Alignmentkorrekturen und Fusion von Dokumentaufnahmen

Am Fachbereich Mathematik und Informatik
in der Arbeitsgruppe Künstliche Intelligenz
der Freien Universität Berlin

Diplomarbeit

vorgelegt von Maxim Schaubert

November 2009

Betreuer:

Prof. Dr. Raúl Rojas und
Dr. Marco Block-Berlitz
Arbeitsgruppe Künstliche Intelligenz

Institut für Mathematik und Informatik
Freie Universität Berlin
Takustr. 9
14195 Berlin
Deutschland

Inhaltsverzeichnis

Kapitel 1

Einführung

Die vorliegende Arbeit beschäftigt sich mit Bearbeitung von Dokumentenaufnahmen. Es werden Methoden vorgestellt, die aus einer Reihe von unterschiedlich belichteten Aufnahmen eines Dokumentes ein Resultatbild erstellen, auf dem die "besten" Regionen der Originalaufnahmen zusammen gestellt werden. Es werden verschiedene Kriterien gezeigt, die für Bewertung der Qualität von Bildregionen und einzelnen Pixeln verwendet werden können. Außerdem wird eine umfangreiche Auswertung der vorgestellten Methoden durchgeführt und Methode mit besten Ergebnissen ermittelt.

1.1 Motivation

Manchmal ist es wegen schlechten Lichtverhältnissen nicht möglich, eine perfekt belichtete Aufnahme zu machen, auf der das ganze Dokument sichtbar ist. Das kann vorkommen, wenn ein Teil des Dokumentes im Schatten liegt oder, umgekehrt, stark beleuchtet ist (Abbildung 1.1). In solchen Fällen können aber mehrere Aufnahmen mit unterschiedlichen Belichtungszeiten gemacht werden, so, dass ein Teil des Bildes, das auf einer Aufnahme unter- oder überbelichtet ist, auf einer der anderen Aufnahmen gut sichtbar wird (Abbildung 1.2). Aus einer Reihe von unterschiedlich belichteten Aufnahmen kann mit Hilfe von Bildfusion ein Bild erstellt werden, auf dem das ganze Dokument gut lesbar ist.

Für die Erstellung des Resultatbildes werden Teile der Originalaufnahmen ausgewählt, die den angegebenen Auswahlkriterien am besten entsprechen. Als Kriterium für Auswahl einer Bildregion kann z.B. der Informationsinhalt oder ein Maß der Fokussierung verwendet werden. In weiteren Kapiteln werden verschiedene Auswahlkriterien näher betrachtet.

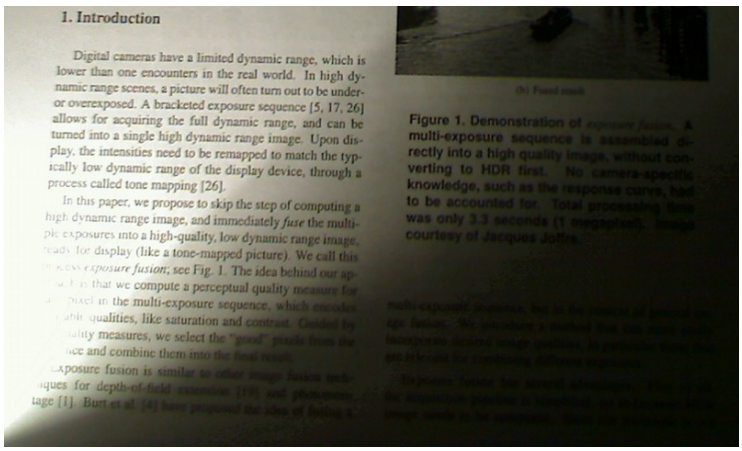


Abbildung 1.1: *Aufnahme mit unter- und überbelichteten Regionen*

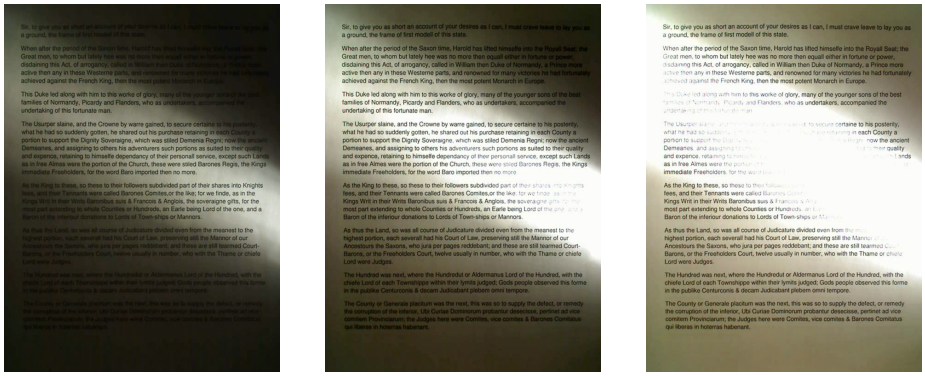


Abbildung 1.2: *Aufnahmen mit unterschiedlichen Belichtungszeiten*

1.2 Aufbau der Arbeit

In diesem Abschnitt wird ein Überblick über einzelne Kapitel dieser Arbeit gegeben und ihre Zusammenhänge beschrieben.

Kapitel 2 beschreibt die in dieser Arbeit verwendete Notation, grundlegende Begriffe und Methoden der Bildverarbeitung. Es werden verschiedene Binarisierungsverfahren erklärt und Bildhistogramme, Lineare Filter und Konvolution besprochen.

Im Abschnitt 2.2 wird ein Einblick in die verwandten Arbeiten und eine Übersicht von Methoden der Bildfusion gegeben. Es wird der Begriff der Hochkontrastbilder erklärt und einige Verfahren für die Erstellung von solchen Bildern erklärt, darunter auch das in dieser Arbeit verwendete Verfahren - *Exposure-Fusion.*

In weiteren Abschnitten werden Methoden der *pixel-* und *regionenbasierter*

Bildfusion betrachtet. Bei der regionenbasierten Fusion spielen die *Qualitätsmaße*, Kriterien für Auswahl von Regionen, eine entscheidende Rolle. Sie werden im Abschnitt 2.2.2 besprochen.

Das Resultatbild der regionenbasierten Methoden kann mit Hilfe eines Blending-Verfahrens nachgebessert werden, dieses Verfahren wird im Abschnitt 2.2.4 beschrieben.

Im Schlußteil des Kapitels werden pixelbasierte Methoden und Kriterien für Pixelauswahl vorgestellt.

Im **Kapitel 3** werden Methoden der Alignmentkorrektur behandelt. Abschnitt 3.1 betrachtet Verfahren der globalen Alignmentkorrektur. Im Abschnitt 3.2 wird ein lokales rekursives Verfahren besprochen. Es wird auch gezeigt, wie die Thin-Plate-Spline Methode für Alignmentkorrektur verwendet werden kann. Im letzten Abschnitt des Kapitels werden Experimente mit Alignmentkorrekturen beschrieben und deren Ergebnisse gezeigt.

Kapitel 4 gibt eine Beschreibung von Experimenten, die im Laufe der Arbeit mit den verschiedenen Bildfusionsmethoden durchgeführt wurden. Es werden Ergebnisse der getesteten Methoden gezeigt und ihre Vor- und Nachteile besprochen.

Bei regionenbasierten Methoden werden verschiedene Kriterien für Auswahl der Regionen aus Originalaufnahmen getestet. Es wird gezeigt, wie durch Auswahl von Bildregionen mit höchstem Qualitätsmaß ein Resultatbild zusammen gestellt werden kann, das die besten Teile der Originalaufnahmen enthält. Es werden Qualitätsmaße für Auswahl von best fokussierten und informationsreichsten Bildteilen getestet.

Ein wichtiger Schritt bei regionenbasierter Fusion ist die Nachbearbeitung des erstellten Resultatbildes. Um die eventuell entstandenen Kanten zwischen einzelnen Bildregionen zu glätten, wird eine Blending-Methode verwendet. Eine Beschreibung der Methode wurde im früheren Kapitel gegeben, im Abschnitt 4.2 werden nun einige Optimierungstechniken vorgeschlagen.

In weiteren Abschnitten werden pixelbasierte Methoden der Bildfusion beschrieben, und im Abschnitt 4.4, auch die auf Kantenintensitäten basierte Methode, die die besten Ergebnisse geliefert hat.

Im Schlussteil des Kapitels werden Experimente mit Focus-Fusion vorgestellt und Ergebnisse gezeigt. Für diese Experimente wurden jeweils zwei unterschiedlich fokussierte Aufnahmen genommen. Eine der Aufnahmen wurde auf einem nahen und andere auf einem fernen Objekt fokussiert. Im Experiment wurden die Aufnahmen zu einem Bild fusioniert, auf dem alle Teile im Fokus sind. Ergebnisse werden im Abschnitt 4.5 vorgestellt.

Im **Kapitel 5** wird eine Auswertung von allen betrachteten Bildfusionsmethoden vorgestellt. Es wird der Prozess der Erstellung von Testaufnahmen

erklärt und dafür benutzte Technik beschrieben. Weiter wird erklärt, wie die Fusionsmethoden getestet und Resultatbilder aus den Testaufnahmen erzeugt wurden.

Im letzten Abschnitt werden Resultate der Auswertung vorgestellt und die gewonnenen Erkenntnisse besprochen. Schließlich wird die beste Methode für Fusion von Dokumentenaufnahmen ermittelt.

Kapitel 6 gibt eine Zusammenfassung der gesamten Arbeit.

Kapitel 2

Theorie und verwandte Arbeiten

In diesem Kapitel werden grundlegende Begriffe und Methoden der Bildverarbeitung beschreiben. Außerdem werden einige Techniken für Erstellung von Hochkontrastbildern erleutert und Methoden der Bildfusion vorgestellt.

Das Thema der Bildfusion wurde schon in einigen wissenschaftlichen Artikeln beschrieben, im Abschnitt 2.2 wird eine Übersicht über die wichtigsten Arbeiten gegeben.

2.1 Grundlagen

2.1.1 Notation für Bild- und Pixelzugriff

Für die Bezeichnung von Bildern werden in dieser Arbeit fett gesetzte Großbuchstaben verwendet - **I**, **J**. Dabei wird ein Bild als eine Matrix von Pixelwerten betrachtet. Im Gegensatz zu Indexierung von Matrixelementen in der Mathematik, bei der der erste Index die Zeilen- und der zweite die Spaltennummer angibt, wird hier bei Indexierung erst der Spalten-, dann der Zeilenindex angegeben. D.h. $\mathbf{I}(x, y)$ ist der Pixelwert der in der Spalte x und Zeile y steht (siehe Abbildung 2.1).

Dimensionen einer Bildmatrix werden mit Großbuchstaben angegeben - $M \times N$, die erste Dimension bezeichnet dabei die Anzahl von Spalten, die zweite - Anzahl von Zeilen. Für Indexe x und y gilt folgendes: $0 \leq x < M$, $0 \leq y < N$.

In einigen Fällen ist es zweckmäßig ein Bild in Form einer Matrix darzustellen (siehe Formel 2.1).

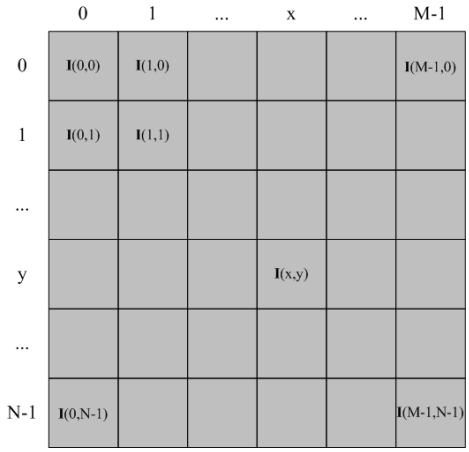


Abbildung 2.1: *Indexierung von Pixel*

$$\mathbf{I}(x,y) = \begin{bmatrix} \mathbf{I}(0,0) & \mathbf{I}(1,0) & \cdots & \mathbf{I}(M-1,0) \\ \mathbf{I}(0,1) & \mathbf{I}(1,1) & \cdots & \mathbf{I}(M-1,1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I}(0,N-1) & \mathbf{I}(1,N-1) & \cdots & \mathbf{I}(M-1,N-1) \end{bmatrix} \tag{2.1}$$

Wenn eine Nachbarschaft eines Pixels angegeben werden muss, wird folgende Form verwendet - $\mathbf{I}_{m\times n}(x,y)$. Hier bezeichnet $m\times n$ die Größe der Nachbarschaft und (x,y) gibt an, in welchem Pixel die Nachbarschaft zentriert ist. Ein Beispiel für eine 3×3 Nachbarschaft ist in der Formel 2.2 gegeben.

$$\mathbf{I}_{3\times 3}(x,y) = \begin{bmatrix} \mathbf{I}(x-1,y-1) & \mathbf{I}(x,y-1) & \mathbf{I}(x+1,y-1) \\ \mathbf{I}(x-1,y) & \mathbf{I}(x,y) & \mathbf{I}(x+1,y) \\ \mathbf{I}(x-1,y+1) & \mathbf{I}(x,y+1) & \mathbf{I}(x+1,y+1) \end{bmatrix} \tag{2.2}$$

Bei mehreren Aufnahmen eines Dokumentes entstehen Bilder, die später zu einem neuen Bild fusioniert werden, in diesem Fall werden die Bilder mit $\mathbf{I}_i(x,y)$, $1 \leq i \leq N$ bezeichnet, wobei N die Anzahl der Aufnahmen ist.

Da es in dieser Arbeit um Fusion von Dokumentaufnahmen geht und für Dokumente Farben irrelevant sind, sind alle betrachteten Bilder - Grauwertbilder, bestehen also aus nur einem Farbkanal. Farbbilder werden, wie im Abschnitt 2.1.2 beschrieben, in Grauwertbilder konvertiert.

2.1.2 Konvertierung Farb- zu Grauwertbild

In Bildverarbeitung gibt es verschiedene Farbräume, der meistverwendete davon ist der **RGB** Farbraum. Eine Farbe wird dabei mit drei Werten kodiert, die Werte sind Intensitäten von **R**ot, **G**rün und **B**lau. Ein Bild im RGB Farbraum hat dementsprechend drei Farbkanäle, d.h. drei Intensitätswerte pro Pixel.

Bei der Konvertierung von Farb- zu Grauwertbild werden drei Farbwerte gemischt, um einen Grauwert zu bilden. Aus einem Bild $\mathbf{I}$ im RGB Farbraum mit drei Farbkanälen $\mathbf{I}_R$,$\mathbf{I}_G$,$\mathbf{I}_B$ kann ein Grauwertbild $\mathbf{G}$ mit nur einem Farbkanal nach folgender Formel erzeugt werden:

$$\mathbf{G}(x,y) = 0.299 * \mathbf{I}_R(x,y) + 0.587 * \mathbf{I}_G(x,y) + 0.114 * \mathbf{I}_B(x,y) \tag{2.3}$$

Die Koeffizienten in der Formel berücksichtigen dabei, dass die Farbrezeptoren im menschlichen Auge unterschiedliches Helligkeitsempfinden für unterschiedliche Farben haben [2], deswegen müssen die Farbwerte bei Konvertierung unterschiedlich gewichtet werden. Rot wird heller wahrgenommen als Blau, und Grün heller als Rot, dementsprechend sind auch die Farbkomponenten gewichtet.

2.1.3 Binarisierung von Grauwertbildern

Binarisierung ist eine Reduktion des Wertebereichs eines Grauwertbildes auf zwei Werte. Bei Grauwertbildern wird der Wertebereich $\{0, ..., 255\}$ auf $\{0, 1\}$ reduziert, somit wird ein Grauwertbild zu einem Schwarz-Weiß-Bild konvertiert. In der Regel wird Schwarz mit 0 und Weiß mit 1 kodiert.

Für Binarisierung können verschiedene Schwellwertverfahren verwendet werden. Die lassen sich in drei Klassen einteilen - *globale*, *lokale* und *adaptive*.

Bei **globalen** Schwellwertverfahren wird ein Schwellwert T für das gesamte Bild gewählt und das Binärbild B nach folgender Vorschrift erzeugt:

$$B(x,y) = T_{global}(x,y) = \begin{cases} 1, & falls\ \mathbf{I}(x,y) > T \\ 0, & falls\ \mathbf{I}(x,y) \leq T \end{cases} \tag{2.4}$$

Eine der bekanntesten Methoden zur Bestimmung des globalen Schwellenwertes ist die Methode von Otsu [3]. Die Grundidee der Methode ist folgende: alle Pixel im Bild werden basierend auf einem Schwellwert T in zwei Klassen K_0 und K_1 eingeteilt, so, dass die Streuung der Grauwerte innerhalb jeder Klasse möglichst klein und zwischen den Klassen möglichst groß ist. Der Schwellwert T wird so gewählt, dass der Quotient $Q(T)$ von der Varianz zwischen den Klassen σ^2_{zw} zu Varianz innerhalb der Klassen σ^2_{in} maximal wird:

$$Q(T) = \frac{\sigma^2_{zw}(T)}{\sigma^2_{in}(T)} \rightarrow max$$

Ein Nachteil der globalen Schwellwertverfahren ist, dass diese Methoden sehr anfällig für Helligkeitsänderungen im Bild sind. Siehe Abbildung 2.2.

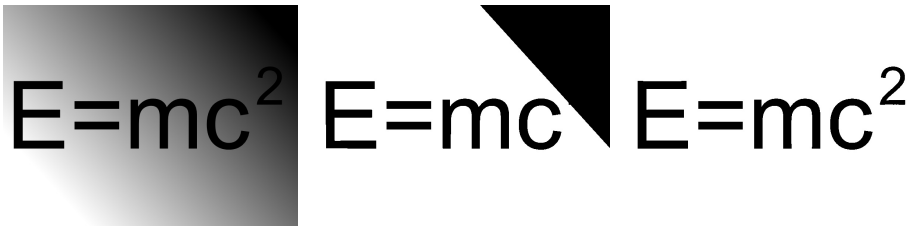
Lokale Schwellwertverfahren arbeiten mit Bildregionen. Das Originalbild wird in N Regionen eingeteilt, und für jede Region $R_i, (i = 1, ...N)$ wird ein Schwellwert T_i festgelegt, d.h. alle Regionen können unabhängig von einander binarisiert werden. Die Vorschrift in diesem Fall ist:

$$B(x,y) = T_{lokal}(x,y) = \begin{cases} 1, & falls\ \mathbf{I}(x,y) > T_i \\ 0, & falls\ \mathbf{I}(x,y) \leq T_i \end{cases} \quad \forall (x,y) \in R_i \tag{2.5}$$

Es kann für Schwellwertbestimmung auch die Nachbarschaft $N(x,y)$ jedes Pixels (x,y) betrachtet werden. Es wird eine Funktion $T(N(x,y))$ von der Nachbarschaft berechnet und dadurch ein Schwellwert bestimmt:

$$B(x,y) = T_{adaptiv}(x,y) = \begin{cases} 1, & falls\ \mathbf{I}(x,y) > T(N(x,y)) \\ 0, & falls\ \mathbf{I}(x,y) \leq T(N(x,y)) \end{cases} \tag{2.6}$$

Diese Klasse von Schwellwertverfahren heißt **adaptiv** (in einigen Quellen - *dynamisch*) weil es für jedes Pixel ein eigener Schwellwert berechnet wird. Dadurch ist der Rechenaufwand bei adaptiven Verfahren erheblich höher als bei globalen und lokalen. Aber ein wichtiger Vorteil von adaptiven Verfahren ist ihr sehr stabiles Verhalten gegenüber lokalen Helligkeitsänderungen im Bild (siehe Abbildung 2.2).



a) Ausgangsbild b) Binärbild, globale Methode c) Binärbild, adaptive Methode

Abbildung 2.2: *Vergleich von Schwellwertverfahren*

In den meisten Fällen liefern die adaptiven Schwellwertverfahren bessere Ergebnisse als globale oder lokale, aber wenn es im Voraus bekannt ist, dass im Bild keine lokalen Helligkeitsänderungen vorkommen, sind globale oder lokale Methoden auf Grund ihrer besseren Performance den adaptiven vorzuziehen.

2.1.4 Histogramme von Bildern

In Bildverarbeitung stellt ein Histogramm die Häufigkeitsverteilung der Pixelwerte eines Bildes dar. Um ein Histogramm zu erstellen muss zuerst für jeden möglichen Pixelwert i die Anzahl der Pixel mit diesem Wert - n_i aufgezählt werden. In der Regel werden Histogramme normalisiert, und statt absoluter Häufigkeiten n_i werden relative Häufigkeiten p_i verwendet:

$$p_i = \frac{n_i}{n} \tag{2.7}$$

n_i ist die Anzahl der Pixel mit dem Wert i, n ist die Gesamtanzahl der Pixel im Bild.

Durch Analyse von Bildhistogrammen können verschiedene Charakteristiken von Bildern berechnet werden, wie z.B. in dieser Arbeit häufig verwendete *Entropie* als Qualitätsmaß eines Bildes. Außerdem existieren verschiedene Bildverarbeitungsmethoden, die durch Veränderung von Histogrammen das Originalbild verbessern können.

Es kann z.B. durch Umverteilung der Pixelwerte die Helligkeit eines Bildes erhöht oder der Kontrast eines Bildes verbessert werden (Abbildung 2.3-2.5). Es existieren auch Algorithmen, die durch Analyse und Veränderung von Histogrammen automatisch den Kontrast eines Bildes verbessern können.

2.1.5 Lineare Filter und Konvolution

In Bildverarbeitung wird ein Filter als eine Operation verstanden, die für ein Ausgangsbild mit Hilfe einer gegebenen mathematischen Abbildung ein Ausgabebild erzeugt. Ein Filter ist durch seine Maske (manchmal auch *Filterkern* genannt) definiert. Die Größe der Filtermaske bestimmt auch die Größe der Nachbarschaft eines Pixels, auf die die gegebene Abbildung angewendet wird. Bezüglich der Größe einer Filtermaske ist es wichtig zu bemerken, dass ihre Dimensionen unbedingt ungerade sein müssen, d.h. für eine Maske der Größe $m \times n$ gilt: $m = 2a + 1, n = 2b + 1$ mit $a, b > 0$. Somit ist die minimale Größe einer Maske 3×3.

Das Resultat der Anwendung einer Filtermaske auf ein Pixel (x, y) wird *Impulsantwort* genannt.

