

HANSER

Digitale Bildverarbeitung

Thorsten Hermes

Eine praktische Einführung

ISBN 3-446-22969-8

Leseprobe

Weitere Informationen oder Bestellungen unter
<http://www.hanser.de/3-446-22969-8> sowie im Buchhandel

Kapitel 4

Vorverarbeitung

In diesem Kapitel geht es um die Aufbereitung respektive die Verbesserung des durch ein bildgebendes Verfahren gewonnenen digitalen Bildes. Die im Folgenden diskutierten Methoden fallen alle unter den Begriff der (Bild-) *Vorverarbeitung* (engl. *preprocessing*). In der Literatur findet sich auch der Begriff der *Bildrestauration* oder nur der *Restoration*.

Wir wollen uns im Wesentlichen auf die Bereiche

- Kontrastverstärkung (Abschnitt 4.1),
- Rauschunterdrückung (Abschnitt 4.2) sowie
- auf Operationen zur geometrischen Entzerrung (Abschnitt 4.3)

konzentrieren. Die Vorverarbeitung kann sowohl auf Binärbilder durch bereits bekannte Operationen aus der Morphologie wie Öffnen und/oder Schließen als auch auf Grauwert- und/oder Farbbilder angewendet werden.

Diese Transformationen/Operationen lassen sich auch in so genannten Bildpunktoperationen, in lokale Nachbarschaftsoperationen und in globale Nachbarschaftsoperationen einteilen. Die Bildpunktoperationen beziehen sich nur auf jeweils einen einzelnen Bildpunkt des Eingabebildes $E(x, y)$, der beispielsweise mittels einer so genannten *Look-Up-Table* (LUT) in den entsprechenden Bildpunkt des Ausgabebildes $A(x, y)$ transformiert wird.

Lokale Nachbarschaftsoperationen berücksichtigen zur Berechnung des Bildpunktes $A(x, y)$ nicht nur $E(x, y)$, sondern zusätzlich die lokale Umgebung – zum Beispiel 3×3 oder 5×5 , ... – um $E(x, y)$. Globale Nachbarschaftsoperationen berücksichtigen das gesamte Eingabebild E , um einen Bildpunkt in $A(x, y)$ zu bestimmen.

Gerade die Operationen, die eine Nachbarschaft berücksichtigen, sind von beson-

derer Wichtigkeit, da ein Bildpunkt in der Regel immer mit seiner Nachbarschaft mehr oder weniger stark korreliert ist.

An dieser Stelle sei allerdings noch angemerkt, dass die beste Vorverarbeitung *keine* Vorverarbeitung ist. Das bedeutet, dass man schon bei der Bildgebung dafür Sorge tragen sollte, bestmögliche Aufnahmen zu erhalten, damit keine Vorverarbeitung nötig ist. Also sollte darauf geachtet werden, dass die aufzunehmende Szene zum Beispiel möglichst optimal ausgeleuchtet ist, damit keine Kontrastverstärkung notwendig wird. Ebenso sollte darauf geachtet werden, dass ein dem bildgebenden Verfahren inhärentes Rauschen auf ein Minimum reduziert wird, etc.

4.1 Kontrastverbesserung

In einigen Fällen kann durch das bildgebende Verfahren das digitale Bild kontrastarm sein, beispielsweise bei Fotografien von mikroskopischen Objektträgern. Oder – wer kennt das nicht – bei eigenen Fotografien, die nicht richtig ausgeleuchtet sind. Ebenso ist bei den neueren Digitalkameras (hauptsächlich im so genannten End-Consumer-Bereich) die Aufnahmequalität nicht unbedingt die beste. Hier möchte man nachträglich noch die Möglichkeit haben, den Kontrast zu verbessern.

Es existieren verschiedene Wege, um den Kontrast zu verbessern. Das kann während des bildgebenden Prozesses zum Beispiel durch *Look-Up-Tables* geschehen oder auch nachträglich durch eine Transformation, die die Grauwerthäufigkeiten manipuliert. Des Weiteren kann, wie bereits erwähnt, Rauschen auf dem Bild vorhanden sein, das zum Beispiel durch so genannte Rauschfilter reduziert werden kann.

Die in den folgenden Abschnitten vorgestellten Operationen sind in der Regel Bildpunktoperationen, da sie den Intensitätswert eines Bildpunktes nur aufgrund des Intensitätswertes verändern und eine (wie auch immer geartete) Nachbarschaft nicht mit berücksichtigen. Die Mehrzahl dieser Operationen ist homogen, was bedeutet, dass alle Bildpunkte $E(x, y)$ mit derselben Transformationsvorschrift bearbeitet werden. Die im Abschnitt 4.1.3 beschriebene Hintergrundkompensation ist ein Beispiel für eine inhomogene Bildpunktoperation.

4.1.1 Look-Up-Table (LUT)

Die LUT ist eine homogene Grauwerttransformation, bei der die Intensitätswerte in einer vordefinierten Art und Weise (eben der LUT) modifiziert werden. Hierbei werden die Grauwerte $E(x, y)$ einer Grauwertskala durch eine Funktion oder eine Tabelle in ein neue Grauwertskala $A(x, y)$ überführt. Eine LUT kann man verstehen als einen Speicher, der unter der Adresse a einen bestimmten Inhalt b

gespeichert hat.

$$b = LUT(a) \quad (4.1)$$

Immer, wenn die Adresse a angesprochen wird, wird der Speicherinhalt b zurückgegeben. Bezieht man das auf Grauwerte, wird schnell ersichtlich, dass der Inhalt (entspricht einem neuen Grauwert) immer dann zurückgegeben wird, wenn ein bestimmter Grauwert im Eingabebild E auftritt (entspricht der Adresse).

Soll eine LUT als Diagramm dargestellt werden, so ist es üblich, auf der Abszisse (x -Achse – entspricht in Gl. 4.1 dem a) die Eingabebildgrauwerte, und auf der Ordinate (y -Achse – entspricht in Gl. 4.1 dem b) die Ausgabebildgrauwerte abzutragen.

Im Folgenden werden ein paar Beispiele für Transformationsfunktionen (auch Kennlinien) grafisch und analytisch dargestellt. In Abbildung 4.1(a) ist eine lineare Kennlinie dargestellt, die die Eingangswerte x (also die Intensitätswerte von 0..255) auf gleichwertige Ausgangswerte $LUT(x)$ (0...255) abbildet, was bedeutet, dass das Eingabebild unverändert in das Ausgabebild transformiert wird ($x = LUT(x)$ – oder einfacher: $x = y$).

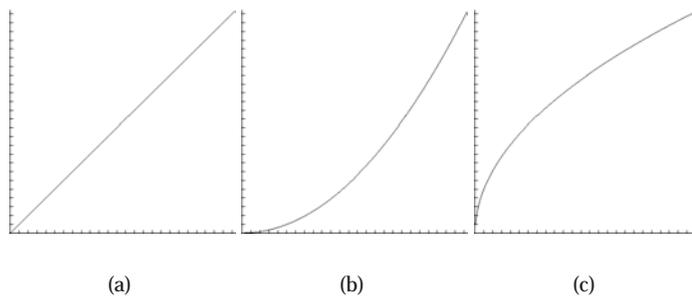


Abbildung 4.1: Drei unterschiedliche Kennlinien, die eine LUT symbolisieren: (a) lineare, (b) quadratische und (c) Wurzelkennlinie. Die Achsen gehen jeweils von 0 bis 255. Die x -Achse steht für die Werte des Eingangsbildes und korrespondierend dazu lassen sich die Werte für das Ausgangsbild an der y -Achse ablesen

Abbildung 4.1(b) zeigt eine quadratische Kennlinie, die die mittleren Intensitätswerte ein wenig abschwächt. In analytischer Form wäre das: $y = \frac{x^2}{255}$. Die 255 im Nenner sorgt dafür, dass die Intensitätswerte im darstellbaren Grauwertbereich bleiben. Abbildung 4.1(c) zeigt eine Wurzelkennlinie. Sie schwächt die mittleren Grauwerte etwas. Dafür wird in den hellen und den dunklen Grauwertbereichen der Kontrast verstärkt. In analytischer Form wäre das $y = \sqrt{255x}$ (auch hier dient die 255 zur Normierung).

