2. Bild im Ortsbereich

2.1 Matrixform

Ein Bild läßt sich als Matrix beschreiben, mit

f(x,y) : Grauwert an der Position (x,y)
M : Spaltenzahl (*columns*)
N : Zeilenzahl (*rows*)

Traditionell ist der Bildursprung oben links.

2.2 Ortsauflösung

512 x 512 Bildpunkte

256 Grauwerte (8 bit)

--> 262 144 Byte

256 x 256 Bildpunkte

256 Grauwerte (8 bit)

--> 65 536 Byte

128 x 128 Bildpunkte

256 Grauwerte (8 bit)

--> 16 384 Byte

64 x 64 Bildpunkte

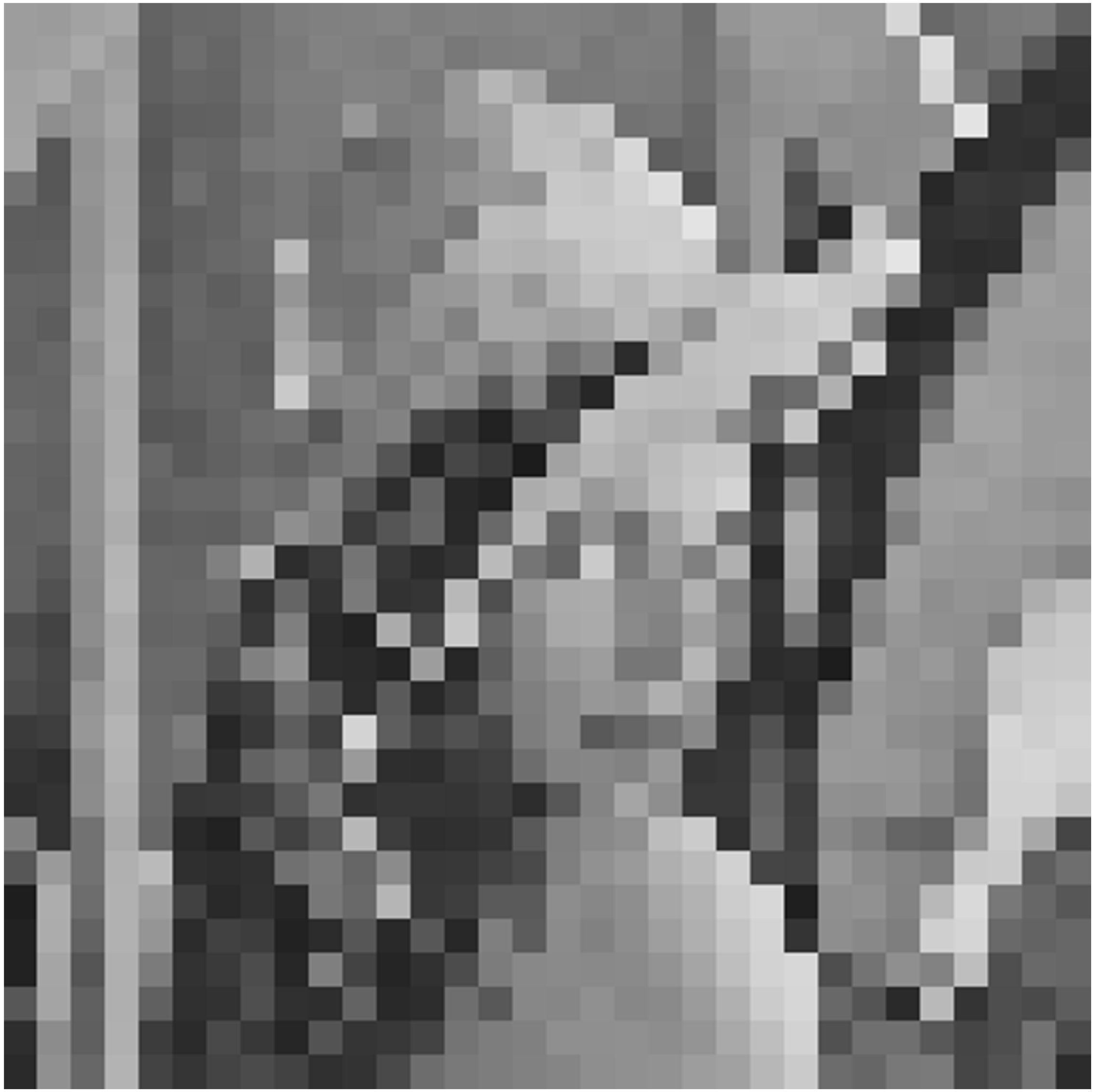
256 Grauwerte (8 bit)