Bei linearen Filtern ist die Impulsantwort gleich Summe der Produkte von Filterkoeffizienten mit den entsprechenden Pixelwerten. Z.B. für eine Filtermaske W der Größe 3×3

$$W = \begin{bmatrix} w(-1,-1) & w(0,-1) & w(1,-1) \\ w(-1,0) & w(0,0) & w(1,0) \\ w(-1,1) & w(0,1) & w(1,1) \end{bmatrix}$$

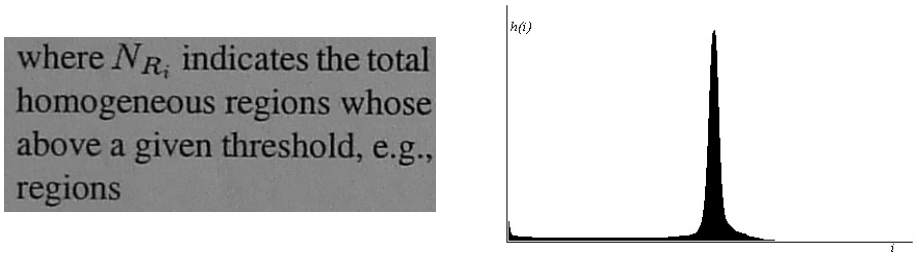


Abbildung 2.3: *Originalbild und sein Histogramm*

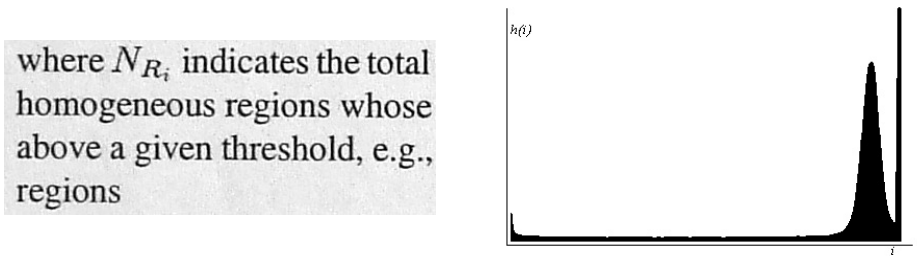


Abbildung 2.4: *Bild mit verbesserter Helligkeit und sein Histogramm*

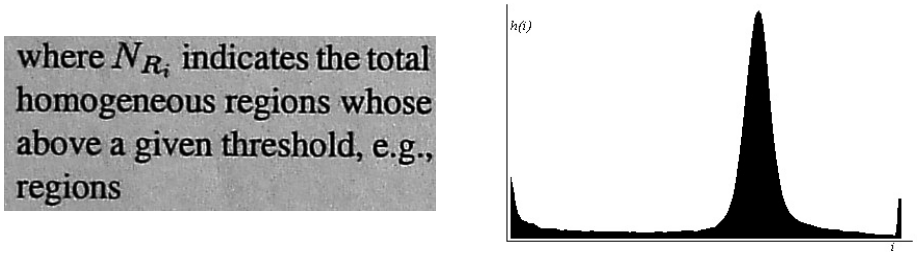


Abbildung 2.5: *Bild mit verbessertem Kontrast und sein Histogramm*

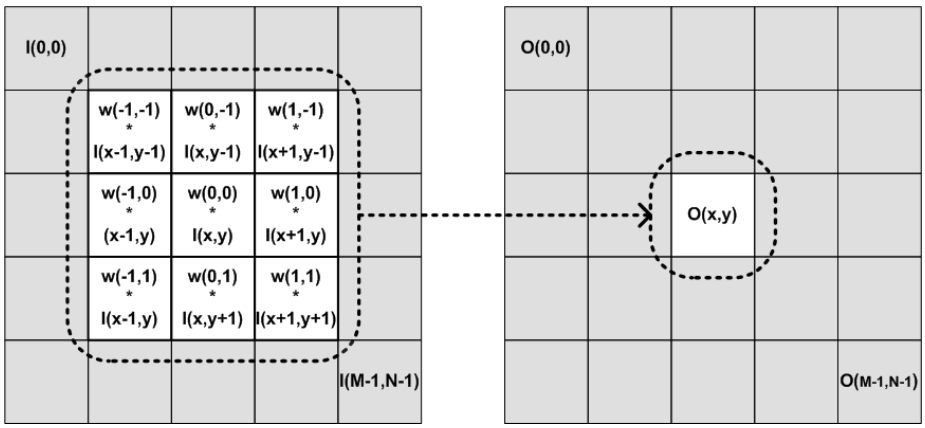


Abbildung 2.6: *Anwendung einer Maske*

und ein Pixel (x, y) mit entsprechender Nachbarschaft

$$\mathbf{I}_{3\times 3}(x,y) = \begin{bmatrix} \mathbf{I}(x-1,y-1) & \mathbf{I}(x,y-1) & \mathbf{I}(x+1,y-1) \\ \mathbf{I}(x-1,y) & \mathbf{I}(x,y) & \mathbf{I}(x+1,y) \\ \mathbf{I}(x-1,y+1) & \mathbf{I}(x,y+1) & \mathbf{I}(x+1,y+1) \end{bmatrix}$$

die Impulsantwort O ist:

$$\begin{aligned} O(x,y) &= w(-1,-1)\mathbf{I}(x-y,y-1) + w(-1,0)\mathbf{I}(x-1,y) + ... \\ &\quad + w(0,0)\mathbf{I}(x,y) + ... + w(1,1)\mathbf{I}(x+1,y+1) \end{aligned}$$

Im Allgemeinen lässt sich ein linearer Filter der Größe $m \times n$ für ein Bild $\mathbf{I}$ der Größe $M \times N$ nach folgender Formel berechnen

$$F(x,y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i,j)\mathbf{I}(x+i,y+j) \qquad (2.8)$$
$$\forall\ 0 \le x < M,\ 0 \le y < N$$

dabei ist $a = \frac{m-1}{2}$, $b = \frac{n-1}{2}$. m und n sind, wie schon besprochen, ungerade. Wie aus der Formel ersichtlich ist, ist die Filtermaske im Pixel (x, y) zentriert. Das bedeutet aber, dass für die Pixel, die nah am Rand sind (mit Abstand kleiner a auf x-Achse oder/und kleiner b auf y-Achse), nicht die ganze Maske angewendet werden kann. Dieser spezielle Fall muss gesondert betrachtet werden.

Wenn die Filtermaske nah am Rand ist, so dass nicht für alle Filterkoeffizienten Ursprungspixel definiert sind, muss eine Lösung gefunden werden. Eine der Möglichkeiten besteht darin, den Definitionsbereich im Ursprungbild so ein-

zugrenzen, dass die Filtermaske bei allen möglichen Positionen nicht über die Ränder des Bildes geht. Das führt aber dazu, dass das Ausgabebild kleiner als das Ursprungsbild wird. Für ein Bild $M \times N$ und eine Maske $m \times n$ wird die Größe des Resultatbildes $(M - m + 1) \times (N - n + 1)$, aber dafür werden alle Pixel mit der ganzen Filtermaske bearbeitet.

Wenn die Ausgabe genau so groß wie das Originalbild sein muss, werden andere Methoden verwendet, die entweder nicht die ganze Filtermaske anwenden, oder des Ursprungsbild an Rändern erweitern.

Bei den Methoden, die nicht die ganze Filtermaske anwenden, wird auf die Randpixel nur der Teil der Maske angewendet, der sich im Bild befindet (siehe Abbildung 2.7).

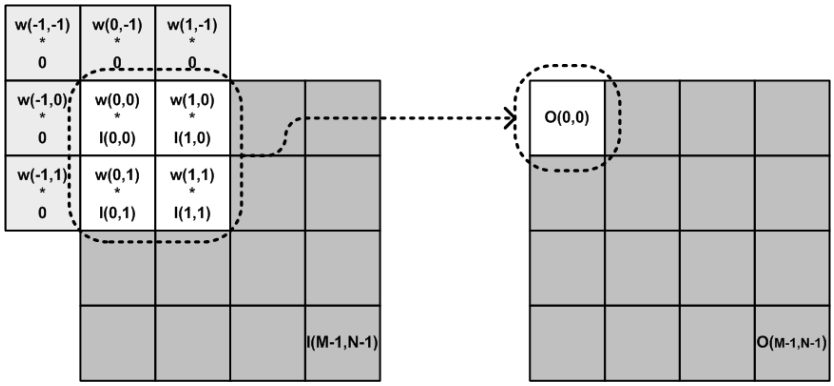


Abbildung 2.7: *Filtermaske am Rand des Bildes*

Bei den Methoden die das Ursprungsbild erweitern, werden an Bildrändern neue Pixel hinzugefügt, so dass ein Bild $M \times N$ für eine Maske $m \times n$ zu $(M+m-1) \times (N+n-1)$ erweitert wird. Dafür werden am linken und rechten Rand je $\frac{m-1}{2}$ neue Spalten und am oberen und unteren Rand je $\frac{n-1}{2}$ neue Zeilen angefügt. Für neue Pixel können die Werte von entsprechenden Randpixel genommen werden oder ein fester Wert aus dem Definitionsbereich der Pixelwerte. Wenn angefügte Pixel auf 0 gesetzt werden, sieht das Ergebnis genau so aus, wie bei der Methode, die nicht die ganze Filtermaske verwendet.

Lineare Filterung, so wie die in Gleichung 2.8 definiert wurde, ist nichts anderes als eine *zweidimensionale diskrete Faltung*. Per Definition lautet die Formel für zweidimensionale diskrete Faltung:

$$h(x, y) = (f * g)(x, y) = \sum_{i=-s}^{s} \sum_{j=-t}^{t} f(i, j)g(x + i, y + j) \tag{2.9}$$

dabei bezeichnet $(*)$ einen Faltungsoperatoren.

Faltung wird auch **Konvolution** genannt, die Maske wird dabei Konvolu-

tionskern oder einfach **Kern** genannt. Somit kann Konvolution eines Bildes $\mathbf{I}$ mit einem Kern H mit folgender Formel beschrieben werden:

$$F(x, y) = (H * \mathbf{I})(x, y) \tag{2.10}$$

mit H als Funktion auf Kernkoeffizienten und $\mathbf{I}$ als Funktion auf Pixelwerten. Oft wird auch abkürzende Schreibweise verwendet:

$$F = H * \mathbf{I} \tag{2.11}$$

Unter Berücksichtigung, dass Faltung kommutativ ist, ist auch andere Schreibweise - $F = \mathbf{I} * H$ möglich.

2.1.6 Erzeugung von Bildpyramiden

Eine Bildpyramide ist eine Sammlung von Bildern, die alle aus einen Ursprungsbild erzeugt werden. Dafür werden in jeder nächsten Pyramidenstufe die Bildgrößen jeweils halbiert bis eine vorgegebene Größe erreicht wird.

Es gibt zwei Arten von Bildpyramiden, die oft in Literatur beschrieben werden: *Gauß-* und *Laplace-Pyramiden*. Gauß-Pyramiden werden für *downsampling*, d.h. für Erstellung von kleineren Bildern, verwendet. Laplace-Pyramiden werden benutzt, um aus einem Bild einer Stufe der Gauß-Pyramide ein Bild der nächst feineren Stufe zu rekonstruieren.

Um aus der Schicht i der Gauß-Pyramide die nächste Schicht $i+1$ zu erstellen, wird das Bild aus der Schicht i mit dem Gauß-Kern der Größe 5×5 gefaltet und im Resultatbild jede zweite Zeile und Spalte entfernt. Somit entsteht ein kleineres Bild mit jeweils halbierten Größen.

Der Gauß-Kern für zweidimensionale Faltung wird wie folgt gebildet:

$$\begin{aligned} \mathcal{G}_{5\times5} &= \frac{1}{16} \cdot \begin{bmatrix} 1 \\ 4 \\ 6 \\ 4 \\ 1 \end{bmatrix} \cdot \frac{1}{16} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix} = \\ &= \frac{1}{256} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} \end{aligned} \tag{2.12}$$

Allgemein kann aus einem Bild $\mathbf{G}_i$ der Schicht i das nächst kleinere Bild der

Schicht $i+1$ nach folgender Formel erstellt werden:

$$\mathbf{G}_{i+1}(x,y) = (\mathcal{G}_{5\times5} * \mathbf{G}_i)\,(2x, 2y) \tag{2.13}$$

Wenn $\mathbf{G}_i$ die Größe $M \times N$ hat, dann ist die Größe von $\mathbf{G}_{i+1}$ dementsprechend $M/2 \times N/2$.

Bildpyramiden werden oft in der Bildverarbeitung benutzt, meistens in Verbindung mit einer *grob-zu-fein* Technik. Dabei wird zuerst das kleinste Bild in der Pyramide verarbeitet und dann für jeweils nächst größere Bild die Verarbeitung unter Benutzung von Ergebnissen des vorhergehenden Bildes durchgeführt.

2.2 Methoden der Bildfusion

In diesem Abschnitt werden die Methoden betrachtet, die aus einer Reihe von unterschiedlich belichteten Aufnahmen einer Szene ein Ausgabebild mit höchstem Informationsinhalt erzeugen. Es wird dabei vorausgesetzt, dass die Aufnahmen mit Hilfe von Alignmentkorrektur-Techniken (Kapitel 3) in die bestmögliche lokale Korrespondenz zueinander gebracht sind.

2.2.1 Hochkontrastbilder

Als *Hochkontrastbilder* (engl. *High Dynamic Range*, *HDR*) werden Bilder bezeichnet, die große Helligkeitsunterschiede in einer Aufnahme speichern können [7]. Analog dazu können konventionelle Digitalbilder auch als *LDR (Low Dynamic Range)* Bilder bezeichnet werden. In LDR-Bildern werden nur 256 Helligkeitswerte pro Farbkanal gespeichert und diese Farbtiefe reicht oftmals nicht aus, um Helligkeitsunterschiede, die in natürlichen Szenen vorkommen, wiederzugeben. Die Verwendung von nur 256 Helligkeitsstufen für Farbtiefe begründet sich auch darin, dass Bildschirme und Druckmedien nicht fähig sind, höhere Farbtiefen darzustellen.

Der typische *Dynamikbereich* (auch *Dynamikumfang* und *Kontrastumfang* genannt) einer (von der Kamera aus) sichtbarer Umgebung hat Helligkeitsunterschiede in Größenordnung von $10^4 : 1$, d.h. die größte *Leuchtdichte* ist 10^4 mal größer als die kleinste. Leuchtdichten von typischen Lichtverhältnissen sind in der Tabelle 2.1 dargestellt. Der Dynamikumfang wird noch höher, wenn innerhalb einer Szene eine Lichtquelle direkt sichtbar ist oder sowohl ein Innenraum als auch ein vom Sonnenlicht erhellter Außenbereich zu sehen sind (Abbildung 2.8). Das menschliche Auge ist in der Lage, sich unterschiedlichen Lichtverhältnissen anzupassen und kann einen Dynamikbereich von 10 Größenordnungen (10^{10}) wahrnehmen. Innerhalb einer Szene kann ein Dynamikbereich von bis zu 10^5 gleichzeitig wahrgenommen werden.

Lichtverhältnisse	Leuchtdichte (cd/m^2)
Sternenlicht	10^{-3}
Mondlicht	10^{-1}
Innenraum	10^{2}
LED-Außenbildschirm	5×10^{3}
60-Watt Glühbirne	120×10^{3}
Sonne am Morgen/Abend	6×10^{6}
Sonne am Mittag	$1,6 \times 10^{9}$
Helligkeit von konventionellen Bildschirmen	$2 \times 10^{2} - 5 \times 10^{2}$

Tabelle 2.1: *Helligkeitswerte für bestimmte Lichtverhältnisse*

Abbildung 2.8: *Aufnahme einer Szene mit erhelltem Außenbereich. Links - eine Aufnahme mit herkömmlicher Kamera. Rechts - ein mit HDR-Techniken erzeugtes Bild (Quelle - [7]).*

Wegen nicht ausreichenden Dynamikbereichs bei herkömmlichen Digitalkameras leiden häufig die damit erzeugten Fotos an Über- und Unterbelichtungen. Die mit HDR-Techniken erzeugten Fotos haben einen größeren Dynamikbereich, der alle in der aufgenommenen Szene vorkommenden Helligkeiten erfassen kann.

Es gibt zwei Möglichkeiten ein HDR-Bild zu erstellen - entweder mit bereits existierenden speziellen Kameras, die Hochkontrastbilder aufnehmen können, oder aus einer Reihe von unterschiedlich belichteten Aufnahmen mit geringem Dynamikumfang. Im zweiten Fall werden die einzelnen Aufnahmen miteinander kombiniert, so dass der Kontrast aufgenommener Szene vollständig erfasst wird.

Tone Mapping

Um ein Hochkontrastbild auf einem Medium mit geringerem Dynamikumfang (Bildschirm, Papier) darzustellen, muss sein großer Kontrastumfang auf einen kleineren Dynamikbereich abgebildet werden. Dieser Schritt der *Dynamikkompression* wird *Tone Mapping* genannt. Besonders wichtig dabei ist, dass die Details, die auf einem HDR-Bild in dunklen und hellen Regionen gut sichtbar sind, auch nach dem Tone Mapping beibehalten bleiben, so gut es möglich ist.

Viele Tone-Mapping-Verfahren benutzen Erkenntnisse über die menschliche

visuelle Wahrnehmung, um ein LDR-Bild zu erzeugen, das möglichst naturgetreu erscheint. Die Zahlreichen *Tone-Mapping-Operatoren* lassen sich in vier Klassen einteilen:

- Bei *globalen Operatoren* wird eine Funktion, die jedem HDR-Wert einen dynamikkomprimierten Wert zuweist, auf jedes Pixel angewendet.
- Im Gegensatz dazu wird bei *lokalen Operatoren* diese Funktion für jedes Pixel in Abhängigkeit von seiner Umgebung variiert.
- Bei *frequenzbasierten Operatoren* wird der Dynamikumfang von Bildregionen je nach Ortsfrequenz reduziert.
- Die *gradientenbasierten Operatoren* schwächen die Helligkeitsgradienten für jedes Pixel des Ausgangsbildes ab.

Prinzipiell andere Verfahren, um eine Reihe von unterschiedlich belichteten Aufnahmen zu einem LDR-Bild zu kombinieren, sind *Exposure Blending* und *Exposure Fusion*. Hierbei ist es wichtig, dass über- und unterbelichtete Bereiche vermieden werden und mehr Details im Ausgabebild erhalten bleiben. Bei diesen Verfahren werden keine HDR-Bilder mit höherem Kontrastumfang erzeugt und dementsprechend kein Tone Mapping notwendig.

Exposure Blending

Bei *Exposure Blending* werden ausschließlich Methoden der Bildbearbeitung verwendet. Es werden dabei die kontrastreichsten Stellen von jeder Aufnahme ausgewählt und im Ausgabebild kombiniert. Die Stelle, die auf einer Aufnahme überbelichtet ist, wird mit der entsprechender Stelle aus der nächstdunkleren Aufnahme ersetzt. Eine unterbelichtete Stelle wird dementsprechend mit einer Stelle aus der nächsthellerer Aufnahme ersetzt. Dies wird üblicherweise manuell mit Hilfe eines Bildbearbeitungsprogramms gemacht. Es existieren auch automatische Methoden der Exposure-Blending, die brauchen aber die korrekten Eigenschaften der verwendeten Kamera, wie z.B. eine “Kamerakurve”. Es ist eine Charakteristik, die angibt, wie die Kamera auf unterschiedliche Helligkeiten reagiert.

Exposure Fusion

Exposure Fusion Methoden können sowohl mit Bildregionen, als auch mit einzelnen Pixeln arbeiten. Bei pixelbasierten Methoden wird für jedes Pixel ein Gewicht berechnet und basierend auf den Pixelgewichten wird entschieden, ob ein Pixel in das Ausgabebild mit reingenommen wird, und wenn ja, wie stark dieses Pixel das Endergebnis beeinflusst. Die Berechnung von Gewichten kann

a) Eine Reihe von Aufnahmen mit unterschiedlichen Belichtungszeiten - unterbelichtet, normal und überbelichtet.

b) *Das entsprechende HDR-Bild nach dem Tone Mapping*

Abbildung 2.9: *Eine Belichtungsreihe und entsprechendes HDR-Bild (Fotos von Jacques Joffre).*

auf Werten aller an der gleichen Position stehenden Pixel aller Aufnahmen basieren, oder es kann eine lokale Nachbarschaft innerhalb einer Aufnahme betrachtet werden. In einigen Methoden werden auch globale Charakteristiken einer Aufnahme, wie z.B. mittlere Helligkeit, berücksichtigt. Stärkere Gewichtung von Pixel mit bestimmten Eigenschaften, wie z.B. höherer Kontrast oder Farbsättigung, führt dazu, dass das Ausgabebild aus den "besten" Pixeln jeder Aufnahme kombiniert wird.

Regionenbasierte Verfahren basieren auf dem gleichen Prinzip - Gewichtung einzelner Bildteile und anschließende Fusion des Ausgabebildes entsprechend berechneten Gewichten. In diesem Fall wird aber nicht mit einzelnen Pixeln, sondern mit Regionen bestimmter Größe gearbeitet. Jede Aufnahme wird in mehrere Regionen gleicher Größe geteilt, für jede Region werden bestimmte Charakteristiken berechnet, die dann ein Gewicht bilden. Anschließend wird entsprechend den Gewichten aus den "besten" Bildregionen ein Ausgabebild

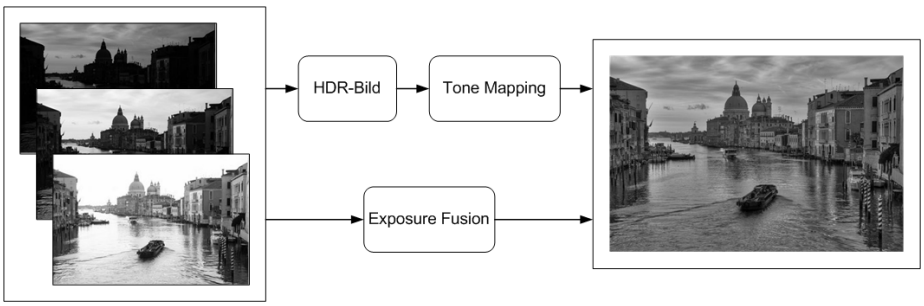


Abbildung 2.10: *Erstellung eines Hochkontrastbildes mit HDR-Techniken und mit Exposure-Fusion*

kombiniert.

Ein großer Vorteil von Exposure-Fusion gegenüber Exposure-Blending ist, dass es keine Kameraeigenschaften notwendig sind, um Exposure-Fusion anzuwenden. Alle für die Fusion benötigte Parameter können aus der Analyse von Aufnahmen gewonnen werden.

Im Vergleich zu HDR-Techniken ist Image Fusion schneller und effizienter, weil es kein HDR-Bild erzeugt werden muss und demzufolge kein Tone Mapping notwendig ist. Ein anderer wichtiger Vorteil besteht darin, dass Fusion-Methoden nicht nur mit ganzen Bilder, sondern auch mit einzelnen Pixel oder Bildregionen arbeiten können. Dadurch lassen sich nicht nur *unterschiedlich belichtete* Aufnahmen zu einem Bild kombinieren, es kann auch aus einer Reihe von Aufnahmen mit *unterschiedlichen Schärfentiefen* ein Bild erzeugt werden, auf dem alle Objekte der aufgenommenen Szene sich im Schärfebereich befinden. Diese Technik wurde auch in mehreren Arbeiten untersucht und beschrieben, unter anderen in Artikeln über *Multifocus Fusion* von S. Li and B. Yang [11], W. Huang and Z. Jing [10] und S. Li, J. T. Kwok and Y. Wang [12].