Weitere Funktionen, die als LUT-Kennlinie dienen können, wäre eine Binarisierungskennlinie. Sie lässt sich durch eine entsprechende Kennlinie¹ ebenfalls er-

¹ Binarisierungskennlinie mit x_u und x_o als beliebige untere bzw. obere Grenze: $y = \{0, \text{ wenn } x <$

reichen. Ebenso können solche Kennlinien auch stückweise linear sein. Auch die explizite Angabe von Werten in der LUT ist denkbar. Allen LUTs gemein ist die Tatsache, dass der Inhalt der jeweiligen LUT (die Werte für das Ausgangsbild A) lediglich einmal berechnet werden muss. Sie werden dann im Anschluss nur noch abgefragt. Die LUT-Operationen sind in der Regel sehr schnell und werden von einigen Bildverarbeitungssystemen unterstützt.

Für den 2D-Fall (ein Bild mit M Zeilen und N Spalten) kann eine allgemeine Form einer LUT wie folgt festgelegt werden:

$$A(x, y) = LUT(E(x, y)) \quad (4.2)$$

mit $x = 0, \dots, M-1$ und $y = 0, \dots, N-1$.

4.1.2 Histogrammtransformationen

Zunächst soll kurz erläutert werden, was ein *Histogramm* ist und wie man es berechnet, da wir dies für das Verständnis des Folgenden benötigen, um Kontrastverbesserungen durchzuführen.

Ein Histogramm berechnet die *Grauwerthäufigkeit* eines Bildes. Hierzu werden die einzelnen Bildpunkte einer bestimmten Graustufe in ihrer Häufigkeit H abgezählt und über die gesamte Grauwertskala i abgetragen. Dies wird auch als *Statistik erster Ordnung* bezeichnet. Das so berechnete Histogramm $H(i)$ ist eine diskrete Funktion. Aus dem Histogramm $H(i)$ lässt sich durch Normierung auf die Anzahl der betrachteten Bildpunkte N die *relative Histogrammverteilung* $h(i)$ bestimmen:

$$h(i) = \frac{H(i)}{N} \quad (4.3)$$

Die Histogrammtransformationen verändern ähnlich wie die LUTs die Grauwerte im Bild. Allerdings wird hierbei die Grauwertverteilung erst nach der Bildaufnahme verändert. D.h.: Es muss ein digitales Bild vorliegen. Von diesem Bild wird zunächst ein *Histogramm* $H_E(i)$ (mit E als Eingabebild und i Anzahl der Intensitätswerte respektive Grauwerte) berechnet. Histogramme können gedehnt werden, d.h. ein Grauwertbereich von i_u bis i_o (mit $i_u \neq 0$ und $i_o \neq \text{MAXGW} = 255$) wird auf einen Bereich von 0 bis MAXGW abgebildet. Eine weitere Möglichkeit, Histogramme zu verändern, ist die so genannte *Histogrammebnung* oder auch *Histogrammäqualisation*, bei der die Verteilung der Grauwerte so verändert wird, dass über eine vorgegebene Anzahl von Grauwerten die tatsächlich vorhandenen Bildpunkte mit ihren zugehörigen Grauwerten gleich verteilt werden.

$x_u \wedge x > x_o | 1, \text{ wenn } x_u \leq x \leq x_o \}$

Dehnung der Grauwertskala

Eine allgemeine Dehnung einer Grauwertskala kann durch folgende Gleichung 4.4 beschrieben werden:

$$A(x, y) = m \cdot E(x, y) + b \quad (4.4)$$

In Abbildung 4.2 ist das allgemeine Prinzip einer Grauwertdehnung grafisch dargestellt. Die Konstante m kann als Kontrast angesehen werden und die Konstante b als Helligkeit. Ist der Kontrast $m > 1$, so wird der Kontrast im Ausgabebild A erhöht. Ist der Kontrast $m < 1$, so wird der Kontrast im Ausgabebild A vermindert. Durch die additive Konstante b wird die Helligkeit des Bildes A beeinflusst. Hiermit kann das gesamte Bild aufgehellt oder verdunkelt werden. Der Kontrast m entspricht der Steigung der Geraden in der Abbildung und die Helligkeit b ist die Grauwertverschiebung. Eine Umsetzung findet dieses Prinzip beispielsweise bei einem herkömmlichen Fernsehgerät, bei dem der Kontrast und auch die Helligkeit durch eine elektronische Widerstandsveränderung auf die Bedürfnisse des jeweiligen Betrachters angepasst werden können.

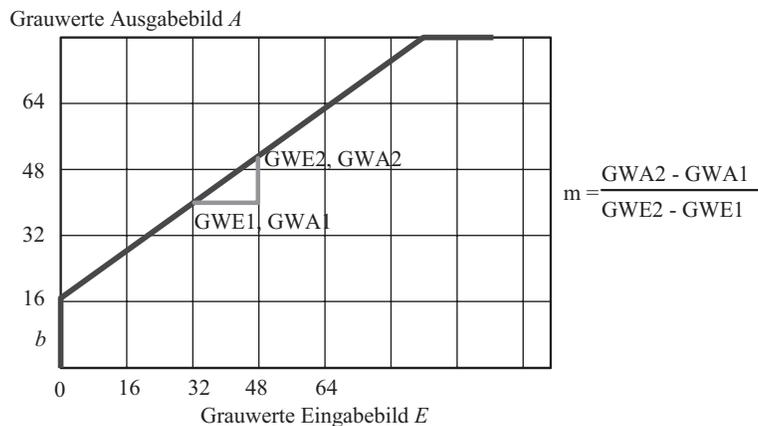


Abbildung 4.2: Allgemeines Prinzip der Grauwertdehnung

Für den diskreten Fall sollte der Grauwertbereich auf den darstellbaren Bereich eingegrenzt werden. Unter der Annahme, dass es sich bei einem Grauwertbild um ein 8 Bit-Bild handelt, ist der gültige Darstellungsbereich das Intervall von 0 bis 255. Dadurch ergibt sich folgende angepasste Gleichung einer allgemeinen Vorschrift zur Dehnung der Grauwertskala:

$$A(x, y) = \begin{cases} 0, & m \cdot E(x, y) + b < 0 \\ 255, & m \cdot E(x, y) + b > 255 \\ m \cdot E(x, y) + b, & \text{sonst} \end{cases} \quad (4.5)$$

Wenn wir die tatsächlichen Grauwerte eines digitalen Bildes betrachten, so stellen wir u.U. fest, dass nicht immer der ganze mögliche Grauwertbereich ausgeschöpft wird. Dies mag durch das zugrunde liegende bildgebende Verfahren verursacht sein.

Das Eingabebild E weist also einen minimalen Grauwert E_{min} auf, der größer Null ist, und einen maximalen Grauwert E_{max} , der kleiner 255 ist. Ein Ausgabebild soll den minimalen Grauwert A_{min} und den maximalen Grauwert A_{max} nach der Transformation aufweisen. In der Abbildung 4.3 ist grafisch das Schema einer solchen Kennlinie dargestellt.