--> 4096 Byte

32 x 32 Bildpunkte

256 Grauwerte (8 bit)

--> 1024 Byte



2.3 Grauwertauflösung

256 x 256 Bildpunkte

256 Grauwerte (8 bit)

256 x 256 Bildpunkte

16 Grauwerte (4 bit)

256 x 256 Bildpunkte

8 Grauwerte (3 bit)

256 x 256 Bildpunkte

4 Grauwerte (2 bit)

256 x 256 Bildpunkte

2 Grauwerte (1 bit)

= Binärbild

ÜBUNG: Graustufenreduktion

Geben Sie einen Algorithmus (in C++) an, welcher die Graustufen eines 8-bit Grauwertbildes (Name: `pic`) auf 16 Graustufen reduziert.

Hinweise:

Bild „pic" sei vom Typ `channel8` (= unsigned char = 8bit-Grauwertbild).

Die Bildgröße eines Bildes "*pic*„ in x- und y-Richtung erhält man mit den Befehlen:

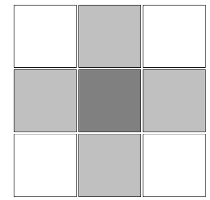
```
y_size = pic.rows();        // Integer-Format
x_size = pic.columns();
```

Den Grauwert eines Bildpunktes an der Stelle (x,y) erhält bzw. setzt man mit :

```
pic[y][x] = Grauwert;        // unsigned char
Grauwert  = pic[y][x];
```

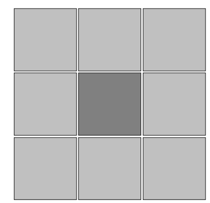
2.4 Beziehungen zwischen Bildpunkten

4-er-Nachbarschaft eines Bildpunktes (x,y):
(x-1,y), (x+1,y), (x,y-1), (x,y+1)



(1)

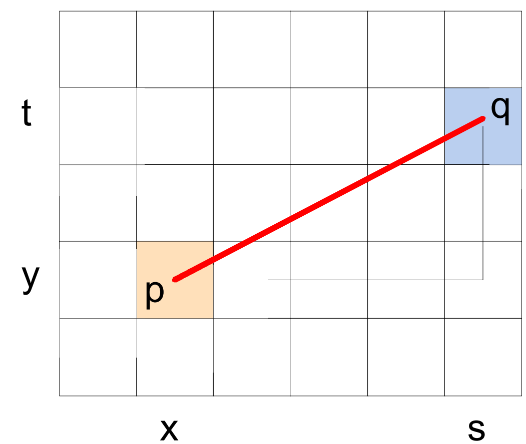
8-er-Nachbarschaft eines Bildpunktes (x,y):
wie (1) und zusätzlich (x-1,y-1), (x+1,y+1),
(x+1,y-1), (x-1,y+1)



(2)

Euklidische Distanz zwischen 2 Bildpunkten
p=(x,y) und q=(s,t):

$$D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$$



ÜBUNG: Glättungsfilter "gleitender Mittelwert"

Geben Sie einen Algorithmus (in C++) an, welcher ein 8-bit Grauwertbild dadurch glättet, indem der Grauwert eines Bildpunktes (x,y) durch den Mittelwert über seine 8-er Nachbarschaft ersetzt wird (**f**=Quellbild, **g**=Zielbild).

$$g(x,y) = \frac{1}{9} \cdot \left[f(x,y) + f(x-1,y) + f(x-1,y-1) + \;\ldots.. \; + f(x-1,y+1) \right]$$

2.5 Bildeigenschaft : Histogramm

2.5.1 Definition

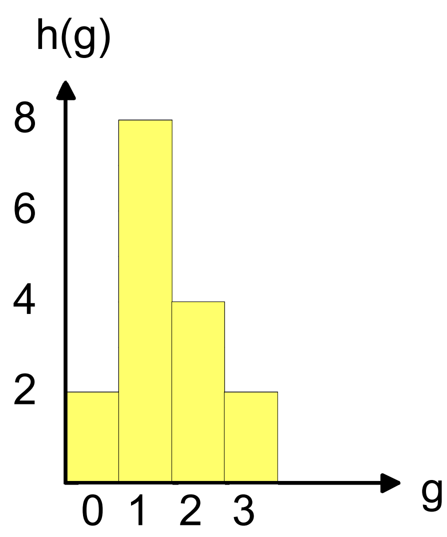
Das Histogramm gibt die Häufigkeitsverteilung der Grauwerte im Bild wieder.