Eine besonders wichtige Rolle bei der Bildfusion spielen Kriterien, nach welchen Pixel und Bildregionen gewichtet werden. Im nächsten Abschnitt werden einige Qualitätsmaße besprochen, die als Kriterien für Auswahl bestimmter Bildregionen verwendet werden. In weiteren Abschnitten werden pixel- und regionenbasierte Methoden der Bildfusion näher betrachtet.

2.2.2 Qualitätsmaße der Aufnahmen

Um Güte einer Aufnahme zu bewerten, können verschiedene Qualitätsmaße verwendet werden. Es können sowohl ganze Bilder als auch einzelne Bildregionen bewertet werden. Verschiedene Qualitätsmaße können für verschiedene Zwecke benutzt werden: z.B. *Spatial Frequency* wird für Ermittlung von bestfokussierten Aufnahmen verwendet [10, 11, 12, 14]. *Entropie* kann dazu dienen, den Infor-

mationsinhalt einer Aufnahme zu berechnen und somit die informationsreichste Aufnahme aus einer Serie auszuwählen.

Durch Berechnung der Qualitätsmaße von ganzen Bildern kann das am besten fokussierte oder das informationsreichste Bild aus einer Reihe von Aufnahmen ausgewählt werden. Durch Berechnung der Qualitätsmaße von Bildregionen werden die "besten" Regionen ermittelt, aus denen mit Bildfusionsverfahren das Resultatbild erstellt werden kann.

Entropie

Entropie ist ein passendes Maß für den Informationsinhalt von Bildern. Für die Berechnung der Entropie werden zuerst relative Häufigkeiten der Pixelwerte berechnet - eine Häufigkeitsverteilung der Grauwerte des Bildes:

$$p_i = \frac{n_i}{n} \tag{2.14}$$

n_i ist die Anzahl der Pixel mit dem Grauwert i, n ist die Gesamtanzahl der Pixel im Bild.

Die Entropie wird dann nach folgender Formel berechnet:

$$E = \sum_{i=0}^{255} -p_i \log(p_i) \tag{2.15}$$

Die Variable i nimmt dabei alle möglichen Grauwerte an, und p_i ist die Auftrittswahrscheinlichkeit des Grauwertes i.

Entropie als Qualitätsmaß wird für die Ermittlung von informationsreichsten Bildregionen in der Methode von Goshtasby verwendet. [9].

Maße der Fokussierung

Ein Maß der Fokussierung ist so definiert, dass es für das am besten fokussierte Bild den größten Wert liefert, und mit der Abnahme der Fokussierung nimmt auch das Maß ab [10, 13]. Somit kann durch Berechnung des Maßes das best fokussierte Bild ermittelt werden (Beispiel siehe Abbildung 2.11). Es wurden außerdem einige Anforderungen an die Maße der Fokussierung formuliert [10]:

1. unabhängig von dem Bildinhalt
2. monoton bezüglich der Unschärfe
3. es muss nur einen Maximalwert geben
4. große Änderungen im Wert bei Variation der Bildschärfe
5. minimaler Aufwand der Berechnung

a) *gut fokussiertes Bild*

b) *defokussiertes Bild*

Abbildung 2.11: *Beispiele für Bilder mit hohem* ***a)*** *und niedrigem* ***b)*** *Fokussierungsmaß*

6. robust gegen Rauschen

Spatial Frequency

Dieses Qualitätsmaß wurde in mehreren Arbeiten für Ermittlung von schärfsten, bestfokussierten Bildern verwendet [10, 11, 12, 14]. Für ein Bild $\mathbf{I}$ mit Breite N und Höhe M ist Spatial Frequency wie folgt definiert:

$$SF = \sqrt{(RF)^2 + (CF)^2} \tag{2.16}$$

Wobei RF ist *Row Frequency*:

$$RF = \sqrt{\frac{1}{M \times N} \sum_{m=0}^{M-1} \sum_{n=1}^{N-1} \left[\mathbf{I}(m,n) - \mathbf{I}(m,n-1)\right]^2}$$

CF ist *Column Frequency*:

$$CF = \sqrt{\frac{1}{M \times N} \sum_{m=1}^{M-1} \sum_{n=0}^{N-1} \left[\mathbf{I}(m,n) - \mathbf{I}(m-1,n)\right]^2}$$

Ein wichtiger Vorteil von *Spatial Frequency* ist seine schnelle Berechenbarkeit für die nur zwei Durchläufe gebraucht werden.

Varianz

Ein einfaches und leicht berechenbares Qualitätsmaß ist die Varianz von Grauwerten im Bild

$$Var = \frac{1}{M \times N} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [\mathbf{I}(m,n) - \mu]^2 \tag{2.17}$$

Wobei μ ist der Mittelwert aller Grauwerte des Bildes:

$$\mu = \frac{1}{M \times N} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathbf{I}(m,n)$$

Energy of Image Gradient

Ist ein häufig benutztes Maß der Fokussierung, es verwendet Bildgradienten für die Ermittlung von den best fokussierten Bildern [10, 15]:

$$EOG = \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \left(\frac{d\mathbf{I}(x,y)}{dx} \right)^2 + \left(\frac{d\mathbf{I}(x,y)}{dy} \right)^2 \tag{2.18}$$

Die Differentialquotienten sind dabei wie folgt definiert:

$$\frac{d\mathbf{I}(x,y)}{dx} = \mathbf{I}(x,y) - \mathbf{I}(x-1,y)$$

$$\frac{d\mathbf{I}(x,y)}{dy} = \mathbf{I}(x,y) - \mathbf{I}(x,y-1)$$

Es kann auch eine vereinfachte Variante von diesem Qualitätsmaß verwendet werden bei der die Differentialquotienten nicht quadriert werden, sondern ein Betrag davon genommen wird:

$$EOG = \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \left| \frac{d\mathbf{I}(x,y)}{dx} \right| + \left| \frac{d\mathbf{I}(x,y)}{dy} \right| \tag{2.19}$$

Die Formel hat den Vorteil, dass sie schneller berechnet werden kann. Ihre Effizienz wurde auch in der Praxis bewiesen [15].

Tenengrad

Tenenbaum hat eine Methode für die Berechnung eines Fokussierungsmaßes entwickelt, die die Berechnung von Bildgradienten mit anschließender Anwendung von einem Schwellenwert kombiniert [10, 15].

$$Tenengrad = \sum_{x=1}^{N-2} \sum_{y=1}^{M-2} [\nabla S(x,y)]^2 \quad für \;\; \nabla S(x,y) > T \tag{2.20}$$

T ist dabei ein vorgegebener Schwellenwert und $\nabla S(x,y)$ - das Resultat der Anwendung von Sobel-Operatoren auf das Bild $\mathbf{I}$ im Pixel (x,y)

$$\nabla S(x,y) = \sqrt{\nabla S_x(x,y)^2 + \nabla S_y(x,y)^2}$$

Die Sobel-Operatoren $\nabla S_x(x,y)$ und $\nabla S_y(x,y)$ sind definiert als:

$$\nabla S_x(x,y) = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * \mathbf{I} \tag{2.21}$$

$$\nabla S_y(x,y) = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * \mathbf{I} \tag{2.22}$$

Der Operator $(*)$ bezeichnet hier einen 2-dimensionalen Faltungsoperatoren.

Sobel-Operatoren können auch anders, in Form von Gleichungen, ausgedrückt werden:

$$\begin{aligned} \nabla S_x(x,y) = \quad & -\mathbf{I}(x-1,y-1) && +\mathbf{I}(x+1,y-1) \\ & -2\mathbf{I}(x-1,y) && +2\mathbf{I}(x+1,y) \\ & -\mathbf{I}(x-1,y+1) && +\mathbf{I}(x+1,y+1) \end{aligned} \tag{2.23}$$

$$\begin{aligned} \nabla S_y(x,y) = & \\ -\mathbf{I}(x-1,y-1) \quad & -2\mathbf{I}(x,y-1) \quad -\mathbf{I}(x+1,y-1) \\ +\mathbf{I}(x-1,y+1) \quad & +2\mathbf{I}(x,y+1) \quad +\mathbf{I}(x+1,y+1) \end{aligned} \tag{2.24}$$

Die Verwendung von einem Schwellenwert für Sobel-Operatoren erhöht die Strenge der Tenengrad-Methode. Pixel mit kleinen Gradientenwerten haben eine relativ homogene Umgebung und stellen für die Methode kein Interesse dar. In die Berechnung vom Qualitätsmaß werden nur die Pixel einbezogen, dir eine ausreichend große Helligkeitsänderung in der Umgebung aufweisen und somit die stärksten Kanten darstellen.

Energy of Laplacian

Energy of Laplacian ist als Quadrat des Laplace-Operators [10]definiert:

$$EOL = \sum_{x=1}^{N-2} \sum_{y=1}^{M-2} \left[\Delta f(x,y)\right]^2 \tag{2.25}$$

wobei der Laplace-Operator als eine Summe der zweiten Ableitungen definiert ist [1]:

$$\Delta f(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{2.26}$$

Damit diese Gleichung in der Bildverarbeitung verwendet werden kann, müssen die partiellen Ableitungen diskret dargestellt werden. Die meist verwendete diskrete Approximierung der zweiten Ableitungen ist:

$$\frac{\partial^2 f}{\partial x^2} = f(x-1,y) - 2f(x,y) + f(x+1,y) \tag{2.27}$$

$$\frac{\partial^2 f}{\partial y^2} = f(x,y-1) - 2f(x,y) + f(x,y+1) \tag{2.28}$$

Eine diskrete Approximation des Laplace-Operators wäre dann:

$$\begin{array}{ccc} \Delta f(x,y) = & & \\ & f(x,y-1) & \\ +f(x-1,y) & -4f(x,y) & +f(x+1,y) \\ & +f(x,y+1) & \end{array} \tag{2.29}$$

Daraus ergibt sich eine Darstellung des Laplace-Operators mit einer Faltungsmaske:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} * \mathbf{I} \tag{2.30}$$

Der $(*)$-Operator ist hier, auch wie bei (2.21) und (2.22), ein zweidimensionaler Faltungsoperator.

Bei der Berechnung von Energy-of-Laplacian wird aber eine andere, modifizierte, Faltungsmaske verwendet [10]:

$$\begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix} * \mathbf{I} \tag{2.31}$$

Die Maske in (2.30) reagiert nur auf horizontale und vertikale Kanten, d.h. sie arbeitet mit der 4-Nachbarschaft des Zentralpixels. Die modifizierte Variante arbeitet mit der 8-Nachbarschaft, es werden also auch die diagonalen Kanten betrachtet. Der Einfluss von horizontalen und vertikalen Kanten ist aber höher

als von diagonalen.

Sum-modified Laplacian

Bei der Berechnung von Laplacian kann es manchmal vorkommen, dass die partiellen Ableitungen vom Betrag gleich sind, aber unterschiedliche Vorzeichen haben und in der Summe eine Null ergeben. Wie im folgenden Beispiel:

$$\mathbf{I} = \begin{bmatrix} 0 & 1 & 0 \\ 3 & 2 & 3 \\ 0 & 1 & 0 \end{bmatrix}$$

$\mathbf{I}$ ist hier ein Bild der Größe 3×3. Berechnung des Laplace-Operators im Pixel $(1, 1)$ ergibt:

$$\frac{\partial^2 \mathbf{I}}{\partial x^2}(1,1) = \mathbf{I}(0,1) - 2\mathbf{I}(1,1) + \mathbf{I}(2,1) = 3 - 2 * 2 + 3 = +2$$

$$\frac{\partial^2 \mathbf{I}}{\partial y^2}(1,1) = \mathbf{I}(1,0) - 2\mathbf{I}(1,1) + \mathbf{I}(1,2) = 1 - 2 * 2 - 1 = -2$$

$$\Delta\mathbf{I}(1,1) = \frac{\partial^2 \mathbf{I}}{\partial x^2}(1,1) + \frac{\partial^2 \mathbf{I}}{\partial y^2}(1,1) = 2 - 2 = 0$$

Bei Texturbildern tritt dies besonders oft auf, und es führt dazu, dass Laplacian als Fokussierungsmaß solche Bilder als weniger fokussierte bewertet, obwohl es in Wirklichkeit nicht der Fall ist. Eine Lösung dieses Problems wurde in der Arbeit von Nayar, Nakagawa vorgeschlagen [16]. Es wurde ein *modified Laplacian* eingeführt:

$$\Delta\mathbf{I}_m = \left|\frac{\partial^2 f}{\partial x^2}\right| + \left|\frac{\partial^2 f}{\partial y^2}\right| \tag{2.32}$$

Eine diskrete Approximierung von diesem Ausdruck ist:

$$\begin{aligned} \Delta\mathbf{I}_m(x,y) &= |\mathbf{I}(x-1,y) - 2\mathbf{I}(x,y) + \mathbf{I}(x+1,y)| \\ &+ |\mathbf{I}(x,y-1) - 2\mathbf{I}(x,y) + \mathbf{I}(x,y+1)| \end{aligned} \tag{2.33}$$

Außerdem, um die Formel an unterschiedlich große Texturelemente anzupassen, wurde eine Variable *step* eingeführt, die den Abstand zwischen Pixeln angibt:

$$\begin{aligned} ML(x,y) &= |\mathbf{I}(x-step,y) - 2\mathbf{I}(x,y) + \mathbf{I}(x+step,y)| \\ &+ |\mathbf{I}(x,y-step) - 2\mathbf{I}(x,y) + \mathbf{I}(x,y+step)| \end{aligned} \tag{2.34}$$

Die endgültige Formel für *Sum-modified-Laplacian* lautet dann:

$$SML = \sum_{x=step}^{N-step-1} \sum_{y=step}^{M-step-1} ML(x,y) \qquad für \;\; \Delta\mathbf{I}_m(x,y) > T \tag{2.35}$$

Der Schwellenwert T wird hier verwendet, um homogene Regionen, für die ML niedrige Resultate liefert, aus der Berechnung auszuschließen, genauso wie es bei der Tenengrad-Methode gemacht wurde.

2.2.3 Regionenbasierte Methoden

Bei regionenbasierten Methoden werden Bilder in rechteckige Regionen geteilt, für die dann ein Qualitätsmaß berechnet wird. Die Regionengröße ist ein wichtiger Optimierungsparameter, der stark das Ergebnis beeinflusst, er wird in der Regel in Abhängigkeit von der Bildgröße und manchmal auch in Abhängigkeit vom Bildinhalt ausgewählt. Es existieren außerdem Verfahren, die automatisch eine optimale Regionengröße bestimmen.

Es kann durchaus vorkommen, dass einige der best belichteten Bildbereiche keine rechteckige Form haben. In diesem Fall kann ihre Form mit rechteckigen Regionen kleinerer Größe approximiert werden. Kleinere Regionengrößen führen aber zu höherer Anzahl der zu analysierenden Regionen und folglich zu höherem Rechenaufwand. Deswegen ist es immer wichtig, eine optimale Regionengröße zu finden.

Für die Bestimmung von den best belichteten, best fokussierten oder nach einem anderen Kriterium besten Regionen, wird zuerst ein entsprechendes Qualitätsmaß gewählt. Dann wird für jede Region das Qualitätsmaß berechnet und anschließend ein Bild ermittelt, in dem diese Region das höchste Qualitätsmaß aufweist. Aus den ermittelten besten Regionen wird das Resultatbild zusammengestellt (Abbildung 2.12).

In einigen Fällen weist das aus Regionen zusammengestellte Bild sichtbare Übergänge zwischen einzelnen Bildregionen auf (Abbildung 2.14). Um solche Regionübergänge zu glätten, kann auf das Resultatbild eine Blending-Methode angewendet werden.

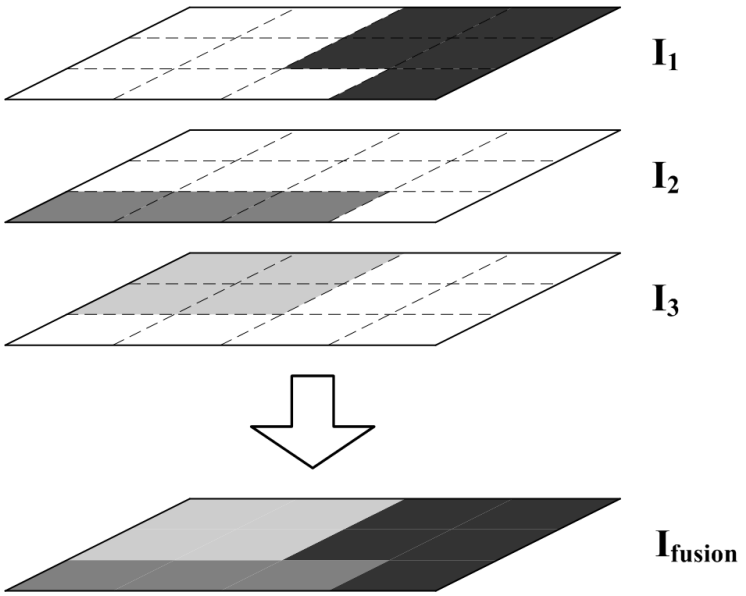
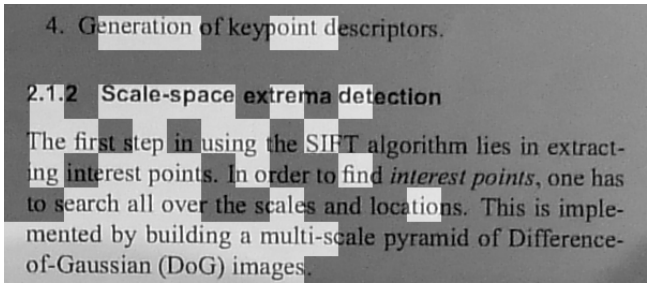


Abbildung 2.12: *Regionenbasierte Image-Fusion. Aus den "besten" Regionen der drei Bilder wird das Resultatbild zusammengestellt.*


4. Generation of keypoint descriptors.

2.1.2 Scale-space extrema detection

The first step in using the SIFT algorithm lies in extracting interest points. In order to find *interest points*, one has to search all over the scales and locations. This is implemented by building a multi-scale pyramid of Difference-of-Gaussian (DoG) images.

Abbildung 2.13: *Mosaikartiges Bild. Ergebnis der regionenbasierter Fusion mit Regionengröße* 32×32.

2.2.4 Blending von Regionen

Obwohl für die Zusammenstellung des Ausgabebildes nur Regionen mit höchstem Qualitätsmaß verwendet wurden, sind sie nicht aneinander angepasst. Deswegen sieht das Ergebnis oft mosaikartig aus (Abbildung 2.13).

Um Übergänge zwischen Regionen zu glätten, kann ein *Blending*-Verfahren angewendet werden. Es werden dabei einzelne Regionen so miteinander *verschmolzen*, dass es keine scharfen Grenzen mehr sichtbar sind (Abbildung 2.14). In dieser Arbeit wurde eine Blending-Methode verwendet, die von Goshtasby in seinem Artikel über Fusion von unterschiedlich belichteten Aufnahmen vorgeschlagen wurde [9].

Für Blending wird mehr Information benötigt als das mosaikartige Ausga-

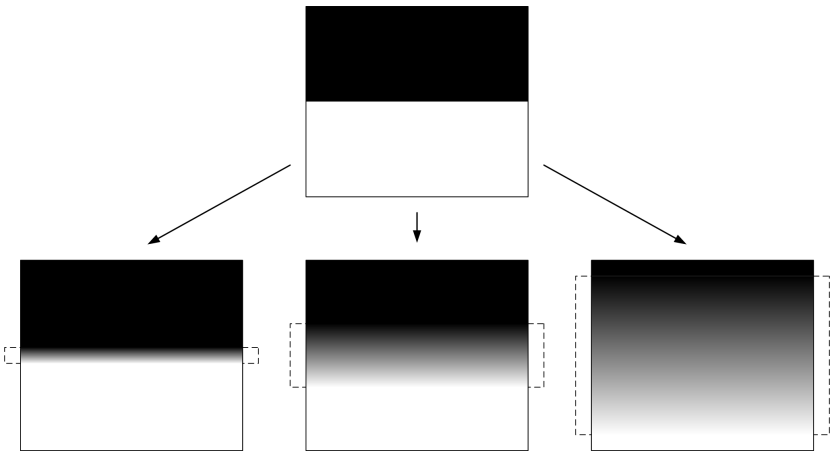


Abbildung 2.14: *Blending von Übergängen zwischen Bildregionen mit verschiedenen Breiten der Blending-Funktion*

bebild beinhaltet, dafür müssen alle Ursprungsbilder neu analysiert werden. Als Eingabe braucht die Methode Koordinaten des Zentrums für jede Bildregion und die Nummer des Ursprungsbildes, in dem für eine gegebene Region die besten Ergebnisse bei der Qualitätsmaßberechnung erzielt wurden. Ausgehend aus den Koordinaten der Regionen und den Nummern der entsprechenden Ursprungsbildern kann ein neues Ausgabebild erstellt werden, beim Blending wird es aber nicht aus Regionen zusammengestellt, sondern für jedes Pixel im Ausgabebild wird ein neuer Wert berechnet. Die neuen Pixelwerte werden aus gewichteten Summen gebildet, ein Summand ist dabei das Produkt eines Pixelwertes aus einem Ursprungsbild mit einem entsprechenden Gewicht.

Angenommen, es sind N Ursprungsbilder und jedes Bild wurde in $N_r \times N_c$ rechteckige Regionen geteilt. N_r ist dabei die Anzahl der Zeilen, N_c - Anzahl der Spalten. $\mathbf{I}'_{jk}$ bezeichnet das Ursprungsbild, das für die Region mit Index jk den höchsten Wert des Qualitätsmaßes liefert.

Für ein Pixel $\mathbf{O}(x, y)$ im Ausgabebild wird sein Wert nach folgender Formel berechnet:

$$\mathbf{O}(x,y) = \sum_{j=1}^{N_r} \sum_{k=1}^{N_c} W_{jk}(x,y)\mathbf{I}'_{jk}(x,y) \tag{2.36}$$

$\mathbf{I}'_{jk}(x, y)$ ist dabei ein Pixelwert im Ursprungsbild, und $W_{jk}(x, y)$ ist entsprechendes Gewicht, das mit Hilfe einer Blending-Funktion berechnet wird.

Für die Berechnung der Gewichte wird in der Mitte jedes Blocks eine Gauß-Funktion platziert. Die Funktion dient hier als eine Art inverse Abstandsfunktion; für die Pixel in der Nähe des Blockzentrums liefert sie den Maximalwert und je weiter desto kleiner der Funktionswert. Das höchste Gewicht bekommt

somit der Block, in dem sich das Pixel (x, y) befindet. D.h. obwohl in der Summenbildung alle "beste" Regionen ihren Anteil haben, der höchste Beitrag ist von dem Pixel, das geblendet wird.