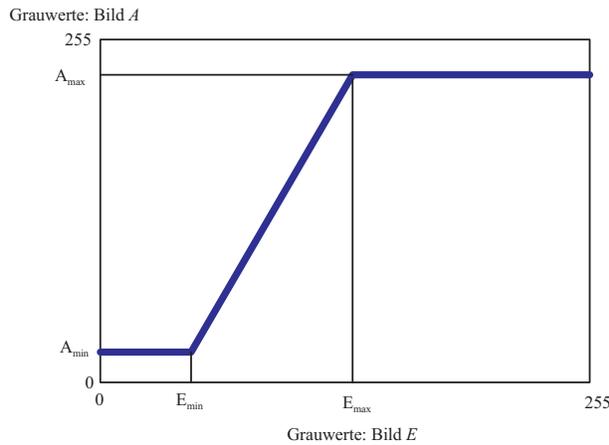


Abbildung 4.3: Beispiel: Kontraststeigerung durch eine Kennlinie

Für den beschriebenen Fall $E_{min} \leq E(x, y) \leq E_{max}$ und $A_{min} \leq A(x, y) \leq A_{max}$ drückt die folgende Beziehung (vgl. Gl. 4.6) eine allgemeine Kontraststeigerung aus:

$$A(x, y) = \frac{A_{max} - A_{min}}{E_{max} - E_{min}} \cdot (E(x, y) - E_{min}) + A_{min} \quad (4.6)$$

Geht man von $A_{min} = 0$ und $A_{max} = 255$ aus, so lässt sich vereinfacht schreiben:

$$A(x, y) = \frac{255}{E_{max} - E_{min}} \cdot (E(x, y) - E_{min}) \quad (4.7)$$

Es kann auch der Fall eintreten, dass der minimale Grauwert E_{min} und der maximale Grauwert E_{max} nur von sehr wenigen Bildpunkten erreicht werden, so dass es durchaus sinnvoll erscheint, diese Grauwerte abzuschneiden durch ein so genannter *clipping*, um eine verstärkte Kontrastanhebung zu erzielen. Dann wird nicht das gesamte Intervall $E_{max} - E_{min}$ benutzt, sondern ein abweichendes $E_{max}^{cl} - E_{min}^{cl}$, mit $E_{min} < E_{min}^{cl} < E_{max}^{cl} < E_{max}$. Diese angepassten Grenzen werden sinnvollerweise so gewählt, dass in den Intervallen $[E_{min} : E_{min}^{cl}]$ bzw. $[E_{max}^{cl} : E_{max}]$ nur eine geringe Anzahl von Bildpunkten liegt. Der Großteil der

Bildpunkte sollte dann schon im Intervall $[E_{max}^{cl} : E_{min}^{cl}]$ liegen. Typischerweise wählt man die neuen Grenzen so, dass sich in dem neuen Intervall 95%–99% der Bildpunkte befinden.

Histogrammebnung

Zusätzlich zu den bisher vorgestellten Skalierungsverfahren in den vorangegangenen Abschnitten existieren eine Reihe weiterer Verfahren zur Histogrammmodifikation. Ein weiteres Verfahren wird im Folgenden vorgestellt. Dieses Verfahren bezieht sich auf die Modifikation von Häufigkeiten, mit denen ein Grauwert im Bild vorkommt. Ein übliches Verfahren ist hier die Histogrammebnung. Bei der Histogrammebnung werden alle Grauwerte so transformiert, dass sie nach der Transformation mit gleicher Häufigkeit $P_A(g)$ im Bild vorkommen (Haberäcker (1985)). Im Ergebnisbild A ist

$$P_A(g) = \frac{1}{N_g}, \text{ für alle } g \in G.$$

Nach Haberäcker (1985) bewirkt eine Histogrammebnung bei natürlichen Bildern eine Verbesserung der Helligkeit und des Kontrastes.

Die Umsetzung der Histogrammebnung nach Rosenfeld und Kak (1982) in ORASIS^{3D} findet sich in Anhang B.2. Demzufolge gilt für eine Histogrammebnung (eine Umsetzung der Histogrammebnung lässt sich im Anhang B.2 finden):

- E das Ursprungsbild mit $N \times M$ Bildpunkten und K Grauwerten z_1, \dots, z_K .
- p_i sei das Vorkommen der Bildpunkte im Bild des Grauwertes z_i .
- Im entsprechenden Histogramm von E sei die Höhe des i -ten Balkens proportional zu p_i .
- $\sum_{i=1}^K p_i = MN$.
- Die Anzahl der Punkte mit dem Grauwert z_i im geebneten Histogramm sei q_i .
- $\sum_{i=1}^K q_i = MN$.

Ein wesentlicher Schritt ist die Transformation des Histogramms p nach q . Für diesen Schritt gilt:

$$\sum_{i=1}^{k_1-1} p_i \leq q_1 \leq \sum_{i=1}^{k_1} p_i.$$

Somit wird allen Bildpunkten in E mit den Grauwerten z_1, \dots, z_{k_1-1} in A' der neue Grauwert z_1 zugeordnet. Ist diese Anzahl kleiner als q_1 , so muss zusätzlichen Punkten mit dem Grauwert z_{k_1} der Grauwert z_1 zugeordnet werden, um

die noch fehlenden Punkte für q_1 zu erhalten. Diese zusätzlichen Punkte können zufällig oder in Abhängigkeit von den Grauwerten benachbarter Punkte gewählt werden.

Für:

$$\sum_{i=1}^{k_2-1} p_i \leq q_1 + q_2 \leq \sum_{i=1}^{k_2} p_i.$$

ergeben sich aus dieser Ungleichung die Punkte, denen der Grauwert z_2 zuzuordnen ist, und zwar:

- Unter der Voraussetzung $k_2 > k_1$ wird allen Punkten mit dem Grauwert z_{k_1} , denen nicht bereits der Grauwert z_1 zugeordnet wurde, der Grauwert z_2 zugewiesen. Falls $k_2 = k_1$, müssen die Punkte mit dem Grauwert z_{k_1} weiter geteilt werden.
- Allen Punkten mit $z_{k_1+1}, \dots, z_{k_2-1}$ als Grauwerten.
- Ist die Anzahl kleiner als q_2 , so muss zusätzlichen Punkten mit dem Grauwert z_{k_2} der Grauwert z_2 zugewiesen werden.

Dementsprechend sind für das resultierende Histogramm die Schritte für die Grauwerte z_3, \dots, z_K fortzusetzen. Es gilt im Allgemeinen für z_h :

$$\sum_{i=1}^{k_h-1} p_i \leq \sum_{i=1}^h q_i \leq \sum_{i=1}^{k_h} p_i.$$

Nach Gool u. a. (1985) stellt eine solche Normierung, bei der die Statistiken erster Ordnung der resultierenden Bilder in jeder Hinsicht übereinstimmen, sicher, dass ein Bild, das eine monotone Transformation eines Originalbildes darstellt, dieselben Resultate bei der Berechnung von Statistiken zweiter Ordnung liefert. Histogramme sind ja gerade Statistiken erster Ordnung (vgl. Kapitel 6).

Nachteilig gegenüber den in den vorangegangenen Abschnitten vorgestellten Verfahren ist, dass zum einen der Berechnungsaufwand größer ist, und zum anderen, dass die Bildinformation verändert wird, je nachdem, wie unterschiedlich die Histogramme des Originalbildes und das geebnete Histogramm sind. Ebenso hat die Strategie, nach der die noch nicht sortierten Bildpunkte ausgewählt werden, Einfluss auf Veränderung.