Das Histogramm ist eine diskrete Funktion, für die gilt:

$$h(g) = n_g \qquad \text{mit } g: \quad \text{Grauwert}$$
$$n_g: \quad \text{Anzahl der Bildpunkte, die diesen Grauwert haben.}$$

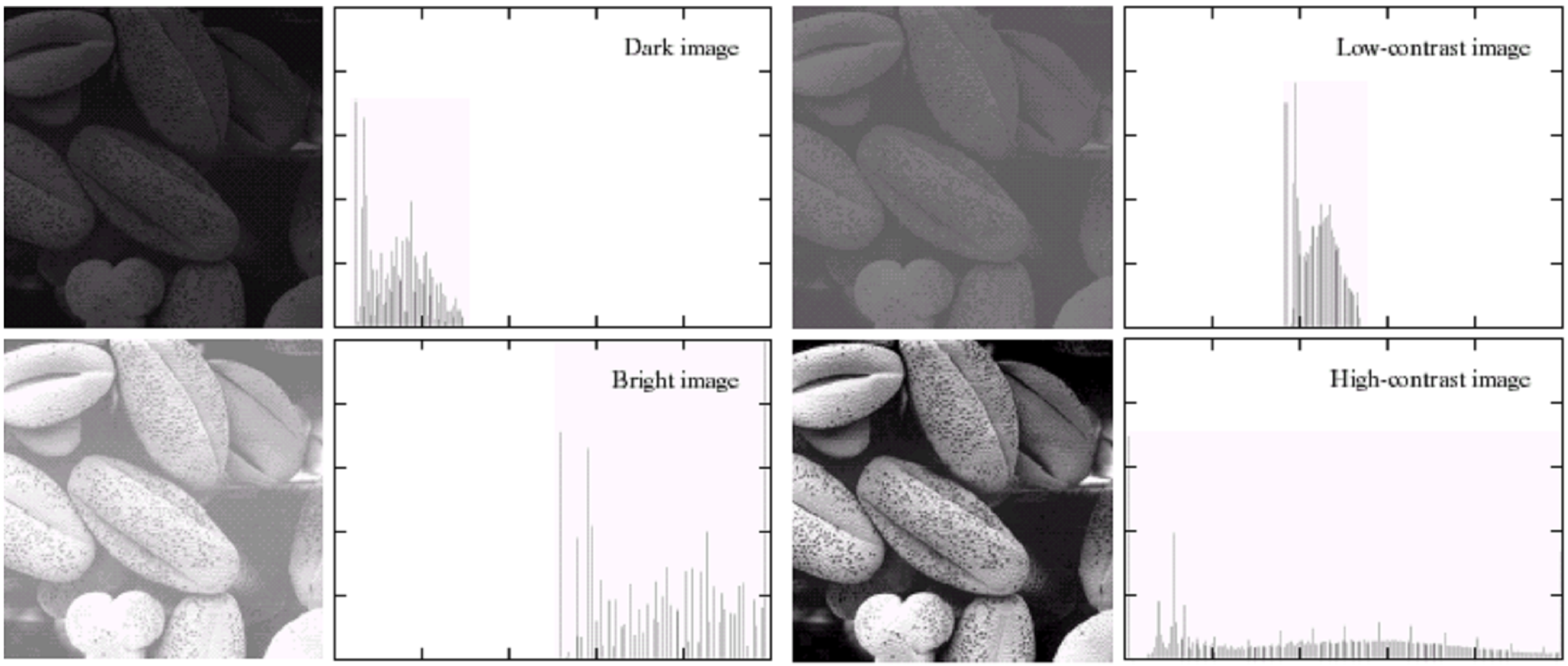
1	1	1	1
1	3	3	1
2	2	2	0
2	1	1	0

4x4 Bild mit 2 bit Grauwertauflösung



Histogramm

Zusammenhang zwischen dem Histogramm des Bildes und seinem Erscheinungsbild (Helligkeit, Kontrast).



aus "Digital Image Processing", Gonzalez/Woods

ÜBUNG: Histogrammberechnung

Geben Sie einen Algorithmus (in C++) an, das Histogramm eines Bildes berechnet.