Die Blending-Funktion $W_{jk}(x, y)$ für die Berechnung der Gewichte ist definiert als:

$$W_{jk}(x, y) = \frac{G_{jk}(x, y)}{\sum_{m=1}^{N_r} \sum_{n=1}^{N_c} G_{mn}(x, y)} \tag{2.37}$$

$G_{jk}(x, y)$ bezeichnet hier eine zweidimensionale Gauß-Funktion, die im Zentrum des Blocks jk platziert wurde. Sie ist definiert als:

$$G_{jk}(x, y) = e^{-\frac{1}{2}\frac{(x-x_{jk})^2+(y-y_{jk})^2}{\sigma^2}} \tag{2.38}$$

(x_{jk}, y_{jk}) sind hier Koordinaten des Zentrums des Blocks jk, und σ bezeichnet die Standardabweichung der Gauß-Funktion und wird auch *Breite* der Blending-Funktion genannt. σ ist ein wichtiger Optimierungsparameter, das stark das Aussehen des Ausgabebildes beeinflusst (Abbildung 2.14).

Hier ist noch zu beachten, dass die Anzahl der Terme in der Summe nicht gleich der Anzahl der Ursprungsbilder N ist, sondern gleich der Anzahl der Bildregionen - $N_r \times N_c$. Also, kleinere Bildregionen erhöhen den Rechenaufwand.

Es besteht aber eine Möglichkeit, die Berechnung zu optimieren. Bei der Summenbildung werden ohne Ausnahmen alle Blöcke betrachtet, z.B. wenn der blendende Pixel sich am linken Rand des Bildes befindet, werden auch für Blöcke am rechten Rand Gewichte berechnet, obwohl es im Voraus bekannt ist, dass die Blending-Funktion für weit liegende Pixel Werte nahe Null liefert. Also, wenn nur die Blöcke betrachtet werden, die zum blendenden Pixel nah genug sind, kann die Berechnungszeit vermindert werden.

2.2.5 Pixelbasierte Methoden

Pixelbasierte Bildfusion-Verfahren berechnen Pixelwerte des Ausgabebildes basierend auf lokalen Charakteristiken entsprechender Pixel in Ursprungsaufnahmen. Verschiedene Charakteristiken können sowohl direkt aus den Werten der Ursprungspixel, als auch aus lokalen Nachbarschaften der betrachteten Pixel berechnet werden (Abbildung 2.15).

Im Weiteren werden einige Pixelbasierte Methoden besprochen und entsprechende Formel für die Berechnung der Pixelwerte für das Ausgabebild gezeigt.

Mittelwert

Eine sehr einfache und naheliegende Idee ist die Mittelwertberechnung:

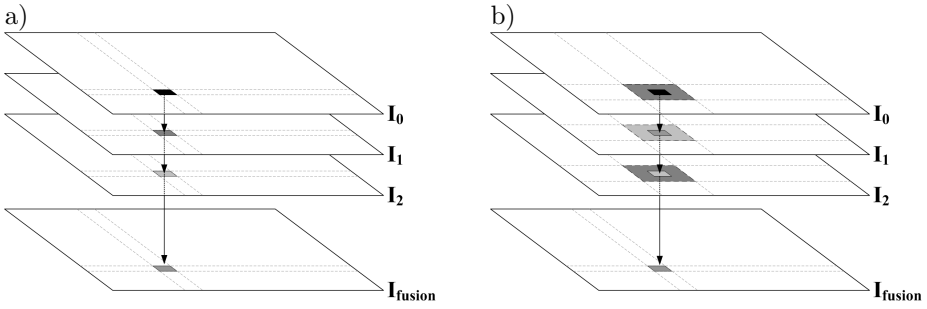


Abbildung 2.15: *Pixelbasierte Bildfusion, basierend nur auf Pixelwerten a) oder auf Charakteristiken der Nachbarschaft b)*

$$\mathbf{O}^{mean}(x,y) = \frac{1}{N}\sum_{i=0}^{N-1}\mathbf{I}_i(x,y) \tag{2.39}$$

N ist dabei die Anzahl der Aufnahmen.

Median

Es wird ein Median der Werten aller an der gleichen Position stehenden Pixel bestimmt, d.h. ein Zentralwert der N Pixelwerte $\mathbf{I}_i(x,y)$ in zwei Hälften teilt.

$$\mathbf{O}^{median}(x,y) = median\left(\mathbf{I}_i(x,y)\right),\ 0 \leq i < N \tag{2.40}$$

Um den Medianwert zu bestimmen, werden N Pixelwerte aufsteigend sortiert, und dann nach folgender Formel:

$$median(x_i) = \begin{cases} x_{\frac{N-1}{2}} & falls\, N\, ungerade \\ \frac{1}{2}\left(x_{\frac{N}{2}} + x_{\frac{N}{2}-1}\right) & falls\, N\, gerade \end{cases},\ 0 \leq i < N \tag{2.41}$$

wird der Median bestimmt. $x_i,\ 0 \leq i < N$ sind dabei Pixelwerte in sortierter Reihenfolge.

Gewichtete Summe

Die Idee dieser Methode besteht darin, bei der Summenbildung jede Pixelintensität zu gewichten:

$$\mathbf{O}^{weights}(x,y) = \sum_{i=0}^{N-1} w_i(x,y)\mathbf{I}_i(x,y) \tag{2.42}$$

Die Gewichte $w_i(x,y)$ können nach diversen Kriterien berechnet werden,

dabei können sowohl nur Intensitäten der entsprechenden Pixel, als auch die Pixelnachbarschaften betrachtet werden. Methoden, die mit Nachbarschaften arbeiten, liefern bessere Ergebnisse und werden in weiteren Abschnitten besprochen.

Allgemeine Beobachtung für die drei Methoden - Median, Mittelwert und Gewichtete Summe (hier ohne Berücksichtigung Nachbarschaften) ist, dass die Formel schnell zu berechnen sind, liefern aber keine gute Ergebnisse. Weil es dabei nicht berücksichtigt wird, ob ein Pixel aus einer unter- oder überbelichteten Bildregion kommt und ob seine Verwendung in der Berechnung des neuen Pixelwertes sinnvoll ist.

In weiteren Abschnitten werden Methoden beschrieben, die nicht nur den Intensitätswert des Pixels selbst, sondern auch die Intensitäten der Pixel in der Nachbarschaft betrachten.

Pixel Entropie

Bei dieser Methode wird für jedes Pixel die Entropie seiner Nachbarschaft berechnet. Die Entropie wird hier verwendet, um den Informationsinhalt der Pixelnachbarschaft zu bestimmen. Die Idee der Pixel-Entropie Methode besteht darin, in Ausgangsbildern Pixel mit informationsreichster Nachbarschaft zu bestimmen und nur solche Pixel in das Ausgabebild rein zu nehmen. Die Größe der Nachbarschaft ist dabei ein wichtiger Optimierungsparameter, denn große Nachbarschaften liefern gute Ergebnisse, erhöhen aber den Rechenaufwand, kleine Nachbarschaften ergeben Bilder mit sichtbaren Kanten zwischen unterschiedlich belichteten Bildregionen.

Für jedes Pixel wird seine Nachbarschaft als ein Teilbild betrachtet. Mit $\mathbf{I}^{(i)}_{d\times d}(x,y)$ wird die $d \times d$ Nachbarschaft des Pixels (x,y) im Ausgangsbild $\mathbf{I}^{(i)}$ bezeichnet. Dann wird es für jedes Pixel entschieden, aus welchem der N Ausgangsbilder es seinen Wert bekommt. Dafür wird für jedes der N aus Pixelnachbarschaften erzeugten Teilbilder

$$\mathbf{I}^{(i)}_{d\times d}(x,y),\ 0 \leq i < N$$

die Entropie berechnet. Es wurde schon im Abschnitt 2.2.2 besprochen, wie Entropie als Qualitätsmaß für ein Bild oder eine Bildregion berechnet werden kann. Der Entropiewert wird mit E bezeichnet. Dann wird mit

$$E\left(\mathbf{I}^{(i)}_{d\times d}(x,y)\right)$$

die Entropie des entsprechenden Teilbildes $\mathbf{I}^{(i)}_{d\times d}(x,y)$ bezeichnet. Somit nach folgender Formel:

$$k = \arg \max_{0 \leq i < N} \left\{ E\left(\mathbf{I}^{(i)}_{d \times d}(x, y)\right) \right\} \tag{2.43}$$

kann das Bild $\mathbf{I}_k$ bestimmt werden, in dem sich das Pixel mit informationsreichster Nachbarschaft befindet. Der Pixelwert für das Ausgabebild ist dann:

$$\mathbf{O}^{entropie}(x, y) = \mathbf{I}_k(x, y) \tag{2.44}$$

Wie bereits erwähnt, bei dieser Methode weisen oft die Ausgangsbilder sichtbare Kanten zwischen unterschiedlich belichteten Bildregionen. Solche Kanten können dadurch geglättet werden, dass anstatt des Intensitätswertes eines ausgewählten Pixels, eine gewichtete Summe aus den Werten entsprechender Pixel berechnet wird. Als Gewichte werden die Verhältnisse der Entropiewerte zur Summe der Entropien genommen:

$$w_i^{entropie}(x, y) = \frac{E\left(\mathbf{I}^{(i)}_{d \times d}(x, y)\right)}{\sum_{j=0}^{N-1} E\left(\mathbf{I}^{(j)}_{d \times d}(x, y)\right)} \tag{2.45}$$

Somit ist die endgültige Formel für Pixelwerte im Ausgangsbild:

$$\mathbf{O}^{entropie}(x, y) = \sum_{i=0}^{N-1} w_i^{entropie}(x, y) \mathbf{I}_i(x, y) \tag{2.46}$$

Die Pixel-Entropie Methode liefert relativ gute Ergebnisse, wenn die Nachbarschaft groß gewählt wird. Das führt aber zu extrem hohem Rechenaufwand, weil es für jedes Pixel in jedem Bild Entropie seiner Nachbarschaft berechnet werden muss.

Kantenintensitäten

Um lokale Kantenstärken in einem Bild $\mathbf{I}$ zu berechnen, muss das Bild zuerst mit einem Gauß-Kern $\mathcal{G}_{d \times d}$ geglättet werden:

$$\mathbf{I}^G = \mathbf{I} * \mathcal{G}_{d \times d} \tag{2.47}$$

Die Unterschiede der beiden Bilder sind die kontinuierlichen Kantenintensitäten $\mathbf{E}^G$. Um die zu berechnen, wird vom Ursprungsbild das geglättete Bild $\mathbf{I}^G$ abgezogen:

$$\mathbf{E}^G = v\left(\mathbf{I} - \mathbf{I}^G\right) \tag{2.48}$$

v ist hier eine Skalierungsvariable, die Kantenintensitäten verstärken kann. Die Größe d des Gauß-Kerns beeinflusst das Kantenspektrum, kleine Kerne er-

geben dabei ein breiteres Spektrum und größere - schmaleres (Abbildung 2.16). Große Gauß-Kerne können wegen des schmalen Spektrums erfolgreich für Binarisierung von Dokumentenaufnahmen verwendet werden, weil Pixelwerte aus einem schmalen Spektrum leichter in Hinter- und Vordergrund getrennt werden können.

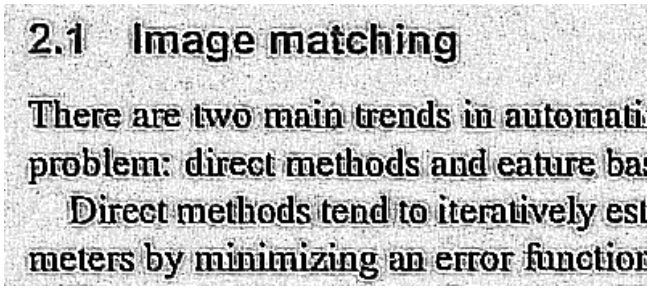
Bei dieser Fusionsmethode wird eine gewichtete Summe von Kantenintensitäten der Eingangsbilder berechnet. Dafür wird für jedes der N Eingangsbilder $\mathbf{I}_i$ sein Kantenintensitätsbild $\mathbf{E}_i^G$ berechnet. Aus den Kantenintensitäten kann dann für jedes Pixel sein Gewicht bestimmt werden:

$$w_i^{intens}(x,y) = \frac{\mathbf{E}_i^G(x,y)}{\sum_{j=0}^{N-1} \mathbf{E}_j^G(x,y)} \tag{2.49}$$

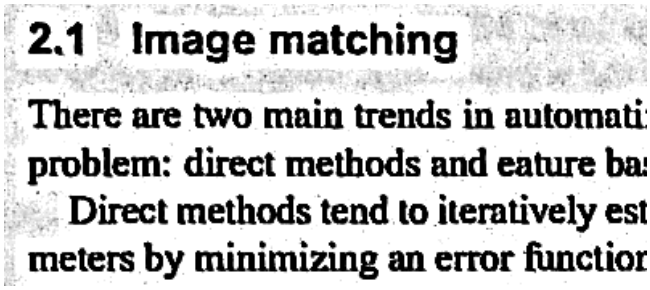
Anschließend wird für jedes Pixel im Ausgabebild sein Wert als gewichtete Summe der Pixelwerte der Ursprungsbilder berechnet:

$$\mathbf{O}^{intens}(x,y) = \sum_{i=0}^{N-1} w_i^{intens}(x,y)\mathbf{I}_i(x,y) \tag{2.50}$$

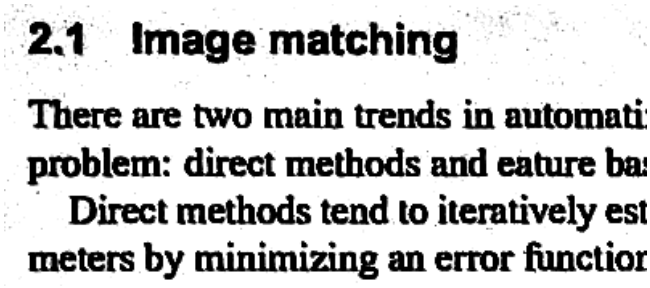
Der Rechenaufwand hängt dabei sehr stark von der Größe des Gauß-Kerns ab, weil es zwar mit schneller Fourier-Transformation effizienter berechnet werden kann, aber für große Kerne wird trotzdem viel Rechenzeit benötigt.



a) Gauß-Kern 3×3



b) Gauß-Kern 21×21



c) Gauß-Kern 121×121

Abbildung 2.16: *Kantenintensitäten einer Dokumentenaufnahme berechnet mit verschiedenen Größen des Gauß-Kerns*

Kapitel 3

Alignmentkorrektur

Bei Erstellung von Hochkontrastbildern müssen einzelne Aufnahmen zuerst in die bestmögliche Übereinstimmung zueinander gebracht werden. Kleine Verschiebungen der Kamera können durch den Menschen oder durch die Hardware ausgelöst werden. Auch trotz Verwendung eines Stativs kann es vorkommen, dass kleine Verschiebungen zwischen einzelnen Aufnahmen entstehen. (Abbildung 3.1). Bevor die Aufnahmen zu einem Bild zusammengefügt werden können, müssen sie in die bestmögliche Übereinstimmung zueinander gebracht werden.

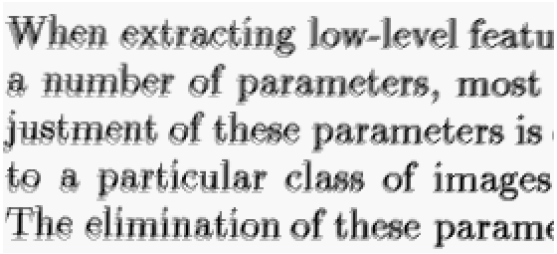


Abbildung 3.1: *Verschiebungen bei Aufnahmen*

In dieser Arbeit wurden drei verschiedene Ansätze betrachtet - *globales* und *lokales* Alignment und die *Thin-Plate-Spline* (TPS) Methode. Mit Alignment-Methoden können nur Verschiebungen korrigieren werden, wobei mit lokalen Alignments können auch kleine Rotationen korrigiert werden. Bei der TPS-Methode werden die Korrekturen mit Hilfe von Interpolation durchgeführt. Dabei wird ein Bild als Referenzbild gewählt, und alle anderen zu diesem Bild *interpoliert*. Dafür werden Translationsvektoren verwendet, die aus lokalen Verschiebungen einzelnen Bildregionen ermittelt werden.

Zum Schluss ist noch anzumerken, dass nach Angaben von Greg Ward in seiner Arbeit über Bildregistrierung ([5]), in 90% aller Aufnahmenserien sind nur kleine Verschiebungen und keine Rotationen vorhanden, somit ist die re-

chenaufwendige TPS-Methode nicht notwendig.

3.1 Globales Alignment

Mit Hilfe des globalen Alignments wird eine Translation eines Bildes bezüglich des Referenzbildes ermittelt, die den kleinsten *Ausrichtungsfehler* liefert. D.h. für ein Bild $\mathbf{I}_t$ und ein Referenzbild $\mathbf{I}_{ref}$ muss ein Translationsvektor $t = (t_x, t_y)$ gefunden werden, so dass die Verschiebung des Bildes $\mathbf{I}_t$ bezüglich $\mathbf{I}_{ref}$ um t_x Punkte auf x-Achse und t_y Punkte auf y-Achse den kleinsten Fehler liefert (Abbildung 3.2).

Formal kann eine Translation eines Bildes $\mathbf{I}_t$ um einen Vektor $t = (t_x, t_y)$ als folgende Operation betrachtet werden:

$$\widetilde{\mathbf{I}}_t(x,y) = \begin{cases} \mathbf{I}_t(x+t_x, y+t_y) & wenn\ (0 \le x+t_x < M) \\ & \quad \wedge (0 \le y+t_y < N) \\ C & wenn\ (x+t_x \ge M) \vee (y+t_y \ge N) \\ & \quad \vee (x+t_x < 0) \vee (y+t_y < 0) \end{cases} \tag{3.1}$$

M und N sind entsprechend Breite und Höhe des Ursprungsbildes $\mathbf{I}_t$, C ist ein konstanter Pixelwert, mit dem leere Bereiche im neuen Bild aufgefüllt werden. Die Größe des neuen Bildes $\widetilde{\mathbf{I}}_t$ wird mit $\tilde{M} \times \tilde{N}$ bezeichnet mit

$$\begin{aligned} \tilde{M} &= M + |t_x| \\ \tilde{N} &= N + |t_y| \end{aligned}$$

d.h., nach der Translation kann das Bild vergrößert werden. Die neuen Pixel, für die es keine Ursprungspixel im Bild $\mathbf{I}_t$ gibt, werden auf den konstanten Wert C gesetzt, dieser Wert gleicht in der Regel der Hintergrundfarbe. Für Dokumentenaufnahmen ist es sinnvoll, den Hintergrund auf Weiß zu setzen, d.h. auf den größten Pixelwert.

Die Berechnung des Ausrichtungsfehlers zwischen zwei Bildern wird auf ihrer Schnittmenge ausgeführt, d.h. auf dem Bereich, der nach der Ausrichtung für beide Bilder gemeinsam ist (Abbildung 3.2).

Im Weiteren wird der Teil eines Bildes $\mathbf{I}_t$, der zur Schnittmenge mit dem Referenzbild gehört, mit $\mathbf{I}'_t$ bezeichnet und kann weiter als eigenständiges Bild betrachtet werden. Das Gleiche gilt auch für den gemeinsamen Teil des Referenzbildes $\mathbf{I}_{ref}$, der entsprechend mit $\mathbf{I}'_{ref}$ bezeichnet wird.

Angenommen, die Größe von $\mathbf{I}_{ref}$ ist $M_{ref} \times N_{ref}$, von $\mathbf{I}_t$ - $M_t \times N_t$, dann kann die Größe der Schnittmenge mit $M' \times N'$ bezeichnet und folgendermaßen berechnet werden:

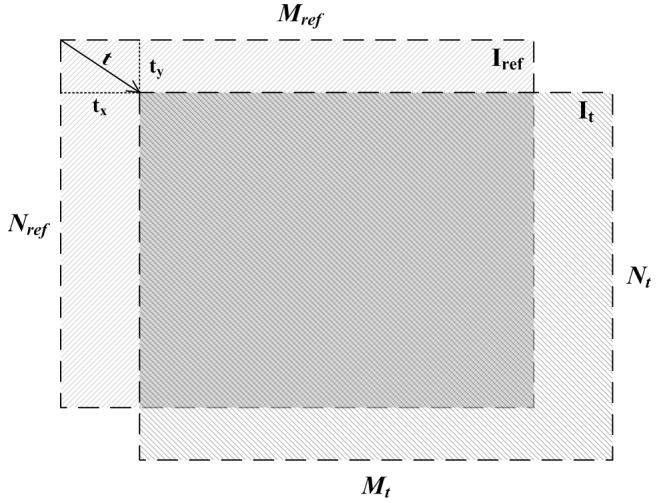


Abbildung 3.2: *Schnittmenge zwei Bilder nach der Translation*

$$M' = \min(M_{ref},\, M_t,\, M_{ref} - t_x) \tag{3.2}$$

$$N' = \min(N_{ref},\, N_t,\, N_{ref} - t_y) \tag{3.3}$$

Für die Berechnung des Ausrichtungsfehlers zwischen $\mathbf{I}'_t$ und $\mathbf{I}'_{ref}$ können folgende Funktionen verwendet werden:

1. quadratischer Abstand

$$E_{SSD}(\mathbf{I}'_{ref}, \mathbf{I}'_t) = \sum_{x=0}^{M'-1} \sum_{y=0}^{N'-1} \left[\mathbf{I}'_{ref}(x,y) - \mathbf{I}'_t(x,y)\right]^2 \tag{3.4}$$

2. absoluter Abstand

$$E_{SAD}(\mathbf{I}'_{ref}, \mathbf{I}'_t) = \sum_{x=0}^{M'-1} \sum_{y=0}^{N'-1} \left|\mathbf{I}'_{ref}(x,y) - \mathbf{I}'_t(x,y)\right| \tag{3.5}$$

3. Anzahl der abweichenden Pixel in binarisierten Bildern

$$E_{XOR}(\mathbf{I}'_{ref}, \mathbf{I}'_t) = \sum_{x=0}^{M'-1} \sum_{y=0}^{N'-1} \mathcal{B}\left[\mathbf{I}'_{ref}(x,y)\right] \otimes \mathcal{B}\left[\mathbf{I}'_t(x,y)\right] \tag{3.6}$$

$\mathcal{B}$ bezeichnet hier eine Binarisierungsoperation auf einem Bild und $\otimes$ - eine binäre XOR-Operation. Es wird somit die Anzahl der Pixel in zwei Binärbildern gezählt, die auf der gleichen Position stehen, aber unter-

schiedliche Werte haben.

Mit einer gewählten Fehlerfunktion kann die Suche nach dem besten Alignment begonnen werden. Es existieren zwei Arten von Verfahren für Alignmentsuche - *merkmalsbasierte* und *direkte* Methoden.

