] J.-F. Aujol, G. Gilboa, T. Chan, S. Osher. Structure-texture image decomposition—modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006. (Zitiert auf den Seiten 8 und 49)
- [All97] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE Transactions on Signal Processing*, 45(4):913–917, 1997. (Zitiert auf Seite 8)
- [Alo88] J. Aloimonos. Shape from texture. *Biological Cybernetics*, 58(5):345–360, 1988. (Zitiert auf Seite 11)
- [Bov10] A. C. Bovik. *Handbook of Image and Video Processing*. Academic Press, 2010. (Zitiert auf Seite 11)
- [CM99] T. F. Chan, P. Mulet. On the convergence of the lagged diffusivity fixed point method in total variation image restoration. *SIAM journal on numerical analysis*, 36(2):354–367, 1999. (Zitiert auf Seite 25)
- [DD02] F. Durand, J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. *ACM Transactions on Graphics*, 21(3):257–266, 2002. (Zitiert auf Seite 8)
- [DH95] D. Dunn, W. E. Higgins. Optimal Gabor filters for texture segmentation. *Image Processing, IEEE Transactions on*, 4(7):947–964, 1995. (Zitiert auf Seite 8)
- [KNPS68] J. Kačur, J. Nečas, J. Polák, J. Souček. Convergence of a method for solving the magnetostatic field in nonlinear media. *Aplikace Matematiky*, 13(6):456–465, 1968. (Zitiert auf Seite 25)
- [KS10] M. Kass, J. Solomon. Smoothed local histogram filters. *ACM Transactions on Graphics*, 29(4):100, 2010. (Zitiert auf Seite 8)
- [Mei99] A. Meister. Numerik linearer Gleichungssysteme. *Vieweg, Braunschweig*, 1999. (Zitiert auf Seite 17)
- [Net89] I. Netuka. *Commentationes Mathematicae Universitatis Carolinae*. Mathematical Institute of the Charles University, 1989. (Zitiert auf Seite 25)
- [Nik04] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120, 2004. (Zitiert auf Seite 8)
- [OR70] J. M. Ortega, W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*, Band 30. SIAM, 1970. (Zitiert auf Seite 17)

- [PF06] E. Prados, O. Faugeras. Shape from shading. In *Handbook of Mathematical Models in Computer Vision*, S. 375–388. Springer, 2006. (Zitiert auf Seite 11)
- [PT11] N. Payet, S. Todorovic. Scene shape from texture of objects. In *Computer Vision and Pattern Recognition, 2011 IEEE Conference on*, S. 2017–2024. IEEE, 2011. (Zitiert auf Seite 11)
- [ROF92] L. I. Rudin, S. Osher, E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992. (Zitiert auf Seite 8)
- [SK11] H.-R. Schwarz, N. Köckler. *Numerische Mathematik*. Springer-Verlag, 2011. (Zitiert auf Seite 14)
- [TM98] C. Tomasi, R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, S. 839–846. IEEE, 1998. (Zitiert auf den Seiten 7 und 33)
- [ves] Y. ves. Oscillating patterns in image processing and in some nonlinear evolution equations, March 2001. *The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*. (Zitiert auf Seite 8)
- [Wei06] B. Weiss. Fast median and bilateral filtering. In *ACM Transactions on Graphics*, Band 25, S. 519–526. ACM, 2006. (Zitiert auf Seite 7)
- [WS99] J. Weickert, C. Schnörr. Räumlich-zeitliche Berechnung des optischen Flusses mit nichtlinearen flußabhängigen Glattheitstermen. In *Mustererkennung 1999*, S. 317–324. Springer, 1999. (Zitiert auf Seite 11)
- [XYX12] L. Xu, Q. Yan, Y. Xia, J. Jia. Structure extraction from texture via relative total variation. *ACM Transactions on Graphics*, 31(6):139, 2012. (Zitiert auf Seite 8)
- [You14] D. M. Young. *Iterative Solution of Large Linear Systems*. Elsevier, 2014. (Zitiert auf Seite 19)
- [ZK83] G. Zimmermann, R. Kories. Die Eignung Spezifischer Bildstrukturen für die Bewegungs-Bestimmung in Bildfolgen. In *Mustererkennung 1983*, S. 60–65. Springer, 1983. (Zitiert auf Seite 11)
- [ZSXJ14] Q. Zhang, X. Shen, L. Xu, J. Jia. Rolling Guidance Filter. In *Computer Vision–ECCV 2014*, S. 815–830. Springer, 2014. (Zitiert auf Seite 34)

Alle URLs wurden zuletzt am 27. 04. 2015 geprüft.

## **Erklärung**

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

---

Ort, Datum, Unterschrift