

6 Bildbearbeitung für Rasterbilder

Die Bildbearbeitung ist häufig der erste Schritt auf dem Wege zu einer umfassenden interaktiven oder automatischen Bildinterpretation. Dabei werden mit den Mitteln der Bildbearbeitung die zu analysierenden Bilder geometrisch und photometrisch manipuliert, um sie beispielsweise in ihrer Größe zu ändern, Helligkeiten oder Farben und Kontraste zu beeinflussen, Störungen zu eliminieren und bestimmte Sachverhalte (z.B. Kanten) hervorzuheben. Das Ergebnis einer Bildbearbeitung ist stets wieder ein Bild.

Darüber hinaus wird die Bildbearbeitung häufig auch ohne nachfolgende Auswerteschritte eingesetzt, um Bilder zu Präsentationszwecken (z.B. Bildersammlungen, Plakate, Web-Graphiken usw.) zuzuschneiden, zu überlagern bzw. zu verschmelzen, zu verbessern, zu retuschieren oder zu verfremden. Diese Arbeitsschritte, die in den Handbüchern der gängigen Bildbearbeitungsprogramme (Paint Shop Pro, Photoshop, Gimp usw.) meist ausführlich beschrieben sind, werden hier nicht betrachtet.

Im folgenden werden nur solche grundlegenden Bildbearbeitungs-Methoden vorgestellt, die sich auch als Vorstufe zu einer Bildinterpretation einsetzen lassen. Dabei werden die Methoden nur am Beispiel von Grauwert- oder Schwarz-Weiß-Bildern behandelt. (Die Übertragung auf Farbbilder ist problemlos möglich, indem man je nach Anwendung beispielsweise alle Farbkanäle des RGB-Farbmodells oder aber z.B. nur den Helligkeits-Kanal eines anderen Farbmodells (siehe Kapitel 3.2.2.2) bearbeitet). Die Grauwerte der Bilder sollen zwischen $g_{\text{MIN}} = 0$ (schwarz) und g_{MAX} (weiß; z.B. $g_{\text{MAX}} = 255$) liegen.

6.1 Photometrische Bildmanipulationen

Unter **photometrischen Bildmanipulationen** versteht man Verarbeitungsschritte, die die Helligkeit oder den Kontrast von Bildern ändern. Unter **Kontrast** versteht man dabei die Differenz zwischen dem minimalen Grauwert g_{min} und dem maximalen Grauwert g_{max} in einem Bildausschnitt bzw. dem gesamten Bild. Um ein objektives Kontrastmaß zu erhalten, kann diese Differenz noch zu dem Intervall der zulässigen Grauwerte $[g_{\text{MIN}}=0, g_{\text{MAX}}]$ in Beziehung gesetzt werden:

$$\text{Kontrast} = \frac{g_{\text{max}} - g_{\text{min}}}{g_{\text{MAX}} - g_{\text{MIN}}} = \frac{g_{\text{max}} - g_{\text{min}}}{g_{\text{MAX}}} \quad (\text{es gilt: Kontrast} \in [0, 1]). \quad (6-1)$$

Eine gute Möglichkeit zur Beurteilung von Kontrast und Helligkeit eines Bildes bietet das sogenannte **Histogramm**. Das Histogramm der absoluten Häufigkeiten ist eine Tabelle, die zu jedem Grauwert g angibt, mit welcher absoluten Häufigkeit $h(g)$ er im Bild auftritt. Wenn man die jeweiligen Pixelhäufigkeiten durch die absolute Auflösung, d.h. die Gesamtzahl der Pixel des Bildes (= Spaltenzahl * Zeilenzahl) dividiert, so erhält man die relativen Häufigkeiten $h^{\#}(g)$; für diese gilt: $0 \leq h^{\#}(g) \leq 1$ für alle Grauwerte g und $\sum h^{\#}(g) = 1$, wobei die Summe über alle Grauwerte läuft.

Ein Histogramm läßt sich auch als Bild darstellen; siehe z.B. mittlere Spalte von Abbildung 6.1. Aus dem Histogramm in Abbildung 6.1 (b) ist zu ersehen, dass das Bild 6.1 (a) von mittlerer Helligkeit und mäßigem Kontrast ist.

Ein weiteres wichtiges Maß zur Beurteilung von Bildern ist die **Entropie** H , die den mittleren Informationsgehalt eines Bildes angibt:

$$H = - \sum_{g \in [g_{\text{MIN}}, g_{\text{MAX}}]} h^{\#}(g) * \log_2(h^{\#}(g)). \quad (6-2)$$

Die Entropie eines Bildes gibt an, wieviel Bits zur Abspeicherung eines Bildpunktes erforderlich sind und in welchem Umfang sich der Speicherplatzbedarf eines Bildes mit geeigneten Bildkompressionstechniken (s. Kapitel 7) mindestens reduzieren läßt.

Beispiele: In einem Binärbild soll die eine Hälfte der Bildpunkte den Grauwert g_1 und die andere Hälfte den Grauwert g_2 besitzen. Aus $h^{\#}(g_1) = h^{\#}(g_2) = 0.5$ folgt:

$$H = -0.5 \cdot \log_2(0.5) - 0.5 \cdot \log_2(0.5) = -0.5 \cdot (-1.0) - 0.5 \cdot (-1.0) = 1.0.$$

Falls $\frac{1}{4}$ aller Pixel den Grauwert g_1 und $\frac{3}{4}$ aller Pixel den Grauwert g_2 besitzt, so ergibt sich:

$$H = -0.25 \cdot \log_2(0.25) - 0.75 \cdot \log_2(0.75) = -0.25 \cdot (-2.0) - 0.75 \cdot (-0.42) = 0.82.$$

Zur verlustfreien Speicherung dieser beiden Bilder benötigt man also ein bzw. sogar weniger als ein Bit pro Pixel.

Im folgenden werden einige Möglichkeiten zur globalen Änderung von Bildhelligkeit und -kontrast, sowie zur „Glättung“ der Bildfunktion (Weichzeichnen und Störungselimination) und zur Hervorhebung von Grauwertübergängen angegeben.