Weiterhin kann der visuelle Eindruck eines Bildes verfälscht werden, weil die Auswahl der Bildpunkte zufällig war, um die ebene Form des Histogramms zu erreichen. Ein Beispiel für eine solche Verfälschung ist in Abbildung 4.4 zu sehen. Das Histogramm des Schachbrettmusters in Abbildung 4.4 (a) wurde geebnet. Das Ergebnis ist, dass die vorher einfarbigen Flächen nach der Ebnung als Grauwertverläufe (siehe Abbildung 4.4 (b)) hervortreten. Die Grauwertverläufe kommen

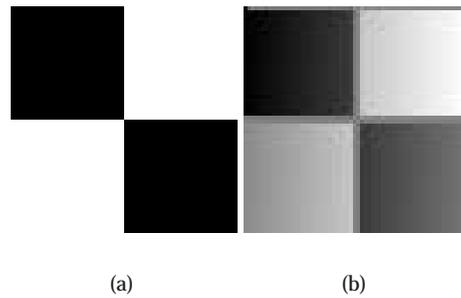


Abbildung 4.4: Histogrammebnung, angewendet auf ein Schachbrettmuster (a). Man erkennt deutlich die Verfälschungen in dem geebneten Bild (b)

durch die Auswahl der zu ändernden Bildpunkte zustande, da hier jeweils immer der erste Punkt gewählt wird. Würden die Punkte zufällig ausgewählt, so entstünde eine Art von Rauschen auf den schwarzen und weißen Flächen.

4.1.3 Shadingkorrektur

Im Allgemeinen geht man davon aus, dass bildgebende Verfahren eine realistische Abbildung des abzubildenden Objektes liefern. Das bedeutet, dass eine gleichmäßig helle Fläche in der realen Welt auch einer entsprechend gleichmäßigen hellen Fläche im digitalen Abbild entspricht. Oftmals ist dem digitalen Bild allerdings ein so genanntes Hintergrundsignal durch das bildgebende Verfahren überlagert. Dieses Signal kann eine Inhomogenität der Beleuchtung verursachen. Sind diese Störeinflüsse bekannt, können sie in einer so genannten *Shadingmatrix* gespeichert werden. Mittels der Shadingmatrix kann dann das jeweilige Bild durch eine Shadingkorrektur (auch: Hintergrundkompensation) verbessert bzw. der Fehleranteil im Signal minimiert werden. Diese Operation kann multiplikativ oder additiv erfolgen. Hierzu wird das Bild, respektive die Bildmatrix, $E(x, y)$ mit der entsprechenden Shadingmatrix verknüpft. Die Shadingmatrix ergibt sich aus einer Aufnahme eines so genannten Weißbildes $E_{white}(x, y)$, mit maximal geöffneter Blende sowie mit maximaler Beleuchtung. Der mittlere Weißwert \overline{W} wird bestimmt. Des Weiteren wird das Gegenstück zum Weißbild, das so genannte Schwarzbild $E_{black}(x, y)$ (bei geschlossener Blende und ohne Beleuchtung), bestimmt sowie der mittlere Schwarzwert \overline{S} berechnet. Das Weißbild beinhaltet allgemeines Rauschen, den Ausleuchtungsfehler und auch Inhomogenitäten in den photo-elektrischen Schichten des bildgebenden Systems. Das Schwarzbild beinhaltet ein gewisses Widerstandsrauschen. Die additive Shadingkorrektur $E_A(x, y)$ wird durch die folgende Beziehung bestimmt:

$$E_A(x, y) = E(x, y) - E_{white}(x, y) + \overline{W} - \overline{S} \quad (4.8)$$

Die multiplikative Shadingkorrektur $E_M(x, y)$ wird durch die folgende Beziehung bestimmt:

$$E_M(x, y) = E(x, y) \frac{\overline{W}}{E_{white}(x, y)} - \overline{S} \quad (4.9)$$

Die Hintergrundkompensation erlaubt die globale Behandlung aller Hintergrundfehler. Die Shadingmatrix selbst kann durch Glättungsoperatoren (vgl. Abschnitt 4.2.2) von Störpartikeln (Rauschpixel) befreit werden.

4.2 Rauschunterdrückung

Rauschen ist ein in der Regel dem Bild überlagertes Signal, das das Bild verfälscht. Es treten sozusagen Unebenheiten in den Intensitätswerten oder auch hohe Frequenzen im Bild auf. Diese Unebenheiten sollen Rauschunterdrückungsverfahren (auch: *Glättungsverfahren*) vermindern oder gar beseitigen. Das ideale Ziel ist, genau diese Frequenzen oder Unebenheiten zu beseitigen, ohne dass das restliche Signal – sprich: Bild – von dieser Glättung beeinflusst wird. Es existieren lineare und nichtlineare Verfahren, die diese Forderung z.T. erfüllen.

4.2.1 Morphologische Operationen

In Kapitel 3 haben wir schon eine einfache Art der Glättung eines Binärbildes kennen gelernt. Durch die morphologische Operation² *Erosion* können – je nach Wahl des verwendeten strukturierenden Elementes – einzelne Bildpunkte gelöscht werden. Diese einzelnen Bildpunkte sind im Allgemeinen Rauschpunkte, die zur Weiterverarbeitung – der Segmentierung, der Merkmalsextraktion und Klassifikation – nicht notwendig bzw. eher störend sind.

Ebenso haben die morphologischen Operationen *Öffnen* oder *Schließen* Einfluss auf einzelne Bildpunkte. Auch hier können einzelne Bildpunkte aus dem Bild gelöscht werden, da ja auch hier die Erosion als Grundoperation angewendet wird.

4.2.2 Filterbasierte Rauschminderung

Die filterbasierte Glättung lässt sich zu den lokalen Bildpunktoperationen zählen, die in der Regel auf Grauwertbilder berechnet werden. Grundlage dieser Operationen ist die diskrete *Faltung* von digitalen Bildern (im zweidimensionalen Fall). Zunächst wollen wir uns also mit der diskreten Faltung beschäftigen, um dann

² Schon bei den simplen Operationen auf Punktgruppen (vgl. Abschnitt 3.1) wird durch die Operation *Shrink* eine Möglichkeit zur Glättung in Binärbildern gegeben.

4	5	5	4	3	4
5	5	3	7	7	7
4	3	2	6	6	8
9	9	8	8	9	9
...					

(a) E

1	1	1
1	1	1
1	1	1

(b) F_M

	4	5	5	6	
	6	6	6	7	

(c) A

Abbildung 4.5: Diskrete Faltung: Das Eingabebild E (a) wird mit der Filtermaske F_M (b) gefaltet. Im Ausgabebild A (c) sind gerundete Werte dargestellt, da die jeweilige Summe der einzelnen Bildpunkte mit $\frac{1}{9}$ multipliziert werden muss, damit der berechnete Wert im Ausgabebild im darstellbaren Bereich $[0...255]$ bleibt

übliche Glättungsoperatoren kennen zu lernen und deren Funktionsweise zu verstehen.