Bei merkmalsbasierten Methoden werden zuerst distinktive Merkmale in jedem Bild ermittelt und dann die entsprechenden Alignments für Positionen von Merkmalen berechnet.

Direkte Methoden untersuchen die Übereinstimmung zwischen den Pixeln zweier Bilder. Dafür wird ein Bild über das andere mit einem bestimmten Alignment gelegt und die Fehlerfunktion berechnet. Um den kleinsten Fehler zu finden, müssen alle möglichen Alignments getestet werden, das wäre die einfachste Vorgehensweise. In der Praxis wird solche Methode aber kaum benutzt wegen des hohen Rechenaufwands. Um die Suche zu beschleunigen, wird eine grob-zu-fein Technik mit Ansatz von Bildpyramiden verwendet.

3.1.1 Alignmentsuche mit Bildpyramiden

Es wird für zwei untersuchte Bilder je eine Bildpyramide erstellt, die Bildgrößen werden dabei jeweils halbiert bis eine vorgegebene Größe erreicht wird (Abbildung 3.3). Dabei ist es wichtig zu bemerken, dass die Bildpyramiden die gleiche Anzahl von Ebenen haben müssen, d.h. wenn bei der Erstellung einer Pyramide die vorgegebene Bildgröße erreicht wurde, werden auch in die andere Pyramide keine Bilder mehr hinzugefügt.

Wenn die Größe des kleinsten Bildes in der Pyramide als $m \times n$ vorgegeben wurde, dann ist für ein Bild der Größe $M \times N$ die Anzahl der Pyramidenebenen (Bilder):

$$S = 1 + \min\{0; \log_2 \frac{M}{m}; \log_2 \frac{N}{n}\}$$

Somit enthält eine Pyramide mindestens eine Ebene - die mit dem Originalbild. Im Weiteren wird mit $\mathbf{I}^{(s)}\,(0 \leq s < S)$ das Bild in der Pyramidenebene s bezeichnet, $\mathbf{I}^{(0)}$ ist dabei gleich dem Originalbild $\mathbf{I}$.

Die Suche fängt mit den kleinsten Bildern, auf der gröbsten Ebene, an. Hier wird eine komplette Suche durchgeführt. Dabei ist es sinnvoll, eine obere Grenze für die Größe der untersuchten Verschiebungen zu setzen, weil es in manchen Fällen vorkommen kann, dass der kleinste Fehler von einem zu weiten und deswegen falschem Alignment geliefert wird (Abbildung 3.4). Außerdem, wenn es im Voraus bekannt ist, dass die Verschiebung sich in bestimmten Grenzen hält, können dadurch unnötige Berechnungen erspart werden.

Nachdem für die kleinsten Bilder $\mathbf{I}^{(S-1)}$ und $\mathbf{J}^{(S-1)}$ ein Translationsvektor

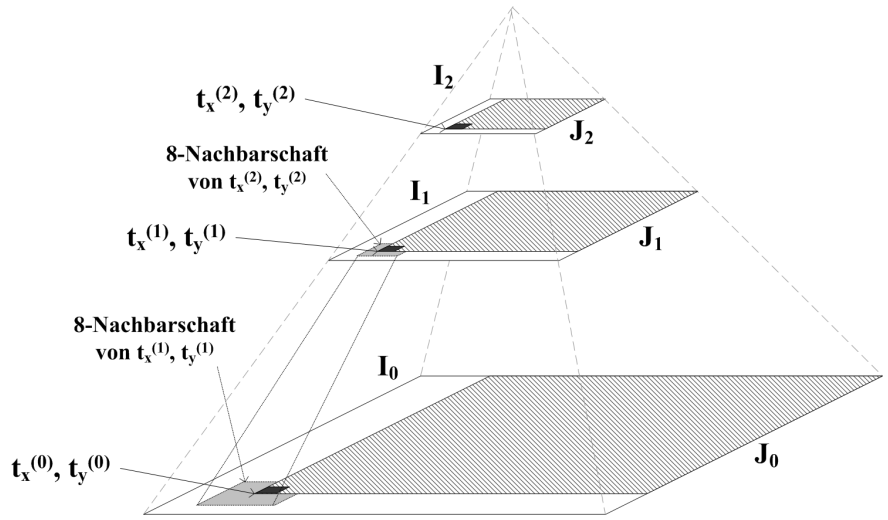


Abbildung 3.3: *Suche von Pixelkorrespondenzen der zwei Bilder* $\mathbf{I}$ *und* $\mathbf{J}$ *mit Hilfe von Bildpyramiden. Auf der nächst feineren Ebene wird nur die 8-Nachbarschaft durchgesucht.*

$t^{(S-1)} = (t_x^{(S-1)}, t_y^{(S-1)})$ ermittelt wurde, kann für die nächst feinere Ebene $S-2$ eine Translation $t^{(S-2)}$ berechnet werden. Weil die Bildgrößen halbiert wurden, sind die möglichen Werte für eine Translation $t^{(s-1)} = (t_x^{(s-1)}, t_y^{(s-1)})$ auf der Ebene s-1:

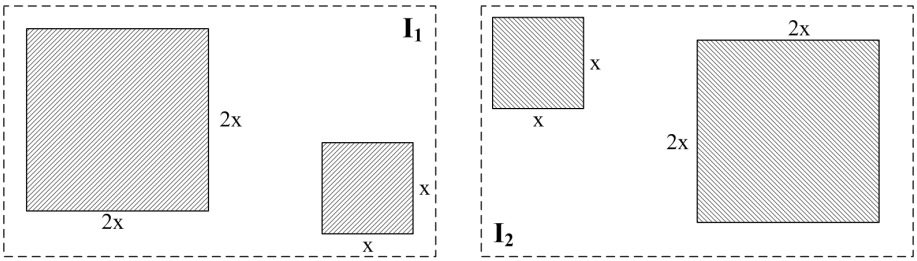
$$\begin{aligned} t_x^{(s-1)} &\in [2t_x^{(s)} - 1,\ 2t_x^{(s)} + 1] \\ t_y^{(s-1)} &\in [2t_y^{(s)} - 1,\ 2t_y^{(s)} + 1] \end{aligned}$$

Es müssen also für die nächst feinere Ebene nur 9 Alignments getestet werden (Abbildung 3.5).

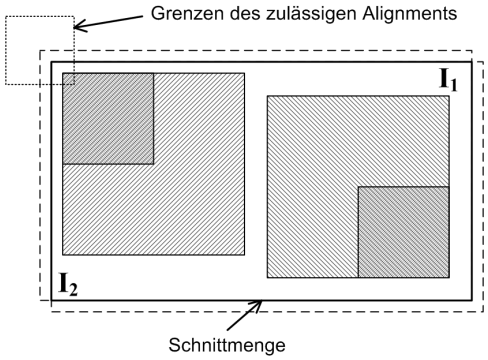
Dies wird bis zur feinsten Pyramidenebene, den Ursprungsbildern $\mathbf{I}^{(0)}, \mathbf{J}^{(0)}$, durchgeführt, und dadurch wird der Translationsvektor $t^{(0)} = (t_x^{(0)}, t_y^{(0)})$ ermittelt (Abbildung 3.3). Dieser Vektor ist die Verschiebung eines Bildes bezüglich des anderen - das gesuchte globale Alignment.

3.1.2 Begrenzung des Suchraums

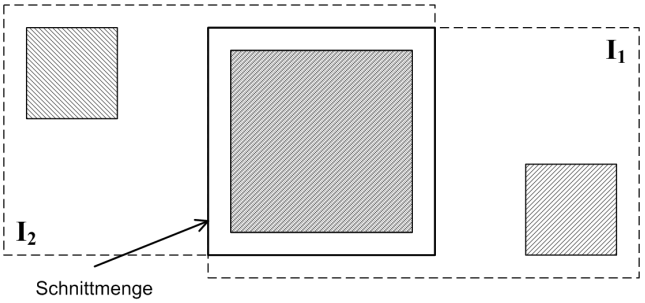
Die einfachste Methode für die Suche nach dem besten Alignment ist ein systematisches Ausprobieren von allen möglichen Translationen. Für jede Translation wird dabei ein Ausrichtungsfehler berechnet und anschließend das Alignment mit dem kleinsten Fehler als bestes gewählt. So eine einfache Vorgehensweise führt aber in meisten Fällen zu unnötigen Berechnungen, weil bei den Dokumentenaufnahmen die Größe der Verschiebungen in einem gewissen begrenzten



a) Zwei Bilder mit angegebenen Regionen



b) Das beste Alignment unter Berücksichtigung der festgelegten Grenze



c) Das Alignment ohne festgelegten Grenze. Falsches Ergebnis

Abbildung 3.4: *Beispiel für falsches Alignment*

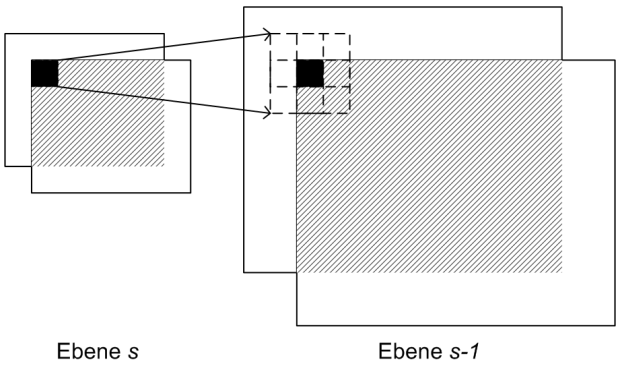


Abbildung 3.5: *Mögliche Alignments auf der nächst feineren Pyramidenebene*

Bereich liegt. Die Experimente mit Stativaufnahmen haben gezeigt, dass die Translationen nie größer als 2% der Bildgröße sind, d.h. bei einem Bild der Größe $M \times N$ kann für Translationen eine Grenze von $0.02 \cdot M$ für horizontale und $0.02 \cdot N$ für vertikale Verschiebung gesetzt werden. Verallgemeinert lässt sich diese Bedingung mit folgender Formel ausdrucken:

$$t = (t_x, t_y), \quad |t_x| < b_x, \ |t_y| < b_y \tag{3.7}$$

t ist dabei ein Translationsvektor, und b_x, b_y sind obere Grenzen für mögliche Verschiebungen in x- und bzw. y-Richtung. Die oberen Grenzen können, wie bereits erwähnt, abhängig von der Bildgröße gesetzt werden oder auf einen bestimmten statischen Wert gesetzt werden, wenn die maximale mögliche Verschiebung im Voraus bekannt ist.

3.1.3 Optimierung der Suche

Noch eine Möglichkeit die Suche zu beschleunigen besteht darin, dass der kleinste Ausrichtungsfehler und entsprechender Translationsvektor gespeichert und ständig aktualisiert werden. Am Anfang wird die Translation auf $(0, 0)$ und der Fehler auf den größtmöglichen Wert gesetzt, dann bei jeder Berechnung des Ausrichtungsfehlers wird der summierte Wert mit dem gespeicherten momentan kleinsten Fehler verglichen. Wenn die Summe größer als der gespeicherte kleinste Fehler wird, dann heißt es, dass es schon eine bessere Translation mit einem kleineren Ausrichtungsfehler gab. Deswegen ist es sinnlos, weitere Berechnungen für diesen Translationsvektor durchzuführen, und es kann die nächste Translation getestet werden. Wenn es ein Alignment mit einem kleineren als momentan gespeicherte Fehler gefunden wird, dann heißt es, dass es ein besseres Alignment ermittelt wurde und die gespeicherten Translationsvektor und Ausrichtungsfehler aktualisiert werden müssen.

Für eine schnellere Suche nach Alignments kann eine *Multiresolution-Technik* verwendet werden. Dafür wird eine Bildpyramide erstellt, wie im Abschnitt 2.1.6 beschrieben. Die Suche fängt mit der kleinsten Pyramidenstufe an, dann für jede weitere (jeweils größere) Ebene wird das Ergebnis des Vorgängers korrigiert bis ein Alignment für Bilder in Originalgröße gefunden wird.

Wie bereits erwähnt, für die Suche nach Pixelkorrespondenzen können diverse Kriterien verwendet werden (Formeln 3.4-3.6). Für eine schnellere Berechnung des Ausrichtungsfehlers kann die Formel 3.6 (Anzahl der abweichender Pixel in Binärbildern) in Kombination mit einem Binarisierungsverfahren verwendet werden. Der Vorteil dabei besteht darin, dass bitweise XOR-Operation schneller als Subtractions- und Multiplikationsoperationen ist.

3.2 Lokales Rekursives Verfahren

Auf den meisten Dokumentenaufnahmen ist die Textdichte ungleichmäßig verteilt. Für Bildregionen mit hoher Dichte muss eine genauere Pixelkorrespondenz ermittelt werden, als für andere, dünn besetzte, Regionen. Dies wird dadurch erreicht, dass für Bildregionen mit hoher Textdichte die Alignmentsuche rekursiv durchgeführt wird. Für die Alignmentsuche wird hier das im Abschnitt 3.1 beschriebene Verfahren mit Ansatz von Bildpyramiden verwendet.

Für zwei Bilder $\mathbf{I}$ und $\mathbf{J}$ wird zunächst eine grobe Ausrichtung mit Hilfe eines globalen Alignments gefunden. Dann wird jedes Bild in vier gleichgroße Teile zerlegt und für jeden Teil das Verfahren rekursiv wiederholt. Solche Teilbilder werden im Weiteren mit $\mathbf{I}_{(t)}$ bezeichnet. Dabei bei jeder Zerlegung für die vier neu entstehenden Bildteile muss das schon gefundene Alignment des teilenden "Vater"-Bereichs berücksichtigt werden. Als Abbruchskriterium der Rekursion kann eine der folgenden Bedingungen ausgewählt werden:

- eine vorgegebene *minimale Bildgröße* wurde erreicht. Das hat den Nachteil, dass alle Bildregionen gleich behandelt werden, sogar die mit niedriger Textdichte oder gar ohne Text. Und wenn ein Bildbereich schon perfekt ausgerichtet wurde, wird es trotzdem in Teile zerlegt und weiter rekursiv bearbeitet. Dies erhöht den Rechenaufwand durch unnötige Berechnungen.

- der *Ausrichtungsfehler ist klein genug.* Die Rekursion wird in diesem Fall so lange durchgeführt, bis der Ausrichtungsfehler kleiner als ein vorgegebener Schwellwert wird. Es kann aber vorkommen, dass es zwei Bildregionen gegeneinander ausgerichtet werden müssen, für die es keine ausreichend gute Ausrichtung gibt. In diesem Fall wird der Fehler nur dann kleiner als der Schwellwert, wenn die Bildregionen klein genug sind und trotz

fehlender Ausrichtung, nur dank ihrer kleinen Größe, einen kleinen Fehler liefern.

- eine *Kombination* von minimaler Bildgröße und minimalem Ausrichtungsfehler. Die Rekursion wird dabei abgebrochen, wenn eine minimale Bildgröße erreicht oder/und wenn der Ausrichtungsfehler klein genug wird. Begrenzung durch minimale Fehlergröße hat den Vorteil, dass auch beim ersten Rekursionsschritt große Teile des Bildes aus der weiteren rekursiven Verarbeitung ausgenommen werden können, wenn für sie eine ausreichend gute Ausrichtung mit einem niedrigen Fehler gefunden wird. Das beschleunigt das Verfahren erheblich. Andererseits, wenn es für zwei Bildregionen keine gute Ausrichtung gibt, wird die Rekursion nur bis zur bestimmten Bildgröße durchgeführt, ungeachtet des Ausrichtungsfehlers.

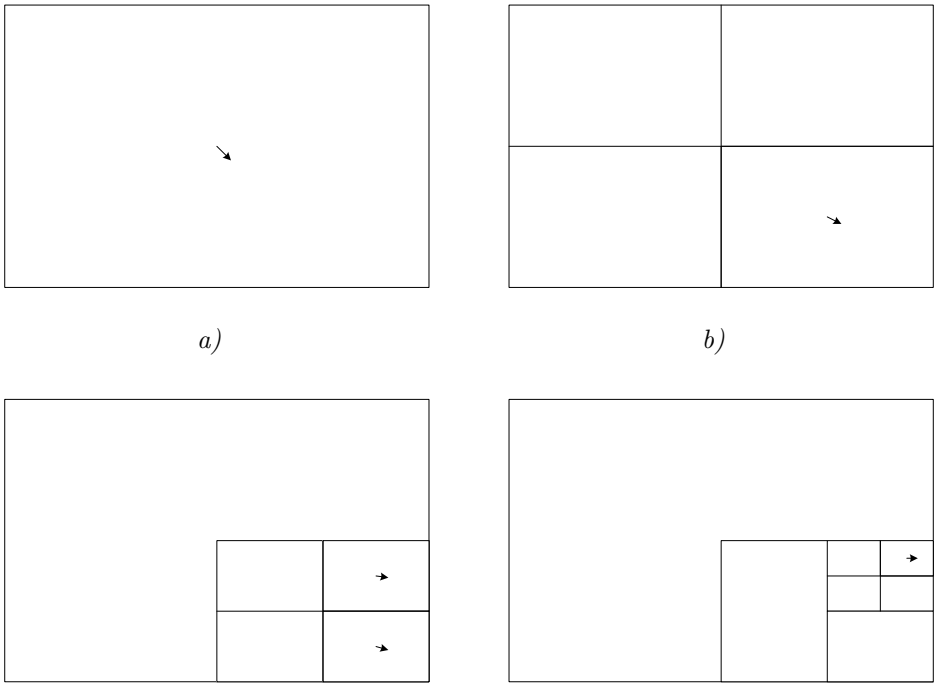
Es ist wichtig hier zu bemerken, dass auf diese Weise das Ursprungsbild in *disjunkte* Teile zerlegt wird, d.h. keine Teile überlappen sich und aus diesen Teilen lässt sich das Ursprungsbild zusammenstellen.

Dieses Verfahren kann mit einer Prioritätswarteschlange realisiert werden. Als Schlüssel, der die Reihenfolge der Abarbeitung bestimmt, wird der Ausrichtungsfehler (Anzahl von Missmatches fürs Alignment) gewählt. Bildteile werden in die Prioritätswarteschlange eingefügt, und nacheinander für jedes eingefügte Bildteil wird sein Alignment bestimmt. Bei der Initialisierung wird das ganze Bild eingefügt und dann gleich im ersten Schritt in vier Teile zerlegt.

Zusätzlich zu den schon erwähnten Abbruchskriterien wie *minimale Größe des Teilbildes* und *Größe des Ausrichtungsfehlers*, kann auch die *maximale Anzahl von Teilbilder* verwendet werden. Damit lässt sich die Anzahl der Alignments begrenzen, was in einigen Fällen sinnvoll sein kann.

Nachdem die rekursive Ausrichtung beendet ist, entsteht eine Menge von Bildteilen mit entsprechenden Translationsvektoren (Abbildung 3.6). Jetzt kann ein Ausgabebild erzeugt werden, dafür wird jeder Teil entsprechend seiner Translation in das Ausgabebild eingefügt. Es kann dabei vorkommen, dass nach der Verschiebung von Bildregionen leere, unbesetzte Bereiche im Ausgabebild entstehen (Abbildung 3.7). Die können zum Beispiel mit der Hintergrundfarbe gefüllt werden.

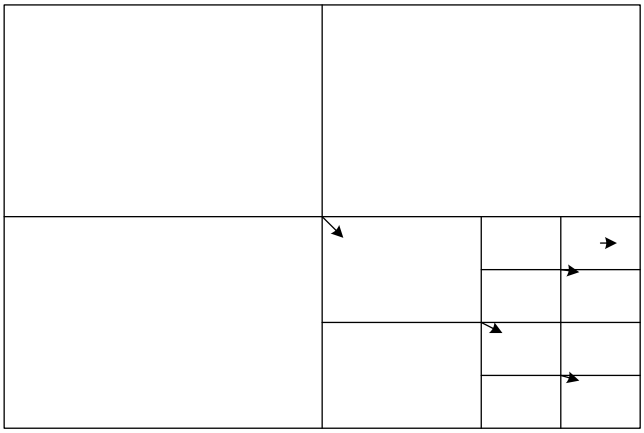
Es existiert außerdem noch eine Möglichkeit, zwei Bilder mit Hilfe von Translationsvektoren gegeneinander auszurichten. Das kann durch Interpolation mit der Thin-Plate-Spline Methode gemacht werden.



a) *b)*

c) *d)*

a) - d) rekursive lokale Alignments



e) ermittelte Translationen

Abbildung 3.6: *Beispiel für lokales rekursives Alignment*

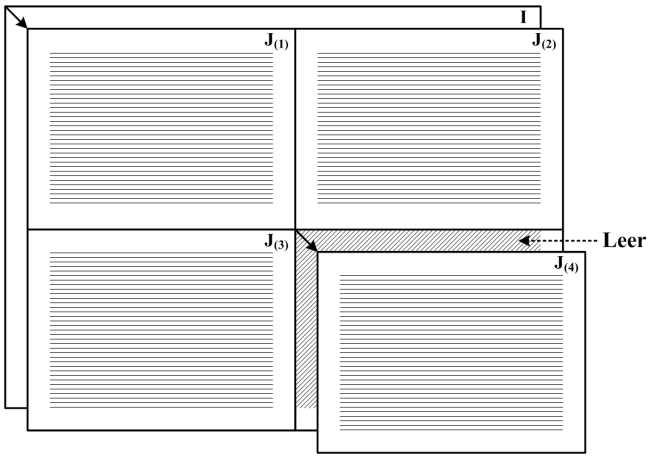


Abbildung 3.7: *Nach Verschiebung von Bildregionen können leere Bereiche im Ausgabebild entstehen. Hier durch Verschiebung des Teilbildes* $\mathbf{J}_{(4)}$

3.3 Thin-Plate-Spline

Thin Plate Spline ist eine Interpolationsmethode, die für eine gegebene Menge von Punkten eine minimal verkrümmte Fläche findet, die durch alle Punkte verläuft [6].