6.1.1 Punktoperatoren

Bei den **Punktoperatoren** wird eine Berechnungsvorschrift auf alle Punkte eines Bildes angewendet. Im Gegensatz zu den „lokalen Bildoperatoren“ (siehe Kapitel 6.1.2) hängt das Ergebnis des Operators hier nur von der gewählten Transformation und dem Grauwert des jeweils zu transformierenden Bildpunktes ab. Mit Hilfe von Punktoperatoren können die Helligkeit und der Kontrast eines Bildes auf besonders einfache Weise verändert werden.

6.1.1.1 Lineare Grauwerttransformation

Bei einer (stückweise) **linearen Grauwerttransformation** errechnet sich der neue Grauwert $g'(i,j)$ aus dem alten Grauwert $g(i,j)$ nach folgender Vorschrift:

$$g'(i,j) = \text{mult} * g(i,j) + \text{add} \quad (\text{mult}, \text{add reelle Zahlen}). \quad (6-3)$$

Um sicherzustellen, dass die Grauwerte das zulässige Intervall $[0, g_{\text{MAX}}]$ nicht verlassen, wird zusätzlich vereinbart:

$$g'(i,j) = \begin{cases} g_{\text{MIN}} = 0 & \text{falls } \text{mult} * g(i,j) + \text{add} < g_{\text{MIN}} = 0 \\ g_{\text{MAX}} & \text{falls } \text{mult} * g(i,j) + \text{add} > g_{\text{MAX}} \end{cases} \quad (6-3a)$$

(Durch dieses „Abschneiden“ kann Bildinformation verloren gehen: vorher unterscheidbare helle bzw. dunkle Bildbereiche werden u.U. auf ein einheitliches Weiß bzw. Schwarz abgebildet. Korrekterweise müsste die Transformation als "stückweise" lineare Grauwerttransformation bezeichnet werden).

Der Zusammenhang zwischen den alten Grauwerten g und den neuen Grauwerten g' wird häufig in einem g - g' -Koordinatensystem dargestellt. Dabei wird nur das zweidimensionale Intervall $[0, g_{\text{MAX}}] \times [0, g_{\text{MAX}}]$ betrachtet.

Die Abbildungen 6.1 (a) und (b) zeigen ein Grauwertbild mit $g_{\text{MAX}}=255$ und sein zugehöriges Histogramm. In Abbildung 6.1 (c) ist der Verlauf der identischen Grauwerttransformation ($\text{mult}=1$ und $\text{add}=0$) dargestellt, bei der das Eingangsbild nicht verändert wird.

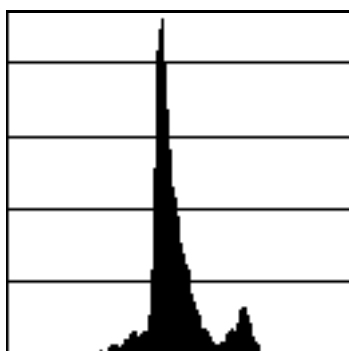
Durch eine Erhöhung bzw. Verringerung des Summanden add in (6-3) erreicht man eine Parallelverschiebung der Diagonalen aus Abbildung 6.1 (c). Dies entspricht einer Aufhellung bzw. Abdunkelung des Bildes, ohne dass der Bildkontrast beeinflusst wird; siehe (6-1). Die Abbildungen 6.1 (d) bis (f) zeigen ein Beispiel für eine lineare Grauwerttransformation mit $\text{add} = 75$ (und $\text{mult} = 1$). Sehr helle Bildpartien des Eingangsbild sind nun einheitlich weiß; siehe Balken am rechten Rand des Histogrammes.

Mit Hilfe des Faktors mult kann man den Bildkontrast - siehe (6-1) - beeinflussen: für $\text{mult} > 1$ nimmt der Bildkontrast zu, und für $0 < \text{mult} < 1$ wird das Bild kontrastärmer. Ein Beispiel für eine Kontraständerung mit $\text{mult}=1.5$ (und $\text{add}=0$) ist in Abbildung 6.1 (g) bis (i) dargestellt. Auch hier sind sehr helle Bildstellen nach der Transformation wieder einheitlich weiß.

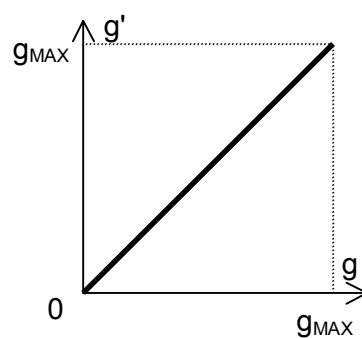
Durch geeignete Wahl von mult und add kann man erreichen, dass der zur Verfügung stehende Grauwertbereich optimal genutzt wird. Wenn g_{min} und g_{max} den kleinsten und den größten im Eingabebild tatsächlich auftretenden Grauwert bezeichnen, so erhält man mit $\text{mult} = g_{\text{MAX}} / (g_{\text{max}} - g_{\text{min}})$ und $\text{add} = -g_{\text{min}} * \text{mult}$ die Grauwerttransformation:



(a)



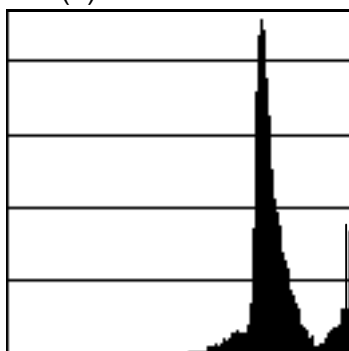
(b)



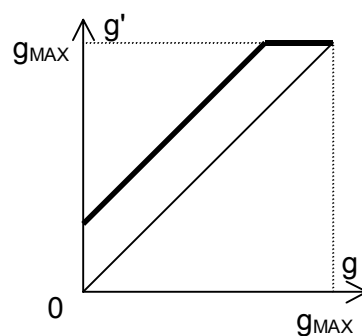
(c)



(d)



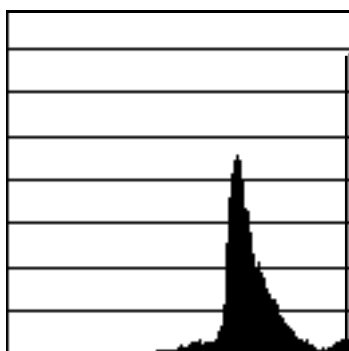
(e)