Wir wollen uns das Prinzip der *diskreten Faltung* mittels einer 3×3 Filtermaske F_M verdeutlichen³ (man kann sich diese Maske auch als Blende vorstellen), die über ein Eingabebild E „geschoben“ wird und damit die gewichtete Summe innerhalb dieser Blende für das Ausgabebild A berechnet. In der folgenden Abbildung 4.5 ist ein Beispiel für eine diskrete Faltung dargestellt:

In analytischer Form bedeutet das für Bildpunkt A_{22} des Ausgabebildes A :

$$A_{22} = \begin{matrix} F_{M_{11}} \cdot E_{11} & + F_{M_{12}} \cdot E_{12} & + F_{M_{13}} \cdot E_{13} \\ F_{M_{21}} \cdot E_{21} & + F_{M_{22}} \cdot E_{22} & + F_{M_{23}} \cdot E_{23} \\ F_{M_{31}} \cdot E_{31} & + F_{M_{32}} \cdot E_{32} & + F_{M_{33}} \cdot E_{33} \end{matrix} \quad (4.10)$$

Ausgedrückt in Zahlen (für das Beispiel in Abbildung 4.5):

$$A_{22} = \frac{1}{9} \begin{pmatrix} 1 \cdot 4 & + 1 \cdot 5 & + 1 \cdot 5 \\ 1 \cdot 5 & + 1 \cdot 5 & + 1 \cdot 3 \\ 1 \cdot 4 & + 1 \cdot 3 & + 1 \cdot 2 \end{pmatrix} = 4 \quad (4.11)$$

Entsprechend der Vorschrift aus Gleichung 4.10 müssen nun alle weiteren Bildpunkte berechnet werden. Die übliche Schreibweise für eine diskrete Faltung lautet:

$$A = F_M ** E$$

mit dem Operator $**$ für die zweidimensionale diskrete Faltung.

Die allgemeine Form der diskreten Faltung eines Bildes E mit einer Filtermaske

³ Für dieses Beispiel wird das Randproblem – s.u. – vernachlässigt.

(oder auch Faltungsmaske) F_M im zweidimensionalen Fall lautet:

$$A(x, y) = k_1 \sum_{i=-k}^{+k} \sum_{j=-l}^{+l} F_M(i, j) E(x + i, y + j) + k_0 \quad (4.12)$$

k_0 und k_1 sind Konstanten, die zur Skalierung auf den erlaubten Intensitätswertbereich (Grauwertbereich) benötigt werden (vgl. Skalierungsfaktor $\frac{1}{9}$ in Gleichung 4.11). $F_M(i, j)$ sind die Koeffizienten der Faltungsmaske. I.d.R. hat die Maske $(2k + 1)$ Spalten und $(2l + 1)$ Zeilen. Das bedeutet, dass in der Praxis meistens quadratische Masken der Größen 3×3 , 5×5 , ... verwendet werden, da in diesem Fall der *Bezugspunkt* der Maske zentral ist.

Ein paar Worte noch zu dem hier auftretenden so genannten Randproblem. Die lokale Nachbarschaft für die Randpunkte ist nicht definiert. In unserem Beispiel ist der Rand lediglich ein Pixel breit. Das ist im Allgemeinen nicht sehr tragisch. Werden die Faltungsmasken allerdings größer, so bleiben u.U. sehr viele Bildpunkte des Randes unberücksichtigt. Die folgenden Möglichkeiten, das Randproblem anzugehen, sind nur eine Auswahl der üblichsten Alternativen. Eventuell sind andere Möglichkeiten für ein spezielles Problem sinnvoller als die hier genannten:

- Die nicht berücksichtigten Randpunkte werden einfach weggelassen. Das bedeutet, dass bei einer Maskengröße von $(2k + 1) \times (2k + 1)$ ein k breiter Rand weggelassen wird. Wird evtl. iterativ gefaltet, so kann das Bild sehr schnell sehr klein werden ...
- Die entsprechenden Randpunkte des Eingabebildes (oder bei Iteration des Vorgängerbildes) werden kopiert.
- Das Bild wird periodisch fortgesetzt. Diese Möglichkeit ist unter systemtheoretischen Gesichtspunkten die optimale (vgl. Wahl (1989)).

Nachdem wir uns die diskrete Faltung angesehen haben, kommen wir im Folgenden zu den drei wohl bekanntesten Glättungsoperatoren. Zunächst wird der klassische Mittelwertfilter besprochen. Im Anschluss wird der Medianfilter als ein nichtlinearer Filter vorgestellt, und zum Schluss eine „Erweiterung“ des Mittelwertfilters, der Gaußfilter.

Mittelwertfilter

Der *Mittelwertfilter* oder auch *Average-Filter* (auch Average-Operator) berechnet den mittleren Intensitätswert innerhalb einer lokalen Nachbarschaft, die durch die Größe des Filters bestimmt wird. Der so ermittelte „neue“ Intensitätswert wird im Ausgabebild dem *Bezugspunkt* zugewiesen. Dadurch wird eine Glättung der Unebenheiten in den Grauwerten erzielt. Der Mittelwertfilter wird auch manchmal als bewegter Mittelwertfilter bezeichnet. Für eine Umsetzung des Mittelwertfilters in ORASIS^{3D} siehe Anhang B.3.

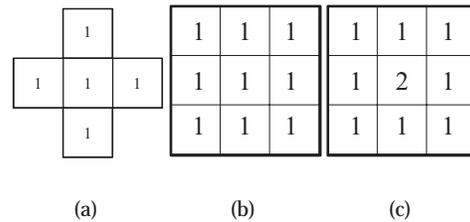


Abbildung 4.6: Verschiedene Mittelwertfilter: (a) Mittelwertfilter bzgl. einer 4er Nachbarschaft, (b) Mittelwertfilter bzgl. einer 8er Nachbarschaft und (c) gewichteter Mittelwertfilter bzgl. einer 8er Nachbarschaft. Die Bezugspunkte sind jeweils in der Mitte

In Abbildung 4.6 (a) ist ein Mittelwertfilter bzgl. einer 4er Nachbarschaft im Rechteckraster dargestellt. Der Skalierungsfaktor für diese Maske beträgt $\frac{1}{5}$. Der Mittelwertfilter ist ein homogener, lokaler und linearer Filter. Im Frequenzbereich entspricht der Mittelwertfilter einem so genannten *Tiefpass* (s. auch Kapitel 11), der nur die niedrigen Frequenzen durchlässt. Oder andersherum, der Sperrbereich des Filters sind die hohen Frequenzen.

Eine andere Form des Mittelwertfilters, die wohl am häufigsten verwendet wird, ist in Abbildung 4.6 (b) bzgl. einer 8er Nachbarschaft im Rechteckraster dargestellt. Der Skalierungsfaktor beträgt für diese Filtermaske $\frac{1}{9}$. Diese Filtermaske wird auch als ungewichtet bezeichnet, da alle Koeffizienten gleich gewichtet sind.

Von einem gewichteten Mittelwertfilter spricht man dann, wenn der mittlere Punkt in der Filtermaske (auch *Bezugspunkt*) beispielsweise – wie in Abbildung 4.6 (c) dargestellt ist – doppelt gewichtet wird. Der Skalierungsfaktor beträgt für diese Maske $\frac{1}{10}$.

Wird ein Mittelwertfilter angewendet, so werden nicht nur die Grauwertunebenheiten sondern auch die Diskontinuitäten im Grauwertsignal (Kantenübergänge) geglättet. Dies bedeutet, dass auch Kanten durch eine Mittelwertglättung an ihrer ursprünglichen Steilheit verlieren. Die Kantenübergänge werden also nicht nur verändert, sondern verfälscht.

In der folgenden Abbildung 4.7 ist ein weiteres Beispiel für eine Mittelwertfilterung und somit eine Glättung des Bildes zu sehen. Man kann deutlich erkennen, dass im Ausgabebild der störende Punkt im Eingabebild mit dem Intensitätswert „13“ einen erheblichen Einfluss auf seine Nachbarpunkte hat. Durch die Faltung mit einem Mittelwertfilter heben sich die Intensitätswerte in der Nachbarschaft an.