Für Interpolation im zweidimensionalen Raum werden zwei gleich große Mengen von Punktkoordinaten benötigt:

$$\begin{aligned} A &= (a_{x_i}, a_{y_i}) \\ B &= (b_{x_i}, b_{y_i}), \quad 0 \leq i < N \end{aligned}$$

Angenommen, A wird zu B interpoliert. Dafür wird eine Menge von Translationsvektoren T generiert mit:

$$T = (t_{x_i}, t_{y_i}) = (b_{x_i} - a_{x_i}, b_{y_i} - a_{y_i}) \tag{3.8}$$

Dann werden zwei TPS-Funktionen berechnet, die unter Berücksichtigung aller Translationsvektoren eine Interpolation für einen Punkt $p = (p_x, p_y)$ finden. Als Parameter bekommt eine Funktion eine Menge von Stützpunkten A, eine Menge von entsprechenden Translationen T_x für x- oder T_y für y-Richtung und Koordinaten eines Punktes, der interpoliert wird. Funktionen werden wie folgt bezeichnet, für x-Richtung:

$$tps(A, T_x, p) \tag{3.9}$$

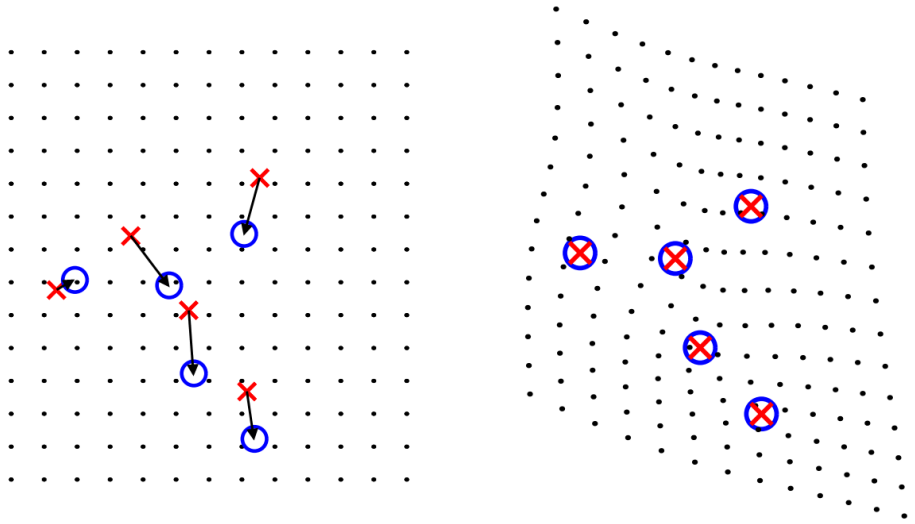


Abbildung 3.8: *Beispiel einer TPS-Transformation. (Aus der Arbeit von Donato, Belongie [6])*

für y-Richtung:

$$tps(A, T_y, p) \tag{3.10}$$

Die Funktionen liefern dabei eine Verschiebung $v = (v_x, v_y)$ mit:

$$v_x = tps(A, T_x, p) \tag{3.11}$$

$$v_y = tps(A, T_y, p) \tag{3.12}$$

die, angewendet auf einen Punkt p, neue, interpolierte, Koordinaten $p' = (p'_x, p'_y)$ liefert:

$$p'_x = p_x + v_x \tag{3.13}$$

$$p'_y = p_y + v_y \tag{3.14}$$

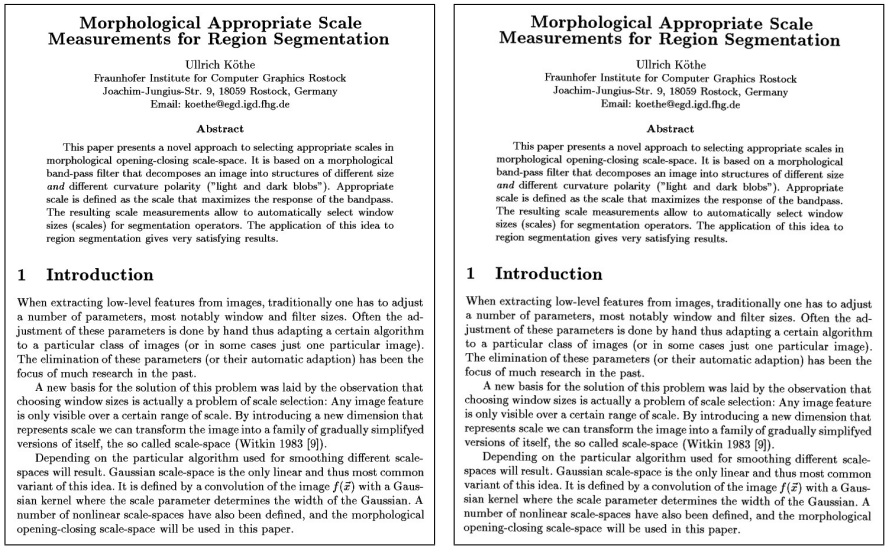
Auf diese Weise kann auf Basis einer Menge von Punktkoordinaten und entsprechenden Translationsvektoren je eine TPS-Funktion für x- und y-Richtung berechnet werden. Mit diesen zwei Funktionen kann dann für einen beliebigen Punkt seine neue Position interpoliert werden (Abbildung 3.8).

Für Alignmentkorrektur kann die TPS-Methode folgendermaßen angewendet werden: mit Hilfe von der im Abschnitt 3.2 beschriebenen Methode werden

lokale Alignments und entsprechende Bildregionen bestimmt. Als Menge A von Stützpunkten werden Mittelpunkte der Bildregionen genommen. Die lokalen Alignments sind nichts anderes als eine Translation aller Punkte eines Bildteils zur gewünschten Position im Ausgabebild, sie bilden somit eine Menge von Translationsvektoren T, die für eine TPS-Funktion benötigt wird. Sie können, also, ohne Änderungen direkt übernommen werden. Somit sind alle für TPS-Funktionen benötigten Parameter vorhanden, und es können für jedes Pixel $p = (p_x, p_y)$ mit Formeln 3.11-3.14 neue Koordinaten berechnet werden.

3.4 Experimente und Ergebnisse

In diesem Abschnitt werden Experimente mit den oben beschriebenen Methoden der Alignmentkorrektur betrachtet und Ergebnisse vorgestellt. Als Begleitbeispiel werden zwei Dokumentenaufnahmen verwendet, die nicht perfekt gegen einander ausgerichtet sind (Abbildung 3.9). Eine Aufnahme wurde bezüglich der anderen um etwa 1 Grad bezüglich des Bildzentrums rotiert.



Morphological Appropriate Scale Measurements for Region Segmentation

Ullrich Köthe
Fraunhofer Institute for Computer Graphics Rostock
Joachim-Jungius-Str. 9, 18059 Rostock, Germany
Email: koethe@egd.igd.fhg.de

Abstract

This paper presents a novel approach to selecting appropriate scales in morphological opening-closing scale-space. It is based on a morphological band-pass filter that decomposes an image into structures of different size *and* different curvature polarity ("light and dark blobs"). Appropriate scale is defined as the scale that maximizes the response of the bandpass. The resulting scale measurements allow to automatically select window sizes (scales) for segmentation operators. The application of this idea to region segmentation gives very satisfying results.

1 Introduction

When extracting low-level features from images, traditionally one has to adjust a number of parameters, most notably window and filter sizes. Often the adjustment of these parameters is done by hand thus adapting a certain algorithm to a particular class of images (or in some cases just one particular image). The elimination of these parameters (or their automatic adaption) has been the focus of much research in the past.

A new basis for the solution of this problem was laid by the observation that choosing window sizes is actually a problem of scale selection: Any image feature is only visible over a certain range of scale. By introducing a new dimension that represents scale we can transform the image into a family of gradually simplifyed versions of itself, the so called scale-space (Witkin 1983 [9]).

Depending on the particular algorithm used for smoothing different scale-spaces will result. Gaussian scale-space is the only linear and thus most common variant of this idea. It is defined by a convolution of the image $f(\vec{x})$ with a Gaussian kernel where the scale parameter determines the width of the Gaussian. A number of nonlinear scale-spaces have also been defined, and the morphological opening-closing scale-space will be used in this paper.

Morphological Appropriate Scale Measurements for Region Segmentation

Ullrich Köthe
Fraunhofer Institute for Computer Graphics Rostock
Joachim-Jungius-Str. 9, 18059 Rostock, Germany
Email: koethe@egd.igd.fhg.de

Abstract

This paper presents a novel approach to selecting appropriate scales in morphological opening-closing scale-space. It is based on a morphological band-pass filter that decomposes an image into structures of different size *and* different curvature polarity ("light and dark blobs"). Appropriate scale is defined as the scale that maximizes the response of the bandpass. The resulting scale measurements allow to automatically select window sizes (scales) for segmentation operators. The application of this idea to region segmentation gives very satisfying results.

1 Introduction

When extracting low-level features from images, traditionally one has to adjust a number of parameters, most notably window and filter sizes. Often the adjustment of these parameters is done by hand thus adapting a certain algorithm to a particular class of images (or in some cases just one particular image). The elimination of these parameters (or their automatic adaption) has been the focus of much research in the past.

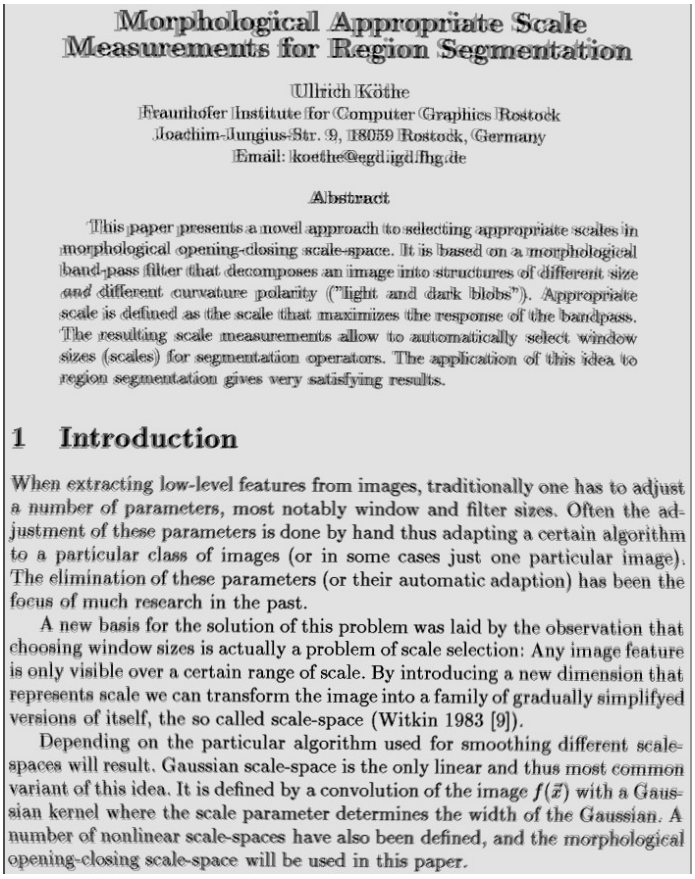
A new basis for the solution of this problem was laid by the observation that choosing window sizes is actually a problem of scale selection: Any image feature is only visible over a certain range of scale. By introducing a new dimension that represents scale we can transform the image into a family of gradually simplifyed versions of itself, the so called scale-space (Witkin 1983 [9]).

Depending on the particular algorithm used for smoothing different scale-spaces will result. Gaussian scale-space is the only linear and thus most common variant of this idea. It is defined by a convolution of the image $f(\vec{x})$ with a Gaussian kernel where the scale parameter determines the width of the Gaussian. A number of nonlinear scale-spaces have also been defined, and the morphological opening-closing scale-space will be used in this paper.

Abbildung 3.9: *Zwei nicht ausgerichtete Aufnahmen, die rechte ist um etwa 1 Grad bezüglich des Zentrums rotiert.*

In diesem Fall kann globales Alignment kein gutes Ergebnis liefern, das Resultatbild ist auf der Abbildung 3.10 zu sehen. Mit dem gefundenen globalen Alignment wurden die Aufnahmen nur an einigen Stellen im unteren Bereich gut ausgerichtet, dabei an den Rändern und besonders im oberen Teil ist die Verschiebung immer noch stark merkbar.

Die Verwendung des lokalen rekursiven Verfahren liefert schon bessere Ergebnisse. Auf der Abbildung 3.11 sind die ermittelten lokalen Alignments und



Morphological Appropriate Scale Measurements for Region Segmentation

Ullrich Köthe
Fraunhofer Institute for Computer Graphics Rostock
Joachim-Jungius-Str. 9, 18059 Rostock, Germany
Email: koethe@egd.igd.fhg.de

Abstract

This paper presents a novel approach to selecting appropriate scales in morphological opening-closing scale-space. It is based on a morphological band-pass filter that decomposes an image into structures of different size *and* different curvature polarity ("light and dark blobs"). Appropriate scale is defined as the scale that maximizes the response of the bandpass. The resulting scale measurements allow to automatically select window sizes (scales) for segmentation operators. The application of this idea to region segmentation gives very satisfying results.

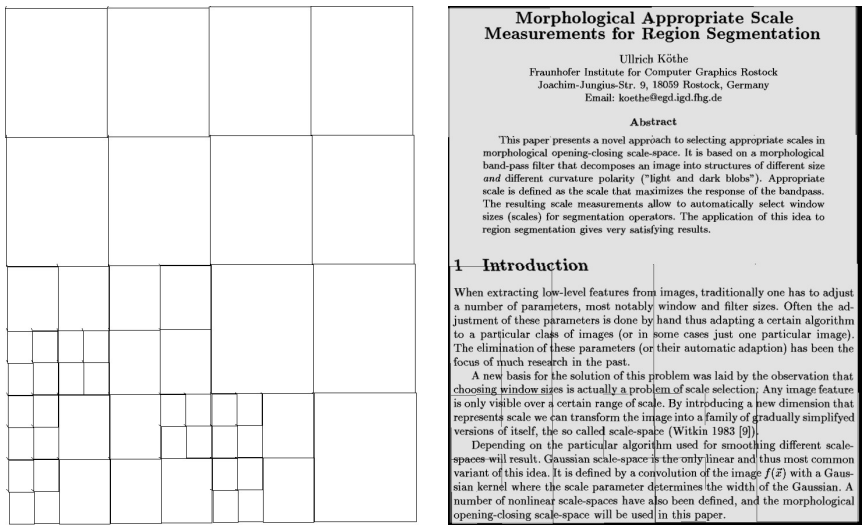
1 Introduction

When extracting low-level features from images, traditionally one has to adjust a number of parameters, most notably window and filter sizes. Often the adjustment of these parameters is done by hand thus adapting a certain algorithm to a particular class of images (or in some cases just one particular image). The elimination of these parameters (or their automatic adaption) has been the focus of much research in the past.

A new basis for the solution of this problem was laid by the observation that choosing window sizes is actually a problem of scale selection: Any image feature is only visible over a certain range of scale. By introducing a new dimension that represents scale we can transform the image into a family of gradually simplifyed versions of itself, the so called scale-space (Witkin 1983 [9]).

Depending on the particular algorithm used for smoothing different scale-spaces will result. Gaussian scale-space is the only linear and thus most common variant of this idea. It is defined by a convolution of the image $f(\vec{x})$ with a Gaussian kernel where the scale parameter determines the width of the Gaussian. A number of nonlinear scale-spaces have also been defined, and the morphological opening-closing scale-space will be used in this paper.

Abbildung 3.10: *Globales Alignment von zwei Aufnahmen*



Morphological Appropriate Scale Measurements for Region Segmentation

Ullrich Köthe
Fraunhofer Institute for Computer Graphics Rostock
Joachim-Jungius-Str. 9, 18059 Rostock, Germany
Email: koethe@egd.igd.fhg.de

Abstract

This paper presents a novel approach to selecting appropriate scales in morphological opening-closing scale-space. It is based on a morphological band-pass filter that decomposes an image into structures of different size *and* different curvature polarity ("light and dark blobs"). Appropriate scale is defined as the scale that maximizes the response of the bandpass. The resulting scale measurements allow to automatically select window sizes (scales) for segmentation operators. The application of this idea to region segmentation gives very satisfying results.

1 Introduction

When extracting low-level features from images, traditionally one has to adjust a number of parameters, most notably window and filter sizes. Often the adjustment of these parameters is done by hand thus adapting a certain algorithm to a particular class of images (or in some cases just one particular image). The elimination of these parameters (or their automatic adaption) has been the focus of much research in the past.

A new basis for the solution of this problem was laid by the observation that choosing window sizes is actually a problem of scale selection. Any image feature is only visible over a certain range of scale. By introducing a new dimension that represents scale we can transform the image into a family of gradually simplifyed versions of itself, the so called scale-space (Witkin 1983 [9]).

Depending on the particular algorithm used for smoothing different scale-spaces will result. Gaussian scale-space is the only linear and thus most common variant of this idea. It is defined by a convolution of the image $f(\vec{x})$ with a Gaussian kernel where the scale parameter determines the width of the Gaussian. A number of nonlinear scale-spaces have also been defined, and the morphological opening-closing scale-space will be used in this paper.

Abbildung 3.11: *Lokale Offsets und das Resultatbild*

das Resultatbild dargestellt. Die leeren Stellen zwischen einzelnen Regionen (als dünne Linien erkennbar) zeigen, wie die einzelnen Bildregionen verschoben wurden. Später sollen sie mit der Hintergrundfarbe gefüllt werden.

Kapitel 4

Fusion von Aufnahmen

Die Technik der Bildfusion kann erfolgreich dazu verwendet werden, um Serien von Aufnahmen zu einem Ausgabebild zusammen zu fügen und dabei aus jeder Aufnahme nur die Teile zu nehmen, die den angegebenen Auswahlkriterien am besten entsprechen. Als Auswahlkriterien dienen dabei Qualitätsmaße von Bildregionen oder von einzelnen Pixeln. Diverse Qualitätsmaße für Bildregionen wurden bereits im Abschnitt 2.2.2 und für Pixel im Abschnitt 2.2.5 beschrieben. Entsprechend den Auswahlkriterien unterteilen sich die Methoden der Fusion in *Regionen-* und *Pixelbasierte*.

In weiteren Abschnitten werden beide Verfahren näher betrachtet und Testergebnisse dargestellt. Dafür werden auf der Abbildung 4.1 dargestellte Aufnahmen mit unterschiedlichen Belichtungszeiten verwendet. Die Aufnahmen wurden mit einer 2-Megapixel Web-Kamera gemacht und haben eine Größe von 1200×1600 Pixel.

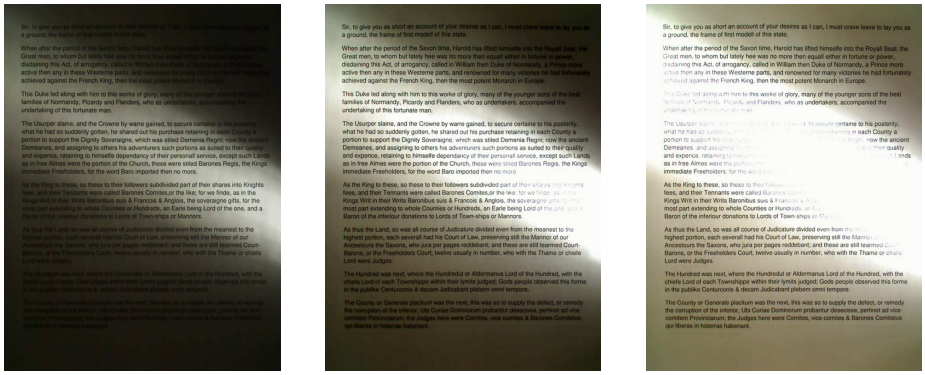


Abbildung 4.1: *Originalaufnahmen mit unterschiedlichen Belichtungszeiten*

4.1 Regionenbasierte Methoden

Die Idee bei den regionenbasierten Methoden besteht darin, in jedem Bild aus einer Aufnahmeserie die Regionen zu bestimmen, die dem Auswahlkriterium, wie z.B. die beste Fokussierung oder der höchste Informationsinhalt, am besten entsprechen. Dafür wird jedes Bild in rechteckige Regionen geteilt, für jede Region ein Qualitätsmaß berechnet und anschließend das Resultatbild aus Regionen mit höchstem Qualitätsmaß zusammen gestellt. Eine ausführliche Beschreibung von diesem Verfahren wurde bereits im Abschnitt 2.2.3 gegeben.

Hier ist es wichtig noch mal anzumerken, dass die Regionengröße ein wichtiger Parameter ist, der in den regionenbasierten Methoden eine entscheidende Rolle spielt. Dabei können sich die optimalen Regionengrößen für unterschiedliche Qualitätsmaße unterscheiden. Um eine optimale Regionengröße für ein Qualitätsmaß zu bestimmen, wird das Bildfusion-Verfahren mit jeder der zu testenden Regionengrößen komplett durchgeführt und ein Resultatbild erstellt. Anschließend wird für jedes entstandene Ausgabebild das Qualitätsmaß berechnet, diesmal nicht für einzelne Regionen, sondern für das gesamte Bild, und somit wird die optimale Regionengröße und gleichzeitig auch das beste Resultatbild ermittelt.

Im Weiteren wird praktische Anwendung von regionenbasierten Methoden beschrieben und Ergebnisse dargestellt.

4.1.1 Entropie von Bildregionen

Diese Methode basiert auf der Idee, dass die Teile von Aufnahmen die über- oder unterbelichtet sind, weniger Information enthalten als gut belichtete [9]. Um den Informationsinhalt zu bestimmen wird Entropie von einzelnen Bildregionen berechnet. Anschließend wird das Resultatbild aus den informationsreichsten Regionen zusammen gestellt.

Auf dem Resultatbild sind die Übergänge zwischen einzelnen Bildregionen stark sichtbar. Um diesen unerwünschten Effekt zu vermeiden werden die ermittelten besten Bildteile nicht einfach ins Ausgabebild kopiert, sondern mit Hilfe von einem *Blendingverfahren* (siehe Abschnitt 4.2) miteinander verschmolzen, so dass die Grenzen zwischen den Bildteilen nicht mehr sichtbar sind.