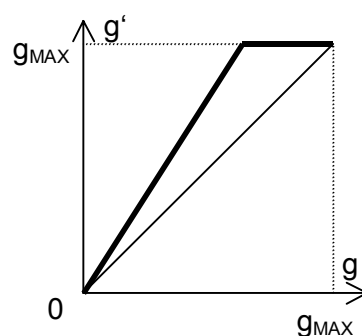
(f)



(g)



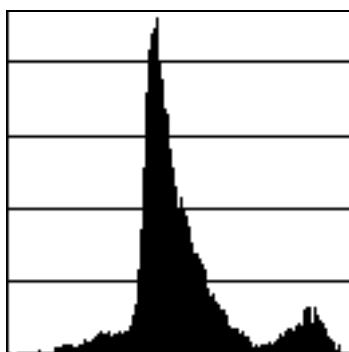
(h)



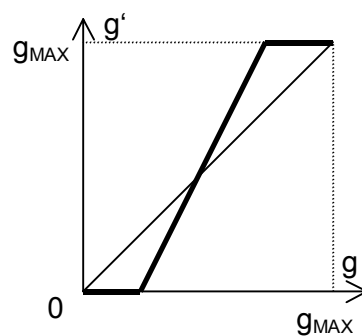
(i)



(j)



(k)



(l)

Abb. 6.1: Beispiele für die lineare Grauwerttransformation:

(a)-(c) Eingangsbild, Histogramm und identische Grauwerttransformation;

(d)-(f) aufgehelltes Bild, Histogramm und Grauwerttransformation;

(g)-(i) Bild mit gespreiztem Kontrast, Histogramm und Grauwerttransformation;

(j)-(l) optimal transformiertes Bild, Histogramm und Grauwerttransformation.

(6-4)

$$g' = \frac{g_{\text{MAX}} * (g - g_{\text{min}})}{g_{\text{max}} - g_{\text{min}}}$$

Nach dieser Transformation wird das zur Verfügung stehende Grauwertintervall $[0, g_{\text{MAX}}]$ vollständig ausnutzt; siehe Abbildung 6.1 (j) bis (l).

Häufig wird die lineare Grauwert-Transformation auch eingesetzt, wenn das Ergebnis einer Bildverarbeitungsoperation eine „Bild“-Matrix mit nicht darstellbaren Werten liefert; (z.B. die Faltung aus Kapitel 6.1.2.1.). Eine Transformation nach (6-4) transformiert die Ergebniswerte auf das zulässige Grauwert-Intervall $[g_{\text{MIN}}=0, g_{\text{MAX}}]$, so dass das Ergebnisbild anschließend dargestellt werden kann.

6.1.1.2 Gamma-Korrektur

Eine weitere Punktoperation zur Beeinflussung der Bildhelligkeit und des Bildkontrastes stellt die **Gammakorrektur** dar, die eine nicht-lineare Veränderung der Grauwerte bewirkt.

Die Gammakorrektur wird vor allem verwendet, um die nicht-linearen Grauwertverfälschungen auszugleichen, die z.B. bei der Aufnahme eines Bildes mit Hilfe einer CCD-Kamera (Charge Coupled Device) oder bei der Ausgabe eines Bildes auf einem Monitor zwangsläufig entstehen. Die Ursache für diese Verfälschungen ist in der Nichtlinearität der jeweiligen Bauelemente zu sehen: ein CCD-Sensorelement liefert bei einer Verdoppelung der empfangenen Lichtmenge nicht die doppelte Ausgangsspannung (d.h. keinen doppelt so hohen Grauwert), und umgekehrt bewirkt die Verdoppelung eines Bildpunkt-Grauwertes nicht die doppelte Helligkeit an der entsprechenden Monitorposition.

Beispielsweise gilt für einen Monitor, dessen Bildschirmpunkte die Grauwerte $g \in [0, g_{\text{MAX}}]$ mit einer Intensität $I \in [0, I_{\text{MAX}}]$ abbilden, die folgende Beziehung:

(6-5)

$$\frac{I}{I_{\text{MAX}}} = \left(\frac{g}{g_{\text{MAX}}} \right)^\gamma$$

wobei der Gammawert γ typischerweise zwischen 1.3 und 2.4 liegt. Damit die Ausgabe-Intensitäten den tatsächlichen (Grauwert-)Gegebenheiten entsprechen, müssen die Grauwerte vor der Bildausgabe mit Hilfe des Kehrwertes von γ korrigiert werden:

(6-6)

$$\frac{g'}{g_{\text{MAX}}} = \left(\frac{g}{g_{\text{MAX}}} \right)^{1/\gamma} \quad \text{d.h.} \quad g' = \left(\frac{g}{g_{\text{MAX}}} \right)^{1/\gamma} * g_{\text{MAX}}$$

(Dabei liegen die Quotienten g/g_{MAX} und g'/g_{MAX} im Intervall $[0, 1.]$)

Bei der Bildaufnahme besteht zwischen der einfallenden Lichtenergie und dem daraus erzeugten Grauwert eine (6-5) entsprechende Beziehung, wobei hier die Gammawerte üblicherweise zwischen 0.4 und 1.0 liegen.

Abbildung 6.2 zeigt ein Beispiel für eine Gammakorrektur nach (6-6). Auf der linken Seite ist ein Eingangsbild mit einem synthetischen Graukeil mit gleichmäßig ansteigenden Grauwerten zu sehen. In dieser unkorrigierten Darstellung sieht man, dass durch den spezifischen Gammawert $\gamma=1.8$ der Bildausgabe große Teile des Graukeils (und damit auch des restlichen Bildes) zu dunkel abgebildet werden. In Abbildung 6.2 (b) ist das Ergebnis der Gammakorrektur mit $\gamma = 1.8$ dargestellt. Der Funktionsverlauf dieser Gammakorrektur ist in Abbildung 6.2 (c) als durchgezogene Linie eingezeichnet.

Schlussbemerkungen zur Gammakorrektur:

Falls der Gammawert bei der Bildaufnahme der Kehrwert des Gammawertes bei der Bildausgabe ist, so gleichen sich die Verzerrungen aus und man muss keine Gammakorrektur durchführen. Dieser Fall ist aber in der Realität fast nie gegeben.