Hinweise:

Die Bildgröße eines Bildes "pic" in x- und y-Richtung erhält man mit den Befehlen

```
y_size = pic.rows();         // Integer-Format
x_size = pic.columns();
```

Den Grauwert eines Bildpunktes an der Stelle (x,y) erhält bzw. setzt man mit

```
pic[y][x] = Grauwert;        // unsigned char
Grauwert  = pic[y][x];
```

2.5.2 Statistische Kennwerte von Histogrammen

Summe der häufigkeitsgewichteten Grauwerte

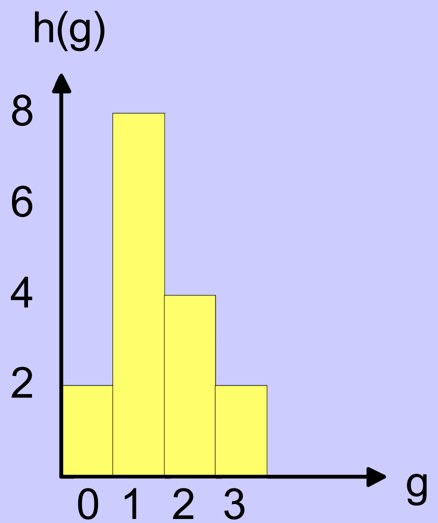
Mittelwert: $$\mu = \frac{\sum_{i=0}^{n} g_i \cdot h(g_i)}{\sum_{i=0}^{n} h(g_i)}$$

Anzahl der Bildpunkte

quadr. Mittelwertabweichung

Varianz: $$\sigma^2 = \frac{\sum_{i=0}^{n} (g_i - \mu)^2 h(g_i)}{\sum_{i=0}^{n} h(g_i)}$$

Beispiel:



Histogramm

$$\mu = \frac{0\cdot2+1\cdot8+2\cdot4+3\cdot2}{2+8+4+2} = \frac{22}{16} = 1.375$$

$$\sigma^2 = \frac{(0-1.375)^2\cdot2 + (1-1.375)^2\cdot8 + \ldots\ldots}{2+8+4+2} = 0.73$$

3. Bildvorverarbeitung

Ziel und Zweck

- Bildverbesserung (bezüglich eines bestimmten Zwecks)
  Beispiel: Kontrastverbesserung

- Hervorhebung interessanter Details (für die nachfolgende Bildverarbeitung)
  Beispiel: Kanten hervorheben

- Segmentierung (Bildbereiche mit gleichen Eigenschaften markieren)
  Beispiel: Hautfarbene Bildbereiche markieren

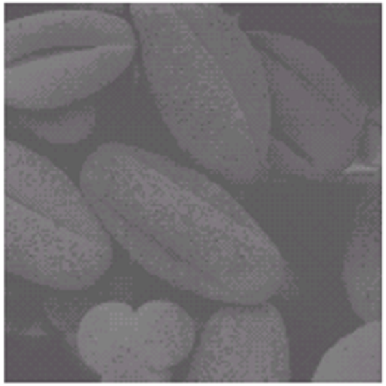
- Geometrische Transformationen (Skalierung, Rotation, Entzerrung, ...)
  Beispiel: Linsenverzeichnungskorrektur

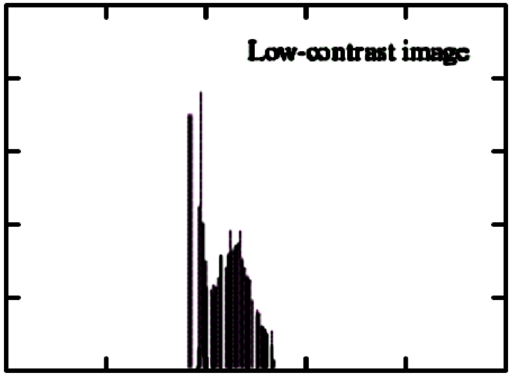
3.1 Grauwert-Transformationen

3.1.1 Histogrammausgleich (Kontrastverstärkung)

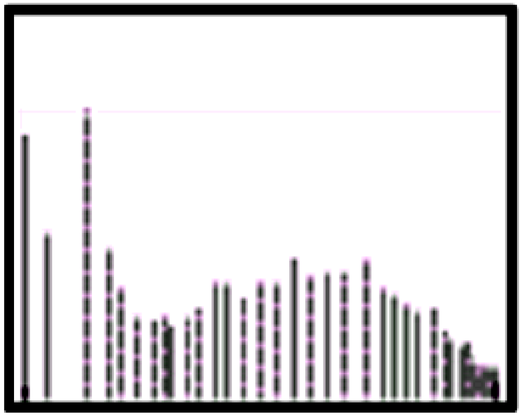
Grundgedanke: Je signifikanter (häufiger) ein Grauwert ist, desto deutlicher sollte er sich von den anderen Grauwerten absetzen.