4.1.2 Maße der Fokussierung

In diesem Abschnitt werden Ergebnisse der Bild-Fusion mit Verwendung von Maßen der Fokussierung als Auswahlkriterium für Bildregionen dargestellt. Obwohl die Fokussierungsmaße nicht für Fusion von unterschiedlich belichteten Aufnahmen gedacht sind, können sie auch für diesen Zweck erfolgreich verwen-

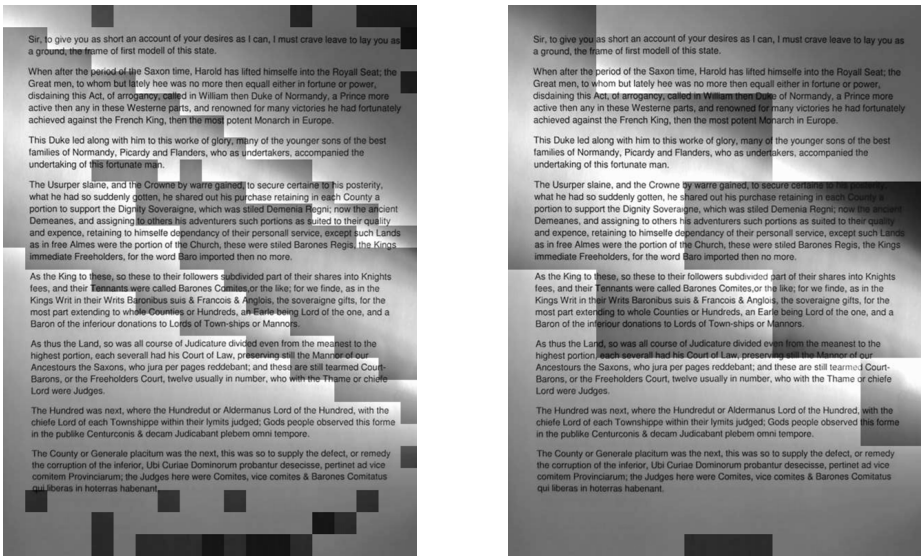


Abbildung 4.2: *Regionenbasierte Bild-Fusion mit Entropie als Auswahlkriterium für Regionen. Links - Regionengröße von 64 Pixel, rechts - 256 Pixel.*

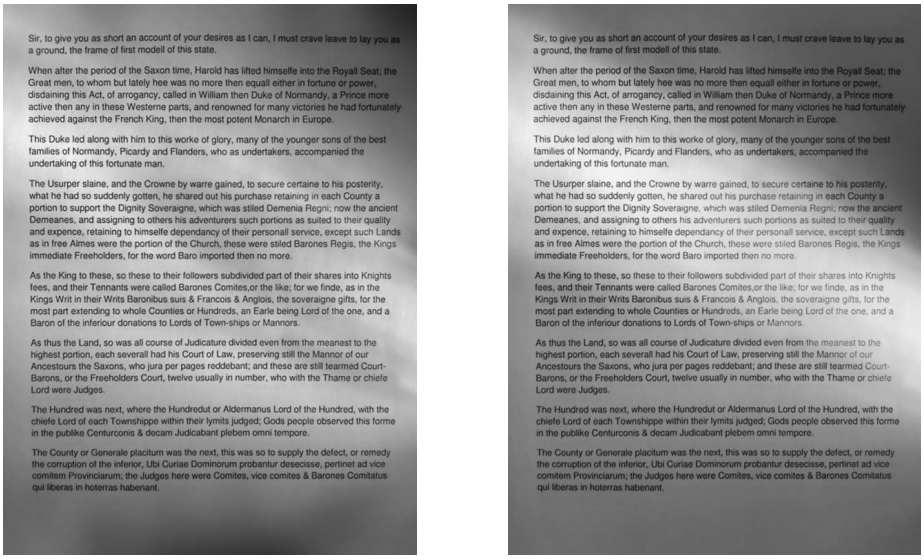


Abbildung 4.3: *Regionenbasierte Bild-Fusion mit Regionenentropie und Verwendung vom Blending-Verfahren. Links - Regionengröße von 64 Pixel, rechts - 256 Pixel.*

det werden. Die Testergebnisse zeigen, dass die Verwendung von Fokussierungsmaße fast die gleichen Ergebnisse liefert als Regionenentropie. Ein ausführlicher Vergleich wird im Kapitel 5 gegeben.

Spatial Frequency

Dieses Qualitätsmaß basiert auf Berechnung der Differenz zwischen benachbarten Pixel. Somit wird für homogene, verschwommene Bildregionen ein kleinerer Wert berechnet als für Regionen mit scharfen, präzisen Kanten. Und weil überbelichtete Regionen meistens homogen weiß und unterbelichtete homogen schwarz sind, bekommen sie kleinere Werte bei der Regionenauswahl und werden somit nicht in das Resultatbild mit reingenommen.

Varianz

Dieses einfache Qualitätsmaß basiert auf Berechnung der Varianz von Grauwerten im Bild. Dafür wird zuerst der Mittelwert aller Pixelwerte in gegebener Bildregion berechnet und dann die Differenzen zwischen Pixelwerten und dem Mittelwert summiert. Je kleiner die Varianz, um so homogener die Bildregion. Höchste Varianz wird in kontrastreichsten Regionen erreicht.

Energy of Image Gradient

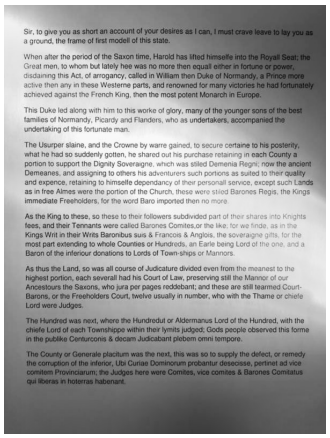
Bei diesem Verfahren wird ein Bild als eine Funktion von zwei Variablen betrachtet und Gradienten dieser Funktion (Bildgradienten) berechnet. Große Änderung zwischen zwei benachbarten Pixelwerten liefert einen größeren Gradienten. Somit können die kontrastreichsten Bildteile ermittelt werden.

Tenengrad

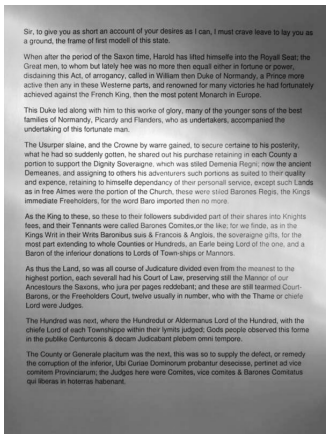
Dieses Qualitätsmaß basiert auch auf Berechnung von Bildgradienten. Aber hier werden Gradienten, die kleiner als ein vorgegebener Wert sind, ignoriert. Die Idee dabei ist, dass Pixel mit kleinen Gradientenwerten eine relativ homogene Nachbarschaft haben und kein Interesse für Berechnung des Fokusmaßes darstellen.

Energy of Laplacian

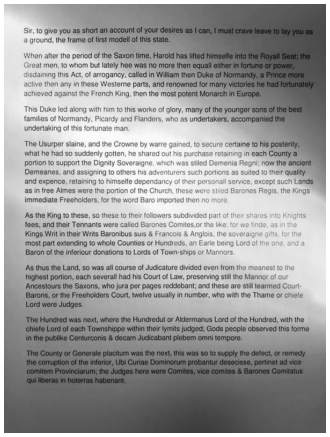
Bei der Berechnung dieses Qualitätsmaßes wird eine Faltungsmaske verwendet, die eine Modifikation der diskreten Approximation des Laplace-Operators darstellt (Formel 2.31). Diese Maske reagiert nicht nur auf vertikale und horizontale Kanten, sondern auch auf diagonale, somit wird mehr Information über die Umgebung eines Pixels analysiert.



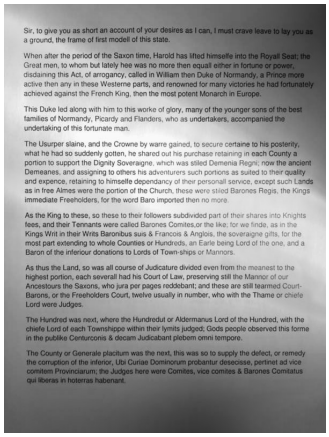
a) Spatial Frequency



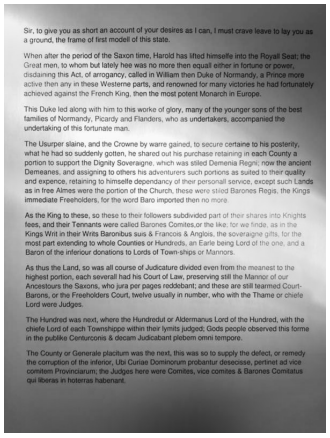
b) Varianz



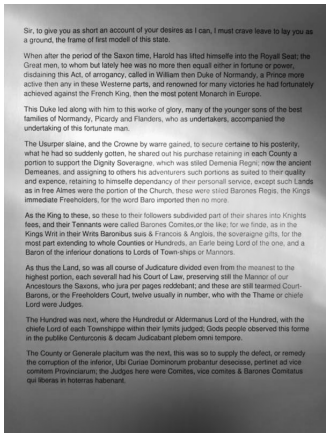
c) Energy of Image Gradient



d) Tenengrad



e) Energy of Laplacian



f) Sum-modified Laplacian

Abbildung 4.4: *Ergebnisse der regionenbasierten Methoden mit Verwendung von Maßen der Fokussierung für Regionenauswahl. Ergebnisse sehen fast gleich aus.*

Sum-modified Laplacian

Dies ist eine Modifikation von *Energy-of-Laplacian*. Es wurde ein *modified Laplacian Operator* eingeführt, der besser für Analyse von Bildern mit Texturen passt. Dabei wird wie bei *Tenengrad* ein Schwellenwert verwendet um Pixel mit homogener Umgebung aus der Analyse auszuschließen.

4.2 Blending

Das Blendingverfahren wird in der regionenbasierten Bildfusion verwendet um sichtbare Grenzen zwischen den Bildteilen zu glätten. Ein Resultatbild mit sichtbaren Übergängen ist auf der Abbildung 4.2 dargestellt, und das entsprechende Bild mit Verwendung von Blending - auf der Abbildung 4.3.

Eine ausführliche Beschreibung des Verfahrens wurde bereits im Abschnitt 2.2.4 gegeben. Hier werden noch einige Ideen für Optimierung vorgestellt.

Optimierung

Nicht alle Bildregionen müssen beim Blending verschmolzen werden. Wenn im Resultatbild ein Block nur von Blöcken aus dem gleichen Originalbild umrandet wird, dann müssen keine Übergänge dazwischen geglättet werden. D.h., es müssen nur die Regionen geblendet werden, die mit einer Region aus einem anderen Originalbild benachbart sind.

Noch eine Möglichkeit für Optimierung wurde in der Arbeit von S. Li, J. Kwok und Y. Wang über Focus-Fusion vorgeschlagen [12]. Ein Block im Resultatbild, der in seiner 8-Nachbarschaft die Mehrheit der Blöcke aus einem anderen Bild $\mathbf{I}_j$ hat, wurde durch den entsprechenden Block aus dem Bild $\mathbf{I}_j$ ersetzt. Das verringert die Anzahl von sichtbaren Übergängen, und beschleunigt das spätere Blending.

Und noch die letzte Optimierung beim Blending - Quantisierung bei Berechnung des Gewichts in der Blendingfunktion. Der Pixelwert $\mathbf{O}(x, y)$ im Ausgabebild wird nach der Formel 2.36 aus dem Abschnitt 2.2.4 berechnet:

$$\mathbf{O}(x, y) = \sum_{j=1}^{N_r} \sum_{k=1}^{N_c} W_{jk}(x, y) \mathbf{I}'_{jk}(x, y)$$

Das Gewicht $W_{jk}(x, y)$ gibt an, wie stark der Pixelwert $\mathbf{I}'_{jk}(x, y)$ im Ausgabewert $\mathbf{O}(x, y)$ ausgeprägt wird. Dieses Gewicht wird für jedes Pixel neu berechnet. Die Idee der Quantisierung besteht darin, die aufwendige Gewichtsberechnung nur für jedes n-te Pixel durchzuführen, die $n-1$ nächsten Pixel bekommen dabei das Gewicht des Vorgängers. Es wird davon ausgegangen, dass der Unterschied

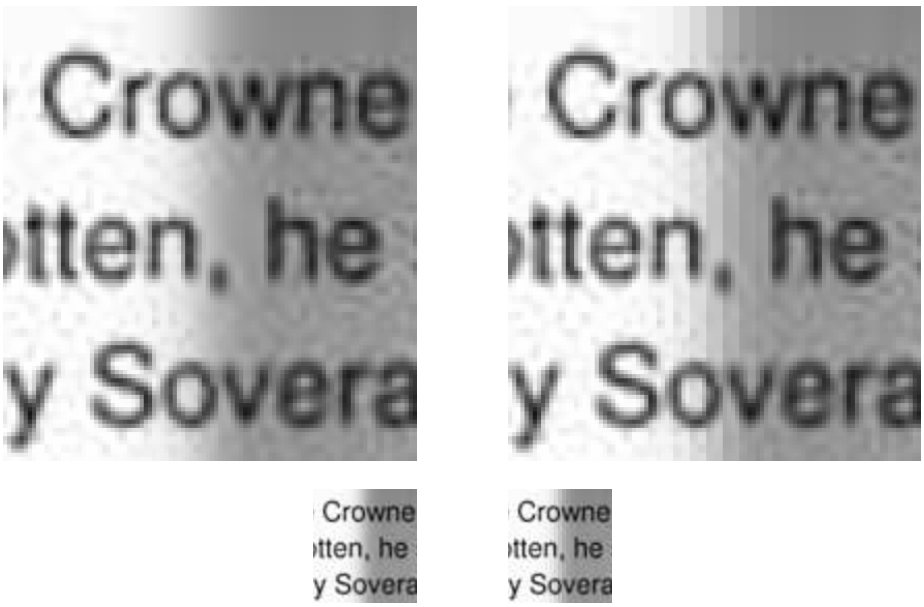


Abbildung 4.5: *Quantisierung bei der Gewichtsberechnung. Links - Gewichte wurden für jedes Pixel berechnet, rechts - nur für jedes 4-te Pixel. Oben sind auf 400% vergrößerte Ausschnitte, unten - in Originalgröße.*

der Pixelwerte in lokaler Nachbarschaft sehr klein ist und dementsprechend klein wird auch die Änderung des Gewichts.

Experimente haben gezeigt, dass, wenn die Gewichte für jedes zweite Pixel aktualisiert werden, dann ist es auf dem Resultatbild kaum zu merken, dabei wird die Hälfte der Gewichtsberechnungen erspart - eine Beschleunigung von 50%.

Zur Veranschaulichung der Idee der Gewichtsquantisierung sind auf der Abbildung 4.5 zwei Bildausschnitte - mit und ohne Quantisierung dargestellt. Der Unterschied zwischen zwei Bilder in Originalgröße ist kaum zu erkennen.

4.3 Pixelbasierte Methoden

Pixelbasierte Methoden stellen das Resultatbild nicht aus den Blöcken, sondern aus einzelnen Pixeln der Originalaufnahmen. Dafür werden Eigenschaften jedes Pixels oder seiner lokalen Umgebung analysiert.

Mittelwert

Es wird ein Mittelwert aller an der gleichen Position stehenden Pixel bestimmt. Dadurch werden dunkle Stellen heller und helle dunkler. Das verschlechtert leider in einigen Fällen das Resultatbild, wenn z.B. auf einer Aufnahme ein Teil

gut sichtbar ist, aber auf einer anderen über- oder unterbelichtet ist, dann wird nach der Berechnung des Mittelwertes der Kontrast verschlechtert.

Gewichtete Summe

Hier werden auch alle Pixel in die Berechnung des Ausgabewertes miteinbezogen. Das Resultatbild hat wenig Kontrast weil gut sichtbare Teile der Aufnahmen durch Kombination mit schlecht belichteten oft nur verschlechtert werden.

Median

Idee dieser Methode besteht darin, keine neuen Pixelwerte berechnen, sondern den Median der Pixelwerte zu bestimmen. Wenn ein Bildteil auf einer Aufnahme gut sichtbar ist und auf anderen über- und unterbelichtet, dann wird als Median das gut sichtbare Pixel gewählt.

Pixel Entropie

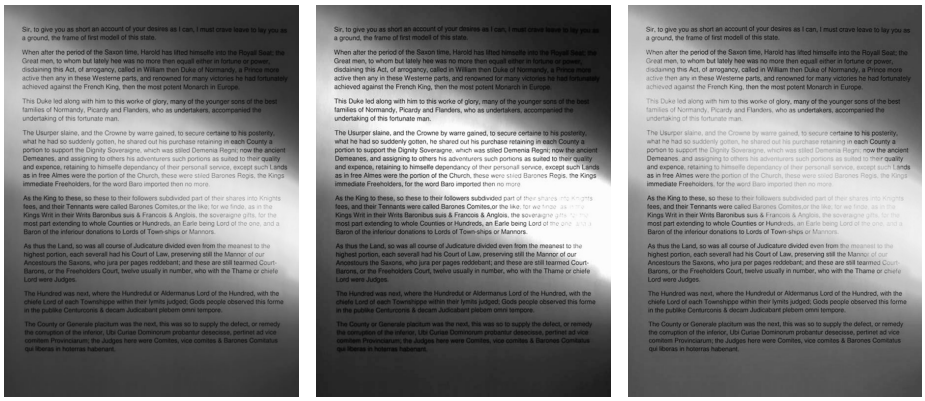
Bei dieser Methode wird für jedes Pixel seine Nachbarschaft analysiert um Pixel mit informationsreichster Umgebung zu bestimmen. Dafür wird Entropie der Pixelnachbarschaft berechnet. In das Resultatbild kommt das Pixel mit höchsten Entropie (siehe Formel 2.43).

Die Experimente haben gezeigt, dass im Resultatbild, an den Stellen wo sich Pixel aus unterschiedlichen Aufnahmen treffen, sichtbare Kanten entstehen können. Um diesen Effekt zu eliminieren wird eine gewichtete Summe der Pixelwerte gebildet, je höher die Entropie eines Pixels, desto größer ist sein Gewicht und desto stärker wird seine Intensität im neuen Pixelwert ausgeprägt (Formel 2.45).

Auf der Abbildung 4.6 sind Ergebnisse von *Mittelwert-*, *Median-* und *gewichtete-Summen-Methoden* dargestellt. Die Ergebnisse der Pixel-Entropie-Methode sind auf der Abbildung 4.7 dargestellt. Linkes Bild wurde ohne Gewichtung erstellt und weist sichtbare Kanten zwischen einigen Bildregionen auf, auf dem rechten Bild wurden die Kanten durch Verwendung von gewichteten Summen eliminiert.

4.4 Methode der Kantenintensitäten

In dieser Methode werden Kantenintensitäten in lokaler Nachbarschaft jedes Pixels berechnet. Dafür wird jede Aufnahme zuerst mit einem Gauß-Kern geglättet, und dann die Differenz zwischen dem Pixelwert im Originalbild und im geglätteten Bild berechnet. Die Differenz wird in der Regel mit einer Skalierungsvariable verstärkt. Anschließend werden anhand der berechneten Kanteninten-



a) Mittelwert *b) Median* *c) Gewichtete Summe*

Abbildung 4.6: *Ergebnisse der pixelbasierten Methoden*

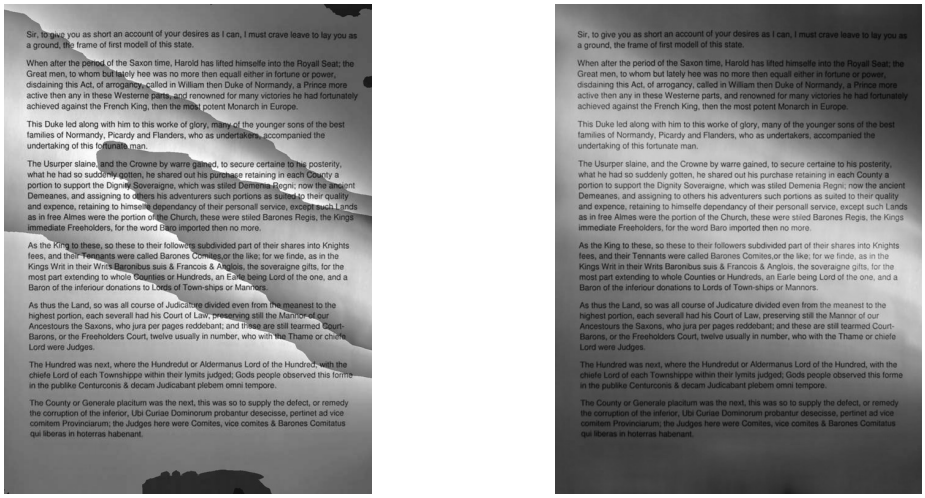


Abbildung 4.7: *Ergebnisse der Pixel-Entropie-Methode. Links - stark sichtbare Kanten, rechts - keine Kanten dank Verwendung von gewichteten Summen.*

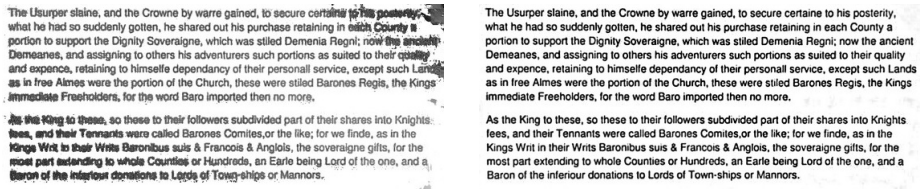


Abbildung 4.8: *Ergebnisse der Kantenintensitäten-Methode. Links - gewichtete Summe der Pixelwerte, rechts - gewichtete Summe der Kantenintensitäten.*

sitäten Gewichte berechnet, mit denen der neue Pixelwert als eine gewichtete Summe berechnet wird.

Hierbei stellte es sich heraus, dass die gewichtete Summe der Pixelwerte aus Originalbildern keine gute Ergebnisse liefert. Aber wenn statt Originalwerten die Kantenintensitäten in der Summenbildung verwendet werden, sieht das Resultatbild deutlich besser aus (Abbildung 4.8).

Eine ausführliche Beschreibung dieser Methode wurde bereits im Abschnitt 2.2.5 gegeben. Hier ist es nur noch zu bemerken, dass in der Formel 2.50 statt Pixelwertes $\mathbf{I}_i(x, y)$ aus der Aufnahme $\mathbf{I}_i$ seine Kantenintensität $\mathbf{E}_i^G(x, y)$ verwendet wird. Die neue Formel ist dann:

$$\mathbf{O}^{intens}(x, y) = \sum_{i=0}^{N-1} w_i^{intens}(x, y)\mathbf{E}_i^G(x, y) \tag{4.1}$$

Die Größe der Nachbarschaft wird durch die Größe des Gauß-Kerns bestimmt. Wie es auf der Abbildung 4.9 zu sehen ist, stärkere Kanten und bessere Resultate werden mit größeren Kernen erzielt. Bei großen Kernen ist aber der Rechenaufwand höher, deswegen ist es in der Praxis immer wichtig eine optimale Kerngröße zu finden.

4.5 Experimente und Ergebnisse mit Focus-Fusion

Die in Abschnitten 2.2.3 und 4.1 vorgestellten regionenbasierten Methode lassen sich ebenfalls dafür verwenden, Aufnahmen mit unterschiedlichen Fokussierungen zu einer Aufnahme zusammenzufügen. Hierbei werden die im Abschnitt 4.1.2 beschriebenen Fokussierungsmaße verwendet.

Im folgenden werden einige Experimente dazu vorgestellt. In jedem Experiment wurden zwei Aufnahmen der gleichen Szene mit unterschiedlichen Fokussierungen gemacht. In einer Aufnahme wurde auf ein nahes Objekt fokussiert und in der anderen auf ein fernes.