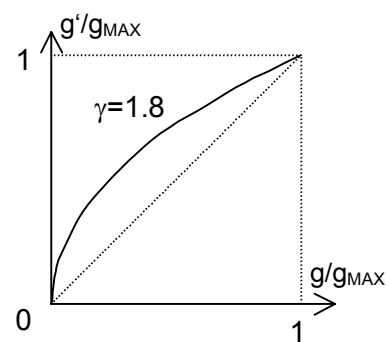
In der Praxis sind (z.B. im Internet) viele Bilddateien bereits im Hinblick auf einen durchschnittlichen Monitor gammakorrigiert, so dass keine weitere Bearbeitung erforderlich ist.



(a)



(b)



(c)

Abb. 6.2: Beispiel für die Gammakorrektur: (a) Eingangsbild mit Graukeil; (b) gammakorrigiertes Bild; (c) Funktionsverlauf der Gammakorrektur.

Nicht vorkorrigierte Bilddaten müssen mit Hilfe des monitorspezifischen Gammawertes nach der Vorschrift (6-6) korrigiert werden, um eine optimale Darstellung zu erzielen. Den monitorspezifischen Gammawert ermittelt man am besten mit Hilfe eines Messprogramms, wie es z.B. in den meisten Bildbearbeitungsprogrammen enthalten ist.

Auch für Scanner, Kameras und andere Geräte zum Bildeinzug kann der spezifische Gammawert z.B. mit Hilfe genormter Grauwertverläufe (z.B. Graukeil) oder spezieller Messgeräte (Densitometer) ermittelt werden.

In Programmen wird die Gammakorrektur in der Regel mit Hilfe einer LUT (Look Up Table) realisiert.

6.1.1.3 Reduktion der Anzahl verschiedener Grauwerte

Ein häufig eingesetztes Mittel, um Bilder künstlerisch zu verfremden oder Speicherplatz einzusparen ist es, die Anzahl der zulässigen Graustufen eines Bildes zu reduzieren.

Wenn man alle Grauwerte g , deren höchstwertiges Bit in der binären Darstellung des Grauwertes nicht gesetzt ist, auf den neuen Grauwert $g'_{\text{MIN}} = 0$ (schwarz), und alle Grauwerte mit gesetztem höchstwertigen Bit auf $g'_{\text{MAX}} = 1$ (weiß) transformiert, so erhält man ein Binärbild; siehe Abbildung 6.3 (a). Das binarisierte Bild lässt sich (auch ohne weitere Kompression) mit einem Bit pro Pixel abspeichern. (Eine flexiblere Alternative zur Erstellung eines **Binärbildes** wird in Kapitel 14 vorgestellt).

Statt nur das höchstwertige Bit zu berücksichtigen, kann man auch die $n = 1, 2, 3, \dots$ höchstwertigen Bits auswerten und erhält damit eine äquidistante Aufspaltung des ursprünglichen Grauwertintervalls und ein neues Bild mit 2^n Graustufen. Die Abbildung 6.3 (b) zeigt ein Beispiel für ein derartiges **Äquidensitenbild** (auch: Poster), wobei hier $n=2$ gewählt wurde. Zur Abspeicherung des Bildes sind nun nur noch n Bits je Pixel erforderlich.

Äquidensitenbilder (oder allgemeiner Binärbilder) werden häufig auch als Vorstufe für eine **Bildsegmentierung** erzeugt.



(a)



(b)

Abb. 6.3: Beispiele für die Reduzierung der Grauwert-Anzahl zu Abbildung 6.1 (j): (a) Äquidensitenbild unter Berücksichtigung des höchstwertigen Bits; (b) Äquidensitenbild unter Berücksichtigung der beiden höchstwertigen Bits.

6.1.2 Lokale Bildoperatoren

Bei den **lokalen Bildoperatoren** wird – wie bei den Punktoperatoren – eine Berechnungsvorschrift parallel auf alle Bildpunkte angewendet, um die alten Grauwerte $g(i,j)$ des Eingangsbildes B in die neuen Grauwerte $g'(i,j)$ des Ergebnisbildes B' zu überführen. Im Gegensatz zu den Punktoperatoren werden dabei aber die Grauwerte aus der Nachbarschaft der jeweiligen Position (i,j) mitberücksichtigt. (6-7)

In der Regel wird eine der beiden folgenden Nachbarschaftsbeziehungen verwendet:

- die **4er-Nachbarschaft (N4)**, die aus den horizontalen und den vertikalen Nachbarn an den Positionen $(i,j-1)$, $(i-1,j)$, $(i+1,j)$ und $(i,j+1)$ besteht (siehe Abbildung 6.4 (links)),
- und die **8er-Nachbarschaft (N8)**, die zusätzlich die diagonalen Nachbarn $(i-1,j-1)$, $(i+1,j-1)$, $(i-1,j+1)$ und $(i+1,j+1)$ umfasst; (siehe Abbildung 6.4 (rechts)).

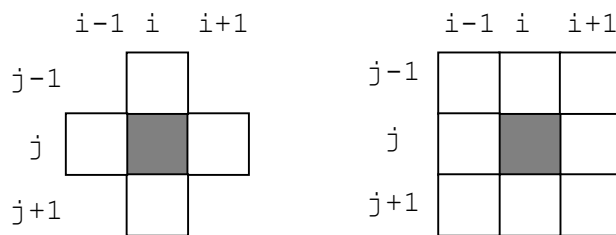


Abb. 6.4: N4-Nachbarschaft (links) und N8-Nachbarschaft (rechts).

Im weiteren wird der Einfachheit halber meist die N8-Nachbarschaft verwendet; möglich wären aber auch die N4-Nachbarschaft oder "größere" Nachbarschaften.

Als Beispiele für lokale Bildoperatoren werden im folgenden die Faltung und die Rangfolgeoperatoren betrachtet. Dabei werden diese beiden Operatoren hier speziell zur Glättung des Bildes, zur Markierung von Kanten und zur Hervorhebung von Grauwertübergängen eingesetzt. Die Bilder sollen M Spalten und N Zeilen besitzen.