Medianfilter

Ein Glättungsoperator, der die Steilheit der Kanten in der Regel nicht beeinflusst, ist der nichtlineare Medianfilter (vgl. Anhang B.4). Dieser Filter ist ein *Rangfolgen-*

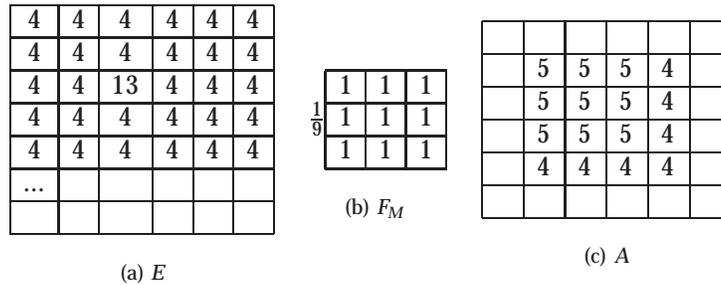


Abbildung 4.7: Glättung von Bild E (a) durch Mittelwertfilter (b). Deutlich wird der Einfluss eines einzelnen Störpixels auf seine Nachbarn im Ausgabebild A (c)

filter (auch: *RangordnungsfILTER*), der für eine bestimmte Nachbarschaft ($3 \times 3, 4 \times 5, \dots$) die Bildpunkte nach der Größe des Intensitätswertes sortiert.⁴ Der mittlere Wert aus der sortierten Liste der Intensitätswerte ist dann der neue Intensitätswert für das geglättete Ausgabebild. Beispielsweise ist für eine 3×3 -Nachbarschaft der mittlere Wert an der Stelle 5 zu finden. Für eine 5×5 -Nachbarschaft ist der mittlere Wert an der Stelle 13 zu finden, etc.

Als Beispiel sollen uns die Bilder aus Abbildung 4.5 dienen. Wird ein 3×3 -Medianfilter über das Eingabebild geschoben, so ergibt sich im Ausgabebild an der Stelle A_{32} eine 5 statt einer 6, die beim Mittelwertfilter berechnet wurde. Interessanterweise ist die erste berechnete Zeile im Ausgabebild für Median- und Mittelwertfilter gleich.

Ein weiteres Beispiel zur Verdeutlichung soll das Bild aus Abbildung 4.8 sein. Geht man vom gleichen Eingabebild aus Abbildung 4.8 (a) aus und verwendet dieses Mal den Medianfilter, so ist das Ergebnis im Ausgabebild optimal. Das Störpixel mit dem Intensitätswert „13“ ist herausgefiltert, ohne Einfluss auf die Nachbarschaft zu haben. Dieser Sachverhalt ist in Abbildung 4.8 nochmals bildhaft wiedergegeben. Man kann deutlich die Unterschiede zu den Ergebnissen in Abbildung 4.7 erkennen.

Gaußfilter

Der Gaußfilter ist im Prinzip nur eine Erweiterung des gewichteten Mittelwertfilters. Das Zentrum wird analog zur Gauß-Glocken-Funktion stärker gewichtet als der Rand. Eine typische 3×3 -Gaußfilter-Maske ist in Abbildung 4.9 dargestellt. Der Mittelpunkt und die direkten Nachbarn werden stärker gewichtet als die Nachbarn, die $\sqrt{2}$ -Pixel vom Mittelpunkt entfernt sind – also die diagonalen

⁴ Bei größeren Filtermasken (so ab 11×11 Bildpunkten) ist es ratsam, sich für die Implementierung des Sortieralgorithmus ein paar Gedanken zu machen. Ein *Bubble-Sort*-Algorithmus ist für größere Filtermasken nur bedingt einsetzbar. Für Sortierverfahren siehe zum Beispiel Sedgewick (1992) oder auch Wirth (1983)

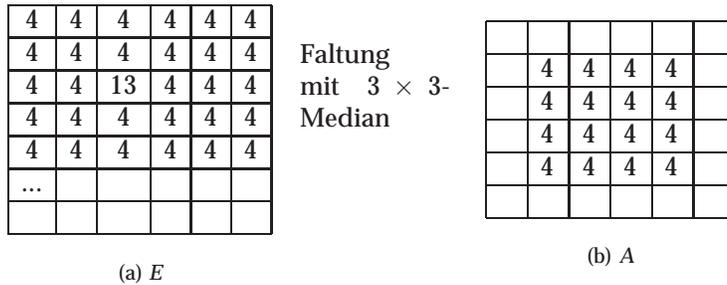


Abbildung 4.8: Glättung durch Medianfilter und (der nicht vorhandene) Einfluss eines einzelnen Störpixels auf seine Nachbarn

Nachbarn. Der Skalierungsfaktor beträgt $\frac{1}{16}$.

1	2	1
2	4	2
1	2	1

Abbildung 4.9: Gaußfilter

In der folgenden Abbildung 4.10 ist das Eingabebild aus den Abbildungen 4.7 bzw. 4.8 mit dem Gaußfilter aus der Abbildung 4.9 gefiltert worden. Im Ausgabebild stehen mit Absicht die Nachkommastellen – obwohl diese für eine Visualisierung auf den nächsten Intensitätswert (in diesem Fall „5“) gerundet würden –, da dies die verbesserte Glättung gegenüber einem einfachen Mittelwertfilter verdeutlicht. Die Berechnungsergebnisse für die Intensitätswerte der einzelnen Pixel liegen bei ca. 4,5625. Das ist gegenüber 5 bei der Mittelwertberechnung schon deutlicher am Ursprungswert 4.

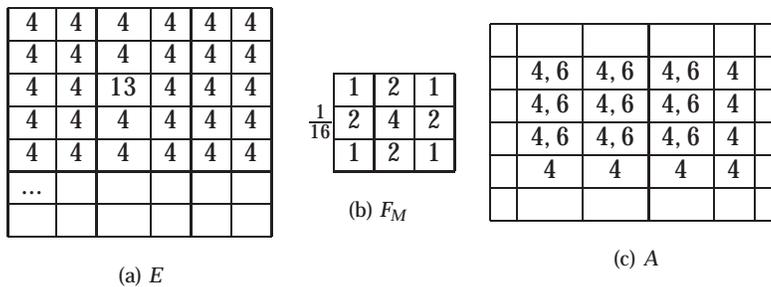


Abbildung 4.10: Glättung durch Gaußfilter und Einfluss eines einzelnen Störpixels auf seine Nachbarn

Zum Abschluss dieses Abschnittes nochmals ein Beispiel für eine Glättung mittels eines Medianfilters auf einem durch Salz-und-Pfeffer-Rauschen gestörten Bild. In

Abbildung 4.11(a) ist ein durch Rauschen gestörtes Bild mit einem zugehörigen Zeilenprofil zu sehen. Die Abbildung 4.11(b) zeigt das mit einem 5×5 -Median geglättete Bild ebenfalls mit dem zugehörigen Zeilenprofil der entsprechenden Zeile. Es sind sowohl im Bild als auch im jeweiligen Zeilenprofil die Störung und die Beseitigung der Störung zu erkennen.