Aufnahmen wurden mit Hilfe von regionenbasierten Fusionsmethoden bearbeitet, dabei wurden für Regionenauswahl Qualitätsmaße der Fokussierung

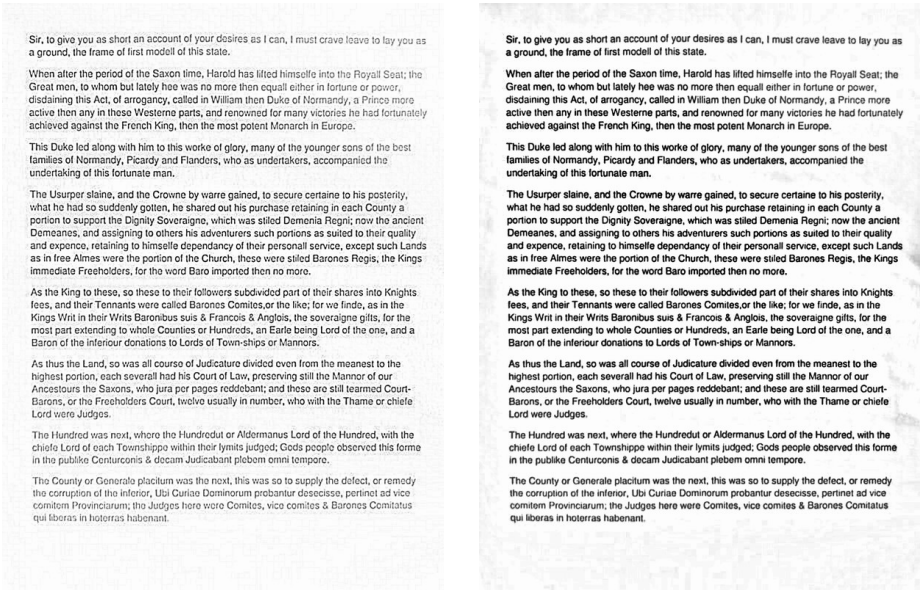


Abbildung 4.9: *Ergebnisse der Kantenintensitäten-Methode. Links - mit Kernelgröße* 21×21*, rechts - Kernelgröße* 121×121.

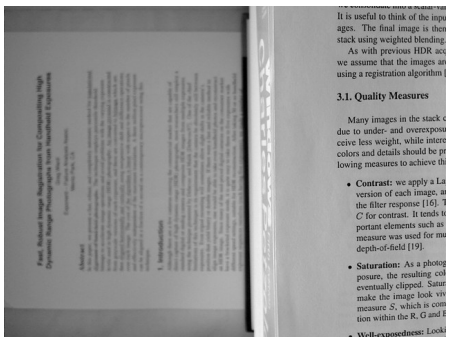
verwendet. Anschließend wurden die noch sichtbaren Grenzen zwischen einigen Bildteilen mit dem Blendingverfahren geglättet.

Auf der Abbildung 4.10 sind Ergebnisse der Focus-Fusion von zwei mit einer Web-Kamera gemachten Aufnahmen. Als Resultatbilder sind Ergebnisse der *Sum-modified-Laplacian* und *Spatial Frequency* Methoden gezeigt. Bei der *SML*-Methode wurde das beste Ergebnis mit Regiongröße von 32 Pixel erzielt, bei *Spatial Frequency* - 128 Pixel. Der Unterschied zwischen den beiden Resultaten ist kaun zu merken, und im Allgemeinen lässt sich behaupten, dass die besten Resultate der getesteten Focus-Fusion Methoden in etwa gleich aussehen.

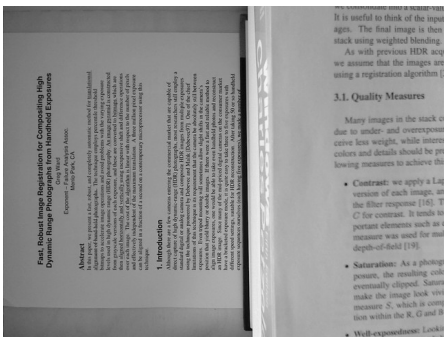
Auf der Abbildung 4.11 sind Resultate der Focus-Fusion von jeweils zwei Testaufnahmen gezeigt. Das sind bekannte Testbilder, die oft in Experimenten mit Fokussierung verwendet werden [11, 12]. Eine der Aufnahmen ist auf einem nahen und andere auf einem fernen Objekt fokussiert.

Es wurden auch hier regionenbasierte Fusionsmethoden mit unterschiedlichen Fokussierungsmaßen getestet. Die besten Ergebnisse wurden bei allen Methoden mit Regiongröße von 32 Pixel erzielt. Dieser Test lässt ebenfalls behaupten, dass die besten Resultate der getesteten Methoden in etwa gleich aussehen.

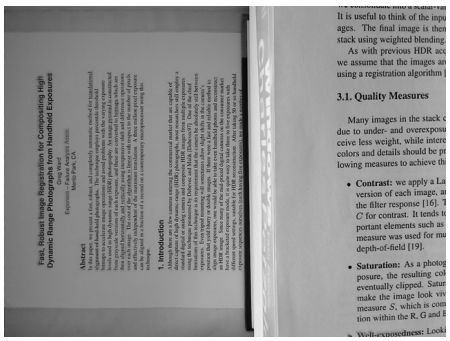
Die bei Focus-Fusion verwendeten Qualitätsmaße können außerdem erfolgreich bei der Suche nach der best fokussierten aus einer Reihe von Aufnahmen eingesetzt werden. Dafür wird nicht mit einzelnen Bildregionen gearbeitet, sondern ein Fokussierungsmaß für das ganze Bild berechnet. Auf diese Weise kann



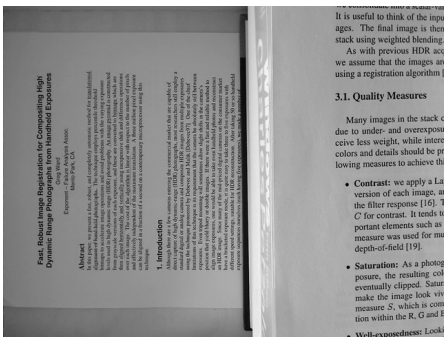
a) Fokus auf dem nahen Objekt



b) Fokus auf dem fernen Objekt



c) Sum-modified-Laplacian Methode



d) Spatial-Frequency Methode

Abbildung 4.10: *Ergebnisse der Focus-Fusion. Oben - zwei Aufnahmen mit jeweils einem defokussierten Teil, unten - das ganze Bild ist im Fokus.*

Auto-Fokus realisiert werden, indem eine Serie von unterschiedlich fokussierten Aufnahmen in niedriger Auflösung gemacht wird, dann ein Fokussierungsmaß für sie berechnet und so der beste Fokus ermittelt. Anschließend wird eine Aufnahme in gewünschter Auflösung mit dem ermittelten Fokus gemacht.

1.a) Fokus nah

1.b) Fokus fern

1.c) Sum-modified-Laplacian Methode

1.d) Spatial-Frequency Methode

2.a) Fokus nah

2.b) Fokus fern

2.c) Sum-modified-Laplacian Methode

2.d) Spatial-Frequency Methode

Abbildung 4.11: *Ergebnisse der Focus-Fusion.*

Kapitel 5

Auswertung der Exposure-Fusion Methoden

In diesem Kapitel werden Ergebnisse einer umfangreichen Auswertung vorgestellt. Es wurden mehrere Aufnahmeserien eines Dokumentes in unterschiedlichen Lichtverhältnissen mit unterschiedlichen Belichtungszeiten gemacht. Die Aufnahmeserien wurden anschließend mit allen Exposure-Fusion Methoden getestet und die entstandenen Resultatbilder mit Hilfe von einem OCR-System ausgewertet.

5.1 Hardwarebeschreibung

Aufnahmen wurden mit einer 2-Megapixel Web-Kamera *Logitech Quickcam Pro 9000* gemacht. Die Kamera hat eine gute Carl Zeiss Optik und lässt sich per Rechner fernsteuern: Fokus und Belichtungszeit automatisch und, was besonders wichtig, manuell einstellen und Aufnahmen machen. Um Verschiebungen der Kamera auszuschließen, wurde sie auf einem stabilen Stativ befestigt und vom Rechner ferngesteuert, das aufgenommene Dokument wurde ebenfalls befestigt. Dadurch wurde erreicht, dass alle Aufnahmen einer Aufnahmeserie perfekt gegeneinander ausgerichtet wurden und Alignmentkorrekturen somit unnötig waren.

Um unterschiedliche Lichtverhältnisse zu simulieren, wurden Aufnahmen mit natürlichen und künstlichen Lichtquellen gemacht. Damit in jeder Aufnahmeserie über- und unterbelichtete Bilder vorkommen, wurden auch Schatteneffekte nachgebildet.

5.2 Erstellung von Aufnahmeserien

Um die Erstellung von Aufnahmeserien zu automatisieren, wurde eine Software geschrieben, die die Kamera gesteuert und Aufnahmen gemacht und gespeichert hat. Dabei wurde die maximale Auflösung der Kamera von 1200×1600 Pixel verwendet. Für Belichtungszeiten wurden bei allen Aufnahmeserien drei feste Werte verwendet: $1/5\,s$, $1/15\,s$, $1/63\,s$. Damit die Kamera sich nicht nach jeder Änderung der Belichtungszeit automatisch neu fokussiert, wurde sie einmal am Anfang des Experimentes fokussiert und danach wurde der Autofokus ausgeschaltet.

Auf diese Weise wurden 40 Aufnahmeserien mit jeweils drei Aufnahmen eines Dokumentes gemacht.

5.3 Exposure-Fusion

Es wurden insgesamt 8 Exposure-Fusion Methoden getestet. Fünf regionenbasierte:

1. Region Entropy
2. Spatial Frequency
3. Varianz
4. Energy-of-Laplacian
5. Sum-modified-Laplacian

und drei pixelbasierte:

1. Pixel Median
2. Pixel Entropy
3. Kantenintensitäten

Jede von dieser Methoden hat eine eigene Liste von Parametern und unterschiedliche Definitionsbereiche für jeden Parameter. Deswegen wurde für jede Methode eine Liste von möglichen (und sinnvollen) Parameterwerten generiert. Danach wurden automatisch, mit Hilfe von dem entsprechenden Qualitätsmaß, die besten Parameter bestimmt und das bestmögliche Resultatbild erzeugt.

Somit wurden pro Aufnahmeserie acht fusionierte Bilder erstellt. Also, insgesamt für 40 Aufnahmeserien - 320 Bilder.

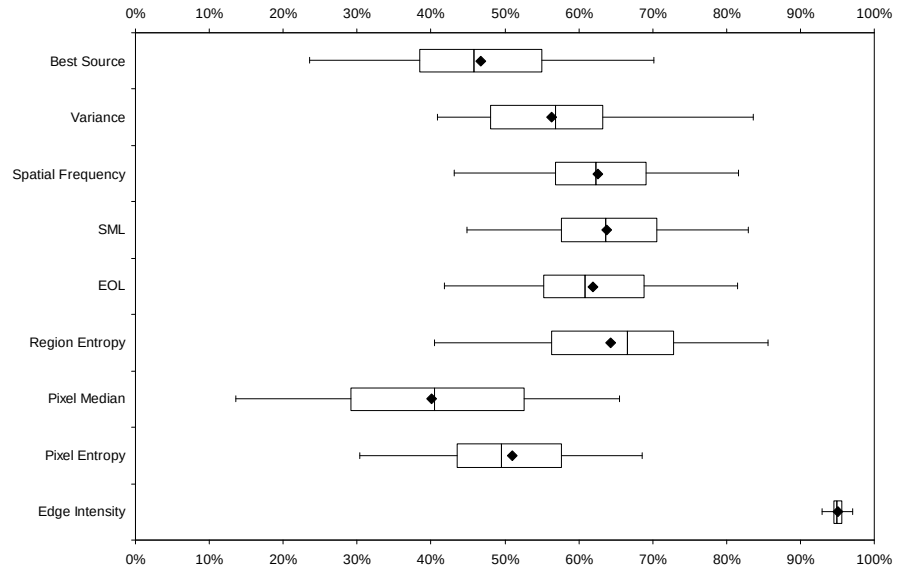


Abbildung 5.1: *Erkennungsraten von verschiedenen Exposure-Fusion Methoden*

5.4 Auswertung von Resultatbildern

Um zu bestimmen, welche Methode die besten Ergebnisse liefert, wurden alle Resultatbilder mit Hilfe des Texterkennungssystems *Tesseract-OCR 2.03* [17] analysiert und der erkannte Text gespeichert.

Schließlich wurde mit dem Algorithmus von Needleman-Wunsch [18] die Ähnlichkeit des erkannten Textes zum Text des Originaldokumentes berechnet. Somit wurden die Erkennungsraten bestimmt.

Auf der Abbildung 5.1 ist ein Box-Whisker Diagramm mit Erkennungsraten der getesteten Exposure-Fusion Methoden dargestellt.

Die obere Zeile *"Source"* zeigt die Erkennungsraten der Originalaufnahmen. Eine Aufnahmeserie enthält drei unterschiedlich belichtete Aufnahmen, sie wurden alle mit dem OCR-System erkannt und auf dem Diagramm nur die höchste Erkennungsrate dargestellt, d.h., es wurde nur die best belichtete Aufnahme betrachtet. Somit ist auf dem Diagramm zu sehen, ob eine Anwendung von einer Exposure-Fusion Methode sinnvoll war, und wie hoch dabei die Erkennungsrate steigt.

In der Tabelle 5.1 sind die Erkennungsraten noch mal in Prozent dargestellt. Wie es aus der Tabelle ersichtlich ist, sogar wenn die beste Aufnahme in jeder Aufnahmeserie ausgewählt wird, im Mittel wird eine Erkennungsrate von nur 47% erreicht.

Regionenbasierte Methoden, mit Ausnahme der einfachsten *Varianz* Me-

Methode	Min	Mittel	Max
Best Source	24	47	70
Variance	41	56	84
Spatial Frequency	43	63	82
SML	45	64	83
EOL	42	62	81
Region Entropy	40	64	86
Pixel Median	14	40	66
Pixel Entropy	30	51	69
Edge Intensity	92	95	97

Tabelle 5.1: *Erkennungsraten (in Prozent) von verschiedenen Exposure-Fusion Methoden*

thode, haben im Schnitt viel bessere Erkennungsraten als Originalaufnahmen. Obwohl die verwendeten Qualitätsmaße - *Spatial Frequency*, *Sum-Modified Laplacian* und *Energy-of-Laplacian* für die Ermittlung der best fokussierten Aufnahmen gedacht sind, liefern sie auch bei Exposure-Fusion gute Ergebnisse - 62-64%. *Region Entropy* mit 64% Erkennungsrate im Schnitt hat sich ebenfalls als gutes Qualitätsmaß bewährt.

Unter den pixelbasierten Methoden, wie es auch zu erwarten war, liefert die *Pixel Median* Methode die schlechtesten Ergebnisse, sogar schlechter als Originalaufnahmen. Das ist dadurch begründet, dass bei der Berechnung des neuen Pixelwertes nicht berücksichtigt wird, ob ein Pixel aus einer unter- oder überbelichteten Bildregion kommt und ob seine Verwendung in der Berechnung des neuen Pixelwertes sinnvoll ist. Die *Pixel Entropy* Methode hat im Schnitt etwas bessere Erkennungsraten als Originalaufnahmen, aber angesichts der sehr aufwändigen Berechnung ist sie kaum brauchbar.

Die besten Ergebnisse wurden mit der *Edge Intensity* Methode (*Kantenintensitäten*) erreicht. Sogar die niedrigste Erkennungsrate - 92% ist größer als die höchste der anderen Methoden - 86% bei *Region Entropy*. Die im Schnitt erreichte Erkennungsrate von 95% zeigt, dass die Methode erfolgreich für die Fusion von Dokumentenaufnahmen verwendet werden kann.

Kapitel 6

Zusammenfassung

In dieser Arbeit wurden Methoden der Bildfusion vorgestellt und eine praktische Anwendung dieser Methoden für die Fusion von Dokumentenaufnahmen gezeigt. Außerdem wurde eine umfangreiche Auswertung der vorgestellten Methoden durchgeführt und die beste Methode ermittelt. Für die Auswertung wurden 40 Aufnahmeserien eines Dokumentes bei unterschiedlichen Lichtverhältnissen gemacht, dabei wurden sowohl natürliche als auch künstliche Lichtquellen verwendet. Außerdem wurden auch Schatteneffekte nachgebildet, damit in jeder Aufnahmeserie über- und unterbelichtete Teile des Dokumentes vorkommen. Pro Aufnahmeserie wurden jeweils drei Aufnahmen mit unterschiedlichen Belichtungszeiten gemacht, dafür wurde eine per Rechner gesteuerte 2-Megapixel Web-Kamera verwendet.

Für die eigentliche Auswertung der Resultatbilder verschiedenen Fusionsmethoden wurde jedes Bild mit Hilfe des Texterkennungssystems *Tesseract-OCR 2.03* analysiert. Für den erkannten Text wurde dann mit dem Needleman-Wunsch-Algorithmus seine Ähnlichkeit zum Text des Originaldokumentes berechnet. Auf diese Weise wurden die Erkennungsraten für alle Methoden bestimmt.

Im Allgemeinen lassen sich die Methoden der Bildfusion in zwei Klassen einteilen - *regionenbasierte* und *pixelbasierte*. Regionenbasierte Methoden teilen Originalaufnahmen in Blöcke einer vorgegebenen Größe, berechnen für jeden Block ein Qualitätsmaß und stellen anschließend das Resultatbild aus den Teilen mit höchstem Qualitätsmaß zusammen. In dieser Arbeit wurden mehrere Qualitätsmaße ausprobiert, wie z.B. *Entropie* - für die Suche nach informationsreichsten Bildteilen, oder *Spatial Frequency* - für die Ermittlung von den best fokussierten Regionen.

Bei der Zusammenstellung des Resultatbildes aus einzelnen Bildteilen können zwischen benachbarten Teilen, die aus den unterschiedlich belichteten Auf-

nahmen stammen, sichtbare Kanten entstehen. Um solche Kanten zu glätten, wird bei der Erstellung des Resultatbildes ein Blending-Verfahren verwendet. Dadurch werden Bildteile so miteinander verschmolzen, dass es keine Übergänge mehr sichtbar sind.

Pixelbasierte Methoden stellen das Resultatbild nicht aus Blöcken, sondern aus den einzelnen Pixel der Originalaufnahmen zusammen. Dafür werden Eigenschaften jedes Pixels oder seiner lokalen Umgebung analysiert. Anhand der Analyse können den Pixeln aus den Originalaufnahmen Gewichte zugewiesen werden, so, dass ein Pixel mit hohem Qualitätsmaß ein entsprechend hohes Gewich bekommt. Für ein Pixel im Resultatbild wird sein Wert als eine gewichtete Summe aus den entsprechenden Pixelwerten der Originalaufnahmen berechnet. Die Experimente haben gezeigt, dass die besten Ergebnisse durch Gewichtung der Pixelwerte erzielt wurden.

An dieser Stelle ist es noch wichtig anzumerken, dass die oben erwähnte Auswertung der Bildfusionsmethoden zeigt, dass die pixelbasierte *Methode der Kantenintensitäten* die mit Abstand besten Ergebnisse liefert - eine Erkennungsrate von 95% im Schnitt. Regionenbasierte Methoden haben dabei eine Erkennungsrate von nur 62-64% im Schnitt. Somit hat sich die *Methode der Kantenintensitäten* als bestens geeignet für die Fusion von Dokumentenaufnahmen gezeigt.

Auf Grundlage der in dieser Arbeit gemachten Experimente mit verschiedenen Bildfusionsmethoden ist eine wissenschaftliche Publikation entstanden [19]. Sie wurde bereits auf der "International Conference on Document Analysis and Recognition" (ICDAR) 2009 vorgestellt und unter dem Titel "*Multi-Exposure Document Fusion Based on Edge-Intensities*" veröffentlicht.

Literaturverzeichnis

[1] Gonzales R.C.,Woods R.E.: "*Digitial Image Processing*", 3.Auflage, ISBN 978- 0131687288, Pearson Verlag, 2008.

[2] Burger W., Burge J.B.: "*Digitale Bildverarbeitung - Eine Einführung mit Java und ImageJ*", ISBN 978-3540309406, Springer Verlag, 2006.

[3] Otsu N.: „*A threshold selection method from gray-level histograms*", IEEE Trans. Systems Man Cybernet 9 (1), pp.62-66, 1979

[4] Szeliski R.: "*Image alignment and stitching"*. In N. Paragios et al., editors, Handbook of Mathematical Models in Computer Vision, pp. 273-292. Springer, 2005.

[5] Ward G.: "*Fast, robust image registration for compositing high dynamic range photographcs from hand-held exposures*". Journal of Graphics Tools, 8(2), pp. 17–30, 2003.

[6] Donato G., Belongie S.: "*Approximation Methods for Thin Plate Spline Mappings and Principal Warps*", ECCV, Vol. 2, pp. 531-542, Kopenhagen/Dänemark, 2002.

[7] Erik Reinhard, Greg Ward, Sumanta Pattanaik, Paul Debevec: "*High Dynamic Range Imaging.: Acquisition, Display, and Image-Based Lighting*", Morgan Kaufmann Publishers. 2005.

[8] Mertens T., Kautz J., Van Reeth F.: "*Exposure fusion*", in Pacific Conference on Computer Graphics and Applications, 2007, pp. 382– 390.

[9] Goshtasby A. A.: *"Fusion of multi-exposure images*", Image and Vision Computing, vol. 23, no. 6, pp. 611–618, Jun. 2005.

[10] Wei Huang and Zhongliang Jing: "*Evaluation of focus measures in multi-focus image fusion*". Pattern Recognition Letters, 28(4): pp. 493–500, Mar. 2007.

[11] Shutao Li and B. Yang: "*Multifocus image fusion using region segmentation and spatial frequency*". Image and Vision Computing, 26(7): pp. 971–979, July 2008.

[12] Shutao Li, James T. Kwok and Yaonan Wang: "*Combination of images with diverse focuses using the spatial frequency*". Information Fusion 2: pp. 169-176, September 2001.

[13] Ligthart G., Groen F.: *"A Comparison of Different Autofocus Algorithms"*. In: Proc. Int. Conf. on Pattern Recognition: pp. 597–600. 1982.

[14] Eskicioglu A. M., Fisher P. S.: "Image Quality Measures and Their Performance", IEEE Transactions On Communications, 43(12), pp: 2959-2965, Dec. 1995.

[15] Eultokhy E. A., Kavusi S.:*"A Computationally Efficient Algorithm for Multi-Focus Image Reconstruction",* Proc. of SPIE Electronic Imaging: pp. 332–341, 2003.

[16] Nayar S. K., Nakagawa Y.: *"Shape from focus"*. IEEE Transactions on Pattern Analysis Machine Intelligence, 16(8): pp. 824-831, Aug. 1994.

[17] Tesseract, *http://code.google.com/p/tesseract-ocr*

[18] Needleman S. B., Wunsch C. D.: "*A general method applicable to the search for similarity in the amino acid sequences of two proteins*", J. Mol. Biol., 48: pp. 443–453, 1970.

[19] Block M., Schaubert M., Wiesel F., Rojas R.: "*Multi-Exposure Document Fusion Based on Edge-Intensities*", The 10th International Conference on Document Analysis and Recognition (ICDAR 2009), Barcelona/Spain, 2009

Anhang A

Erklärung

Hiermit bestätige ich, dass ich alle Hilfsmittel und Hilfen zur Erstellung der Diplomarbeit in der vorliegenden Arbeit angegeben habe. Ich versichere, dass ich die vorliegende Diplomarbeit auf Grundlage der angegebenen Hilfsmittel und Hilfen selbstständig angefertig habe.

Berlin, im November 2009

Maxim Schaubert