6.1.2.1 Faltung

Bei der **Faltung** (engl. convolution) zwischen einem Bild B mit Grauwerten $g(i,j)$ und einer 3×3 -Faltungsmatrix F mit Werten $f(k,l)$ wird die Faltungsmatrix über alle Positionen (i,j) des Bildes mit $0 < i < M-1$ und $0 < j < N-1$ gelegt. Die Elemente der Faltungsmatrix werden auch als Gewichte bezeichnet. An jeder Position (i,j) wird ein Ergebniswert $e(i,j)$ als gewichtete Summe nach folgender Vorschrift berechnet:

$$e(i,j) = \sum_{l=0}^2 \sum_{k=0}^2 \{g(i-1+k, j-1+l) * f(k,l)\}. \quad (6-8)$$

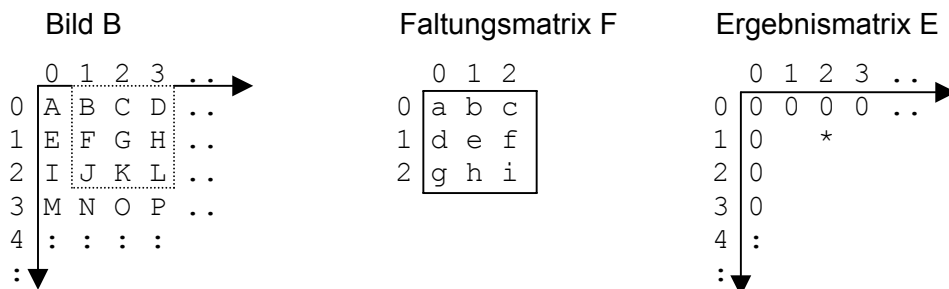
Die vorderste und die hinterste Spalte sowie die oberste und die unterste Zeile der Ergebnismatrix E wird auf 0 gesetzt, um Sonderfallbehandlungen zu vermeiden.

Die Ergebniswerte $e(i,j)$ sind i.d.R. keine zulässigen Grauwerte; (sie können z.B. sehr groß, aber auch negativ sein). Um die Ergebnismatrix als Bild ausgeben zu können, wird deshalb üblicherweise abschließend eine lineare Grauwerttransformation nach (6-4) durchgeführt, die die Ergebniswerte $e(i,j)$ in das Intervall $[g_{\min}=0, g_{\max}]$ abbildet:

$$g'(i,j) = \frac{g_{\max} * (e(i,j) - e_{\min})}{e_{\max} - e_{\min}}. \quad (6-9)$$

Dabei stehen e_{\min} bzw. e_{\max} für den minimalen bzw. den maximalen Ergebniswert der Faltung. Hier können entweder die in der Ergebnismatrix auftretenden minimalen bzw. maximalen Faltungswerte oder aber die theoretisch minimal bzw. maximal möglichen Ergebniswerte eingesetzt werden.

Beispiel:



An der mit einem Sternchen bezeichneten Stelle in der Ergebnismatrix erhält man z.B.:

$$e(2,1) = B*a + C*b + D*c + F*d + G*e + H*f + J*g + K*h + L*i.$$

Die Faltung mit der Faltungsmatrix F_1 (mit I für Identität)

(6-10)

$$F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

verändert ein Bild nicht, wenn man von dem ein-pixel-breiten schwarzen Rand im Ergebnisbild absieht. Diese Faltung wird hier nur vorgestellt, da sie als Teil einiger späterer Faltungen verwendet wird.

Im folgenden werden einige anwendungsorientierte Beispiele für Faltungen angegeben.

Glättung der Bildfunktion (Weichzeichnen).

Mit der Faltungsmatrix

(6-11)

$$F_M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & n & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (\text{beispielsweise mit } n = 1).$$

wird an jeder Bildposition der gewichtete Mittelwert der Grauwerte aus der 3*3-Umgebung eingesetzt. Durch den **gewichteten Mittelwertoperator** werden Störungen verwischt, so dass das Bild „glatter“ wirkt. Als weiterer nachteiliger Effekt werden die Kanten im Bild unscharf; siehe Abbildung 6.4 (b).

Ein weiteres Beispiel zur **Glättung (Weichzeichnung)** des Bildes stellen sogenannte Gauss-Filter dar, die einen Tiefpass-Filter realisieren. Das bedeutet, dass die hohen Frequenzanteile, die den scharfen Kanten entsprechen, herausgefiltert werden und nur die niedrigeren Frequenzen „passieren“ können. Ein Beispiel für einen Gauss-Filter erhält man mit der Faltungsmatrix F_G mit:

(6-12)

$$F_G = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Kantendetektion und Hervorhebung von Kanten

Die im folgenden angegebenen Faltungsmatrizen sind **Kantendetektoren**, die Kanten senkrechter, waagrechter sowie diagonaler Ausrichtung ermitteln:

(6-13)

$$F_S = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad F_w = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad F_{D1} = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \quad \text{und} \quad F_{D2} = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