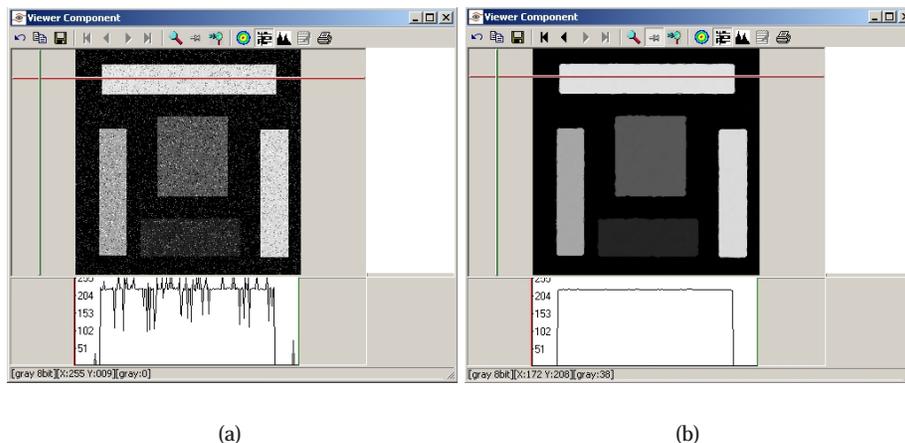


Abbildung 4.11: Glättung durch Medianfilter: (a) gestörtes Bild mit Zeilenprofil und (b) geglättetes Bild mit korrespondierendem Zeilenprofil. Screenshots der ORASIS^{3D}-Oberfläche. Diese Abbildung befindet sich auch auf der beiliegenden CD-ROM im Verzeichnis Abbildungen

4.3 Geometrische Entzerrung

Die bisher vorgestellten Vorverarbeitungsoperationen bezogen sich lediglich auf den Intensitätswert (sprich: Grauwert) eines Bildpunktes, doch auf dessen Ortskoordinaten. Die im Folgenden beschriebenen einfachen, so genannten affinen Transformationen transformieren in der Regel die Ortskoordinaten eines Bildpunktes. Beispielsweise muss bei einem Vergleich von Bildern, die unterschiedliche Größen aufweisen, eine Art Größennormierung (Skalierung) durchgeführt werden. Ebenso kann es notwendig sein, Verzerrungen, die durch das bildgebende Verfahren entstanden sind, zu beseitigen.

Man unterscheidet prinzipiell drei geometrische Transformationen, die im Folgenden näher erläutert werden:

- Translation: Verschiebung von Bildern
- Rotation: Drehung von Bildern und
- Skalierung: Vergrößerung bzw. Verkleinerung von Bildern.

Translatorische Transformationen und auch Drehungen um $n \cdot 90^\circ$ mit $n = 0, 1, 2, \dots$ sind relativ einfache Operationen. Die Topologie der Bildpunkte unter Berücksichtigung des Rechteckrasters wird nicht beeinflusst. Sollen Drehungen um beliebige Winkel durchgeführt werden, sowie beliebige Skalierungen, so werden Interpolationsmechanismen benötigt. Diese Interpolationsverfahren haben die Aufgabe, die Topologie der Bildpunkte und ihrer Grauwerte möglichst gut zu erhalten. Beispielsweise kann auf den nächsten Wert bzgl. einer Nachbarschaft gerundet werden. Oder der zu berechnende Wert kann durch eine *bilineare Interpolation* bestimmt werden. Die bilineare Interpolation bestimmt den neuen Grauwert $A(x^{neu}, y^{neu})$ durch eine gewichtete Summe des gerade betrachteten Grauwertes $E(x, y)$ mit drei seiner Nachbarn:

$$A(x^{neu}, y^{neu}) = w_1 E(x, y) + w_2 E(x, y + 1) + w_3 E(x + 1, y) + w_4 E(x + 1, y + 1) \quad (4.13)$$

Durch die Wahl der w_i kann die Position des neuen Bildpunktes in Hinblick auf seine Nachbarn berücksichtigt werden. Da hierbei u.U. neue Grauwerte entstehen, lässt sich die bilineare Interpolation nicht auf logische (binäre) Bilder anwenden. Werden die w_i so gewählt, dass $\sum_{i=1}^4 w_i = 1$ ist, dann wird eine gewichtete und gemittelte Summe berechnet.

Im zweidimensionalen Raum existieren prinzipiell die drei oben genannten Transformationen – Translation (auch: Versetzung des Ursprungs), Rotation und Skalierung. Weitere Veränderungen können durch Kombination dieser drei Transformationen erzielt werden. Bei diesen Transformationen handelt es sich um Bildpunkttransformationen, sodass jeder Bildpunkt einzeln transformiert werden muss. Werden Matrizen für diese Transformationen genutzt, so lassen sich die Transformationen effektiv umsetzen. In der Regel werden 3×3 -Matrizen für diese zweidimensionalen Transformationen verwendet. Wir benötigen für eine Koordinatentransformation im allgemeinen Fall eine Matrix M und einen Spaltenvektor S mit:

$$M = \begin{pmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{pmatrix} \quad \text{und} \quad S = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix}$$

Für eine Koordinatentransformation gilt: Ein Bildpunkt $E(x, y)$ habe ursprünglich die Koordinaten x, y bezogen auf ein bestimmtes Koordinatensystem. Nach der Transformation habe der Bildpunkt E^{neu} die Koordinaten x^{neu}, y^{neu} bezogen auf ein verändertes Koordinatensystem. Das veränderte Koordinatensystem ergibt sich aus der Anwendung der affinen Transformationen. Die Koordinaten des ursprünglichen sowie des berechneten Bildpunktes lassen sich in homogener Koordinaten-Schreibweise als Spaltenvektoren bestimmen.

$$E = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad \text{und} \quad E^{neu} = \begin{pmatrix} x^{neu} \\ y^{neu} \\ 1 \end{pmatrix}$$

Eine affine Transformation berechnet die neuen Koordinaten als eine lineare Kombination der ursprünglichen Werte unter Berücksichtigung der Koeffizienten der entsprechenden Transformationsmatrix. Allgemein entstehen dadurch die folgenden linearen Gleichungen

$$\begin{aligned} x^{neu} &= M_{11} \cdot x + M_{12} \cdot y + M_{13} \\ y^{neu} &= M_{21} \cdot x + M_{22} \cdot y + M_{23} \\ 1 &= M_{31} \cdot x + M_{32} \cdot y + M_{33} \end{aligned} \quad (4.14)$$

Die letzte Zeile ist nur dann erfüllt, wenn $M_{31} = 0$, $M_{32} = 0$ und $M_{33} = 1$. Ein neues transformiertes Koordinatenpaar (in Spaltenform) wird demnach durch die Multiplikation einer Matrix M mit einem Spaltenvektor S berechnet. Also kann die Koordinatentransformation E^{neu} wie folgt definiert werden:

$$E^{neu} = M \cdot S$$

4.3.1 Translation

Eine Translation T kann als Verschiebung des Ursprungs verstanden werden. Die Achsenrichtung und der Achsenverlauf werden nicht beeinflusst. Der Verlauf bedeutet hier, dass die Skalierung der Achsen unverändert bleibt. Der Ursprung hat im neuen Koordinatensystem die Koordinaten $(-u_x^{neu}, -u_y^{neu})$. Neue Punkte werden wie folgt berechnet:

$$\begin{aligned} x^{neu} &= 1 \cdot x + 0 \cdot y - u_x^{neu} \\ y^{neu} &= 0 \cdot x + 1 \cdot y - u_y^{neu} \end{aligned}$$

Die Transformationsmatrix lautet:

$$T = \begin{pmatrix} 1 & 0 & -u_x^{neu} \\ 0 & 1 & -u_y^{neu} \\ 0 & 0 & 1 \end{pmatrix}$$