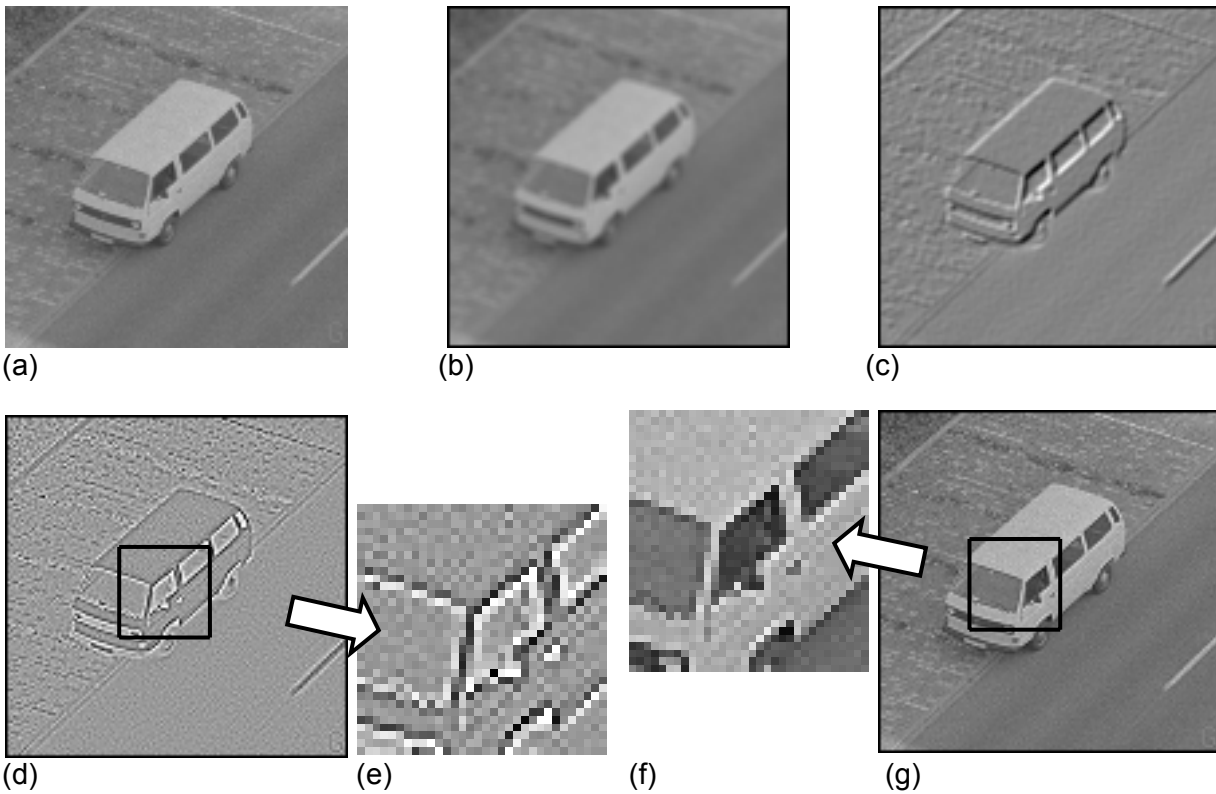


Abb. 6.4: Beispiele für Faltungen:

- (a) Eingangsbild;
- (b) Mittelwertoperator zur Glättung der Bildfunktion (Weichzeichnen);
- (c) Kantendetektor für diagonal (/) verlaufende Kanten;
- (d) Laplace-Operator zur Kantendetektion;
- (e) Ausschnittsvergrößerung aus dem Ergebnis des Laplace-Operators;
- (f) Ausschnittsvergrößerung aus dem Ergebnis des Kontrastverbesserungsfilters;
- (g) Kontrastverbesserungsfilter.

Abbildung 6.4 (c) zeigt ein Beispiel für die Anwendung des Kanten-Filters F_{D2} . Die mittel-grauen Bildbereiche entsprechen Positionen mit einem Faltungswert $e \approx 0$, d.h. es wurden dort keine Kanten detektiert.

Die oben aufgeführten Faltungen basieren auf einer partiellen Ableitung der Bildfunktion in waagrechter, senkrechter bzw. diagonaler Richtung.

Durch eine Addition der vier Kantendetektoren aus (6-13) erhält man den **Kanten-Kombinationsfilter** F_K , der die Kanten aller Richtungen ermittelt:

$$F_K = F_S + F_W + F_{D1} + F_{D2} = \begin{bmatrix} -3 & -3 & -1 \\ -1 & 0 & 1 \\ 1 & 3 & 3 \end{bmatrix} \quad (\text{gleichwertig zu: } \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}). \tag{6-14}$$

Einen **Relief-Filter** F_R , der die Kanten im Bild hervorhebt, erhält man, wenn die Matrizen der identischen Faltung F_I und eines ausgewählten Kantendetektors F_S , F_W , F_{D1} oder F_{D2} gewichtet addiert; z.B.:

$$F_R = n \cdot F_I + F_{D2} = \begin{bmatrix} -1 & -1 & 0 \\ -1 & n & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad (\text{beispielsweise mit } n=5). \tag{6-15}$$

Ein weiterer Operator, der sowohl zur Kantendetektion als auch zur Kontrastverbesserung eingesetzt werden kann, ist der **Laplace-Operator** mit der Faltungsmatrix F_L :

(6-16)

$$F_L = \begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array}$$

Die Anwendung dieser Faltung entspricht einer zweifachen partiellen Ableitung der Bildfunktion in senkrechter und waagrechter Richtung; siehe Abbildung 6.4 (d). Die Ausschnittsvergrößerung in Abbildung 6.4 (e) zeigt, dass im Bereich von Kanten aller Richtungen je ein dunkler und ein heller Streifen entsteht. Die exakte Position der Kante ist der Übergang zwischen diesen beiden Streifen, der einem Nulldurchgang der zweiten Ableitung entspricht.

Durch eine gewichtete Addition der identischen Faltung mit dem Laplace-Operator erhält man den **Kontrastverbesserungsfilter** F_V ; siehe Abbildungen 6.4 (g) und (f):

(6-17)

$$F_V = n \cdot F_L - F_L = \begin{array}{|c|c|c|} \hline 0 & -1 & 0 \\ \hline -1 & n+4 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array} \quad (\text{beispielsweise mit } n=5).$$

6.1.2.2 Rangfolgeoperatoren

Bei den **Rangfolgeoperatoren** werden – wie bei der Faltung – an jeder Bildposition (i,j) der Grauwert der aktuellen Stelle und die Grauwerte z.B. der N8-Nachbarschaft betrachtet. Dabei werden diese neun Grauwerte der Größe nach sortiert:

(6-18)

$$g_0 \leq g_1 \leq \dots \leq g_7 \leq g_8.$$

Im folgenden wird an einigen Beispielen gezeigt, wie man durch die Wahl bestimmter Positionen aus dieser Sortierung Bildstörungen wirkungsvoll eliminieren und Kanten ermitteln kann.

Störungselimination

Wenn man aus der Sortierung (6-18) den Median, d.h. den in der Mitte stehenden Wert auswählt,

(6-19)

$$g'(i,j) = g_4,$$

so erhält man den **Median-Operator**. Dieser Operator unterdrückt Störungen (wie z.B. Bildrauschen), ohne dabei jedoch die Kanten zu verschleifen, wie dies beim Mittelwertoperator (6-11) der Fall ist; siehe Abbildung 6.5 (b).

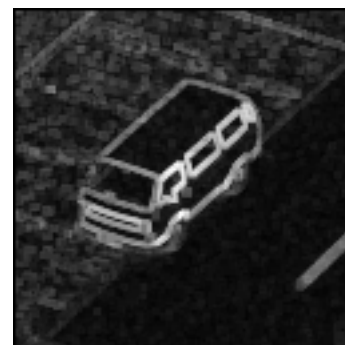
Zwei weitere Möglichkeiten zur Störungselimination mit Rangfolgeoperatoren werden in Kapitel 14 vorgestellt.



(a)



(b)



(c)

Abb. 6.5: Beispiele für Rangfolgeoperatoren:

- (a) Eingangsbild;
- (b) Medianoperator zur Störungselimination;
- (c) Spannweitenoperator zur Detektion von Kanten aller Richtungen.

Kantendetektion

Der **Spannweitenoperator**

(6-20)

$$g'(i,j) = g_8 - g_0$$

ist ein Kantendetektor, der Kanten aller Ausrichtungen ermittelt; siehe Abbildung 6.5 (c).

Schlussbemerkung zu den lokalen Operatoren:

Alle in diesem Teilkapitel vorgestellten lokalen Operatoren wurden unter Verwendung der N8-Nachbarschaft definiert. Alle Formeln sind jedoch direkt auf die Verwendung der N4-Nachbarschaft oder beliebiger "größerer" Nachbarschaften übertragbar.

6.2 Geometrische Bildmanipulationen

Unter **geometrischen Bildmanipulationen** versteht man Verarbeitungsschritte, die die geometrischen Eigenschaften eines Bildes ändern: z.B. Größenänderung, Spiegelung, Drehung, Scherung oder perspektivische Projektion eines Bildes.

Auf diesen globalen Transformationen, die ein komplettes Bild nach einer einheitlichen Vorschrift verändern, bauen weitergehende geometrische Transformationen, wie z.B. Warping und Morphing auf, die unterschiedliche Teile eines Bildes unterschiedlich verändern können. Beim **Warping** kann ein Bild beliebig verzerrt werden: so können z.B. in einem Gesichtsbild gezielt die Mundwinkel angehoben werden, um ein Lächeln zu bewirken. Beim **Morphing** werden Bilder geometrisch und photometrisch verändert, um durch eine zeitliche Interpolation einen fließenden Übergang zwischen zwei Bildern zu erreichen: z.B. kontinuierliche Verwandlung einer Person in eine andere.

Im folgenden wird nur auf die Größenänderung (Skalierung) eines Bildes eingegangen, da diese in der Praxis besonders häufig auftritt: z.B. Anpassung der Größe eines Bildes, damit es auf einer Text- oder Bildschirmseite eine bestimmte Fläche belegt.

Besonders einfach sind **ganzzahlige Vergrößerungen** und **Verkleinerungen**, da sie durch Vervielfachen bzw. Zusammenfassen von Pixeln erreicht werden können. Die Abbildungen 6.6 (b) und (c) zeigen zwei Beispiele für Verkleinerungen des Originalbildes aus Abbildung 6.6 (a) um den Faktor 4 bzw. 8 mit anschließender Rückvergrößerung auf die Originalgröße. Derartige Verkleinerungen (ohne Rückvergrößerung) stellen eine Bilddatenreduktion dar, die aber mit einem erheblichen Verlust an Detailinformation verbunden ist.



(a)



(b)



(c)

Abb. 6.6: Beispiele für ganzzahlige Verkleinerungen eines Bildes mit anschließender Rückvergrößerung:

- (a) Verkleinerung in x- und y-Richtung um den Faktor 4;
- (b) Verkleinerung in x- und y-Richtung um den Faktor 8.

Bei ganzzahligen Vergrößerung ist zu berücksichtigen, dass störende Pixelblöcke entstehen können; siehe Abbildungen 6.6. (b) und (c). Wenn diese Blöcke durch anschließendes Weichzeichnen verschliffen werden, so ergibt sich ein Verlust an Bildinformation. Derselbe Effekt (der einer Tiefpass-Filterung entspricht) ergibt sich bei der Verkleinerung, da auch hier feine Bilddetails und scharfe Kanten durch die Zusammenfassung von Pixeln verloren gehen können.

Bei Verkleinerungen können Effekte entstehen, wie sie auch bei der Abtastung einer Vorlage mit einer zu grob gewählten Auflösung entstehen (Unterabtastung). Abbildung 6.7 (b) zeigt als Beispiel die Verkleinerung der Vorlage aus Abbildung 6.7 (a) auf zwei Drittel der Größe. Das Originalbild enthält ein-pixel-breite senkrecht verlaufende schwarze bzw. weiße Linien, die in der Verkleinerung zu erheblich weniger hell- und dunkelgrauen Streifen zusammengefasst werden.

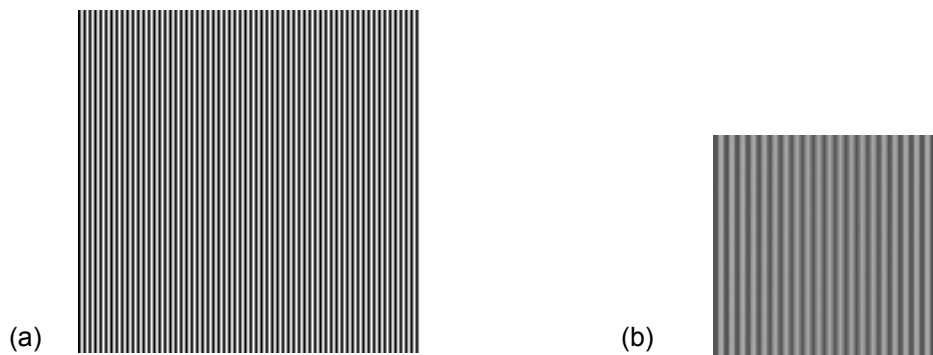


Abb. 6.7: Beispiel für eine Verkleinerung (b) einer Vorlage (a).