4.3.2 Rotation

Bei einer Rotation R werden weder der Ursprung noch die Skalierung der Achsen verändert. Lediglich die Richtung der Achsen wird verändert. Eine allgemein gültige Konvention ist, dass die Richtung der Achsen durch eine Drehung um den Winkel Φ entgegen dem Uhrzeigersinn durchgeführt wird. Die neuen Koordina-

ten berechnet man durch:

$$\begin{aligned}x^{neu} &= \cos \Phi \cdot x + \sin \Phi \cdot y + 0 \\y^{neu} &= -\sin \Phi \cdot x + \cos \Phi \cdot y + 0\end{aligned}$$

Die Transformationsmatrix lautet:

$$R = \begin{pmatrix} \cos \Phi & \sin \Phi & 0 \\ -\sin \Phi & \cos \Phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

4.3.3 Skalierung

Für die Skalierung S gilt: Der Ursprung und die Achsenrichtung werden durch die Transformation nicht verändert. Lediglich die Skalierung der Achsen verändert sich. Die Skalierungsfaktoren $scal_x$ und $scal_y$ bestimmen die Skalierung. Sind die Werte größer als Eins, so vergrößert sich das gesamte Koordinatensystem. Sind die Werte kleiner als Eins, so verkleinert sich das Koordinatensystem. Die neuen Koordinaten werden berechnet durch:

$$\begin{aligned}x^{neu} &= scal_x \cdot x + 0 \cdot y + 0 \\y^{neu} &= 0 \cdot x + scal_y \cdot y + 0\end{aligned}$$

Die Transformationsmatrix lautet:

$$R = \begin{pmatrix} scal_x & 0 & 0 \\ 0 & scal_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

4.3.4 Kombinierte Transformationen

Wie bereits oben erwähnt wurde, können komplexere Ent- bzw. Verzerrungen durch Kombinationen der drei Grundtransformationen erreicht werden. Aufgrund der Matrizendarstellung besteht die einfache Möglichkeit, die Grundtransformationen zu kombinieren. Beispielsweise lassen sich eine Translation T mit einer Rotation R hintereinander ausführen. Durch eine Matrixmultiplikation erhält man die entsprechende Kombinationsmatrix K , die beide Transformationen enthält:

$$K_{TR} = T \cdot R$$

Soll die kombinierte Transformation wieder rückgängig gemacht werden, so ist darauf zu achten, dass die beiden inversen Grundtransformationen in umgekehrter Reihenfolge durchzuführen sind:

$$K_{TR}^{-1} = T^{-1} \times R^{-1}$$

Im Allgemeinen gilt, dass für eine komplexe Transformation zunächst die entsprechende Kombinationsmatrix berechnet und mit dieser dann im Anschluss das Bild transformiert wird.

Die neuen Koordinaten werden in der Regel durch lineare Polynome ermittelt (vgl. Gleichung 4.14). Die Parameter M_{ij} mit $i, j = 1, 2, 3$ (für den 2D-Fall) können beispielsweise durch so genannte Systemkorrekturen bestimmt werden. Zum Beispiel werden die entsprechenden Daten für eine Systemkorrektur von einem Fotokameraobjektivhersteller mitgeliefert, sodass die geometrischen Verzerrungen bei einem Weitwinkelobjektiv „herausgerechnet“ werden können. Sind die Daten für eine Systemkorrektur nicht bekannt, so können die Werte der Parameter M_{ij} durch die so genannte *Passpunktmethode* (s.a. Haberäcker (1985)) bestimmt werden, die in bestimmten Grenzen eine Systemkorrektur ermöglicht. Für die Passpunktmethode wird folgende Annahme getroffen: $S = (A(x, y))$ sei ein verzerrtes Bild. $R = (E(u, v))$ sei ein Referenzbild. S soll an R durch geeignete geometrische Transformationen (auch Kombination) angeglichen werden. Des Weiteren ist über den so genannten funktionalen Zusammenhang der Verzerrung nichts bekannt. Er kann allerdings durch ein Polynom n -ten Grades approximiert werden. $S' = (A'(x', y'))$ sei das zu berechnende entzerrte Bild, das dann im Idealfall kongruent zu R ist.

Zur Vereinfachung wird ein linearer Ansatz angenommen (vgl. Gleichung 4.14). Damit müssen sechs Parameter M_{11} bis M_{23} bestimmt werden. Dafür werden insgesamt drei Passpunkte benötigt. Passpunkte sind dabei sich entsprechende Punkte in R und in S , die eindeutig ermittelt werden können. Dies können zum Beispiel Eckpunkte von Objekten sein, die sowohl in R als auch in S eindeutig bestimmbar sind. Drei Passpunkte liefern drei Koordinatenpaare, die einander entsprechen. Für den i -ten Passpunkt gilt:

$$(u_i, v_i) \hat{=} (x_i, y_i)$$

Daraus resultieren die beiden folgenden Gleichungen:

$$u_i = M_{12} \cdot x_i + M_{12} \cdot y_i + M_{13}$$

und

$$v_i = M_{21} \cdot x_i + M_{22} \cdot y_i + M_{23}$$

mit $i = 1, 2, 3$ oder in Matrixschreibweise

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} \cdot \begin{pmatrix} M_{11} \\ M_{12} \\ M_{13} \end{pmatrix}$$

entsprechend für v

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} \cdot \begin{pmatrix} M_{21} \\ M_{22} \\ M_{23} \end{pmatrix}$$

Aus diesen Gleichungssystemen können nun die sechs Unbekannten (Polynomparameter) ermittelt werden. Im Anschluss daran kann dann S' aufgebaut werden.

4.4 Zusammenfassung

Wir haben in diesem Kapitel einige Methoden der Bildvorverarbeitung zum Zwecke der allgemeinen Bildverbesserung besprochen. Zunächst haben wir in Abschnitt 4.1 einfache Möglichkeiten zur Kontrastverbesserung durch so genannte *Look-Up-Tables* (LUT) kennen gelernt, die bereits während der Bildgebung zur Anwendung kommen können. Ist das Bild erst aufgenommen und liegt es digital vor, können wir mit Histogrammanipulationen – wie zum Beispiel mit der Histogrammebnung – den allgemeinen Kontrast verbessern. Allerdings haben wir – wie am Beispiel mit dem Schachbrettmuster – auch gesehen, dass eine Histogrammebnung nicht immer sinnvoll ist. Eine Histogrammebnung lässt sich mit besseren Resultaten auf eher natürliche Szenen anwenden.

Der zweite große Punkt, den wir in Abschnitt 4.2 behandelt haben, ist die Rauschminderung oder Glättung eines Eingabebildes. Rauschen kann vielfältige Ursachen haben und dadurch auch sehr unterschiedlicher Natur sein. Im Wesentlichen unterscheiden wir Gauß- von Salz-und-Pfeffer-Rauschen. Insbesondere Letzteres lässt sich ohne große Verfälschung des Eingabebildes durch einen Medianfilter vermindern. Die Mittelwertfilter hingegen glätten nicht nur das Rauschen bezogen auf eine bestimmte Nachbarschaft, zusätzlich schwächen sie auch die Kantenübergänge (Diskontinuitäten) im Bild ab. Das ist im Allgemeinen eine eher unerwünschte Eigenschaft.

Der letzte Punkt in Abschnitt 4.3 behandelt geometrische Entzerrungen durch die so genannten affinen Transformationen. Es ist beispielsweise in manchen Anwendungen sinnvoll, Objekte, die verglichen werden sollen, auf eine einheitliche Größe zu skalieren, damit man eine größeninvariante Vergleichsmöglichkeit hat. Der gleiche Sachverhalt kommt auch bei verdrehten und/oder verschobenen Objekten zum Tragen. Die affinen Transformationen finden sehr häufig Anwendung in der Computergrafik (s. Foley u. a. (1993)).