Eine besonders effiziente Möglichkeit zur Bildskalierung geht auf Fant¹ zurück;. Bei diesem Ansatz wird zuerst jede Bildzeile auf die gewünschte Länge skaliert. Dazu wird aus der Eingangsbildbreite und der Zielbildbreite der Skalierungsfaktor berechnet. Nun laufen zwei Indizes an der Eingangsbild- und der Zielbildzeile entlang (siehe waagrechte Pfeile in Abbildung 6.8), wobei der Skalierungsfaktor die Bewegungsgeschwindigkeiten der Indizes bestimmt. Die Pixel der längeren Zeile werden in einem Akkumulator gesammelt. Jedesmal, wenn der Index der kürzeren Zeile ein Pixel vorgerückt ist, wird aus den Pixeln des Akkumulators ein Mittelwert gebildet, und dieser wird als Ausgabe-Grauwertwert in die Ausgabezeile eingesetzt. „Angeschnittene“ Pixel der längeren Zeile werden anteilig berücksichtigt.

Abschließend wird derselbe Vorgang spaltenweise (auf das bereits zeilenweise interpolierte Bild) angewendet.

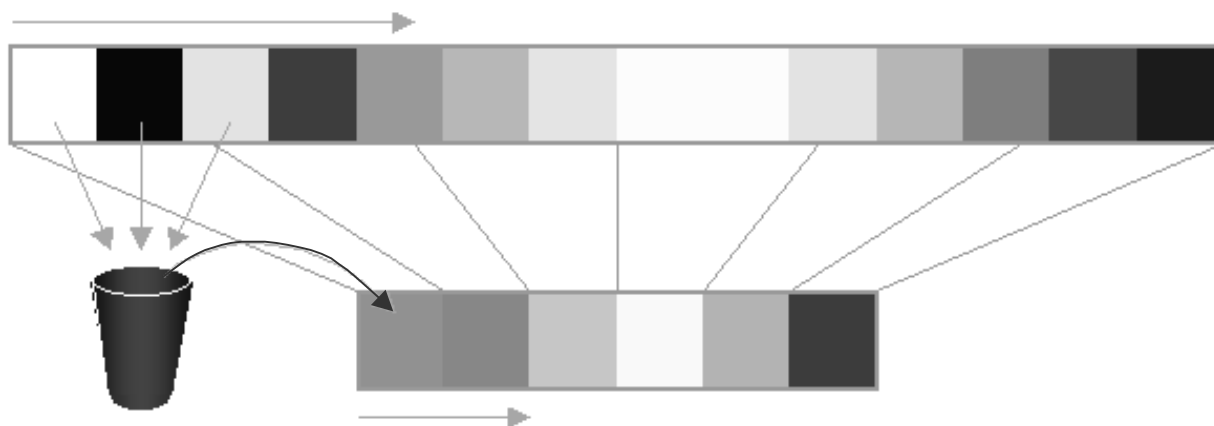


Abb. 6.8: Prinzip der Skalierungsmethode nach Fant; Abbildung aus ².

Die Methode von Fant wird z.B. bei der Windows-Bitmap-Skalierung und bei vielen anderen Bildbearbeitungs-Tools eingesetzt.

Für beliebige geometrische Bildtransformationen kann man die sogenannten **indirekten Verfahren** einsetzen. Bei diesen wird mit Hilfe der inversen Transformationsvorschrift zu jedem Pixel des Ausgabebildes die entsprechende Urbildstelle im Eingabebild berechnet; siehe Abbildung 6.9. Falls diese Stelle nicht exakt im Pixelraster liegt (siehe schwarzer Punkt in Abb. 6.9 (a)), so kann an der Zielposition der Grauwert des nächst liegenden Pixels aus

¹ K. Fant; IEEE CG&A 1/86; Patent Honeywell

² B. Kreling; Skript zur Vorlesung „Multimedia“

dem Eingabebild eingesetzt werden (Interpolation 0. Ordnung: nearest neighbor resampling; siehe Pfeil von rechts oben). Statt den Grauwert des nächstliegenden Pixels einzusetzen, kann der Zielgrauwert auch aus den Grauwerten der im Eingabebild benachbarten Bildpunkte höherwertig interpoliert werden; (z.B. Interpolation 1. Ordnung: bilineare Interpolation mit Hilfe der vier benachbarten Bildpunkte; siehe alle vier kurzen Pfeile in Abbildung 6.9 (a)).

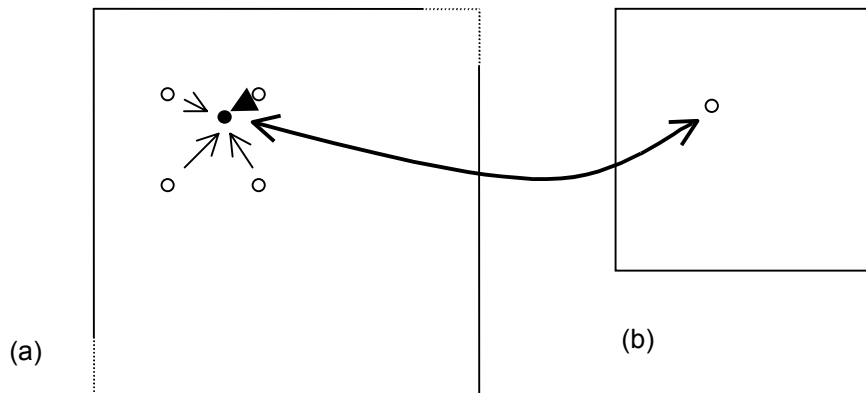


Abb. 6.9: Prinzip der indirekten Methode zur geometrischen Bildtransformation:

- (a) Eingabebild, das verkleinert werden soll; (die nicht gefüllten Kreise stehen stellvertretend für die Mittelpunkte von vier benachbarten Pixeln);
- (b) verkleinertes Ausgabebild.

Schlussbemerkung: Grundsätzlich gilt, dass bei (fast) jeder geometrischen Bildtransformation Bildinformation verloren geht. Hieraus folgt für den praktischen Einsatz, dass man ein Bild nie mehrmals hintereinander transformieren (z.B. skalieren) soll. Statt dessen sollte man bei jedem neuen Versuch wieder auf das Originalbild zurückgreifen.