vorher





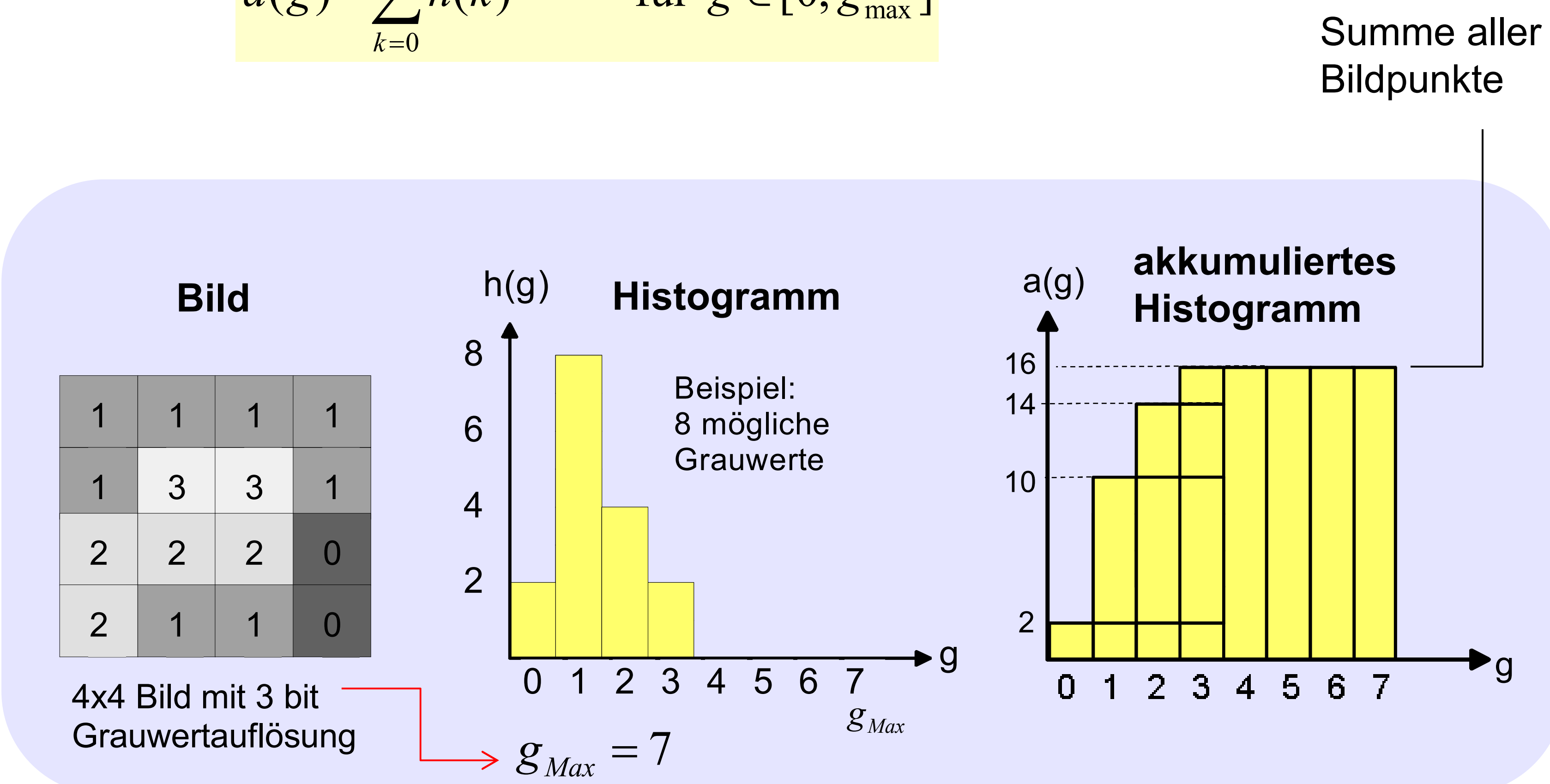
nachher



Funktionsweise des Histogrammausgleichs

Schritt 1: Berechnung des akkumulierten Histogramms

$$a(g)=\sum_{k=0}^{g} h(k) \qquad \text{für } g \in [0, g_{\max}]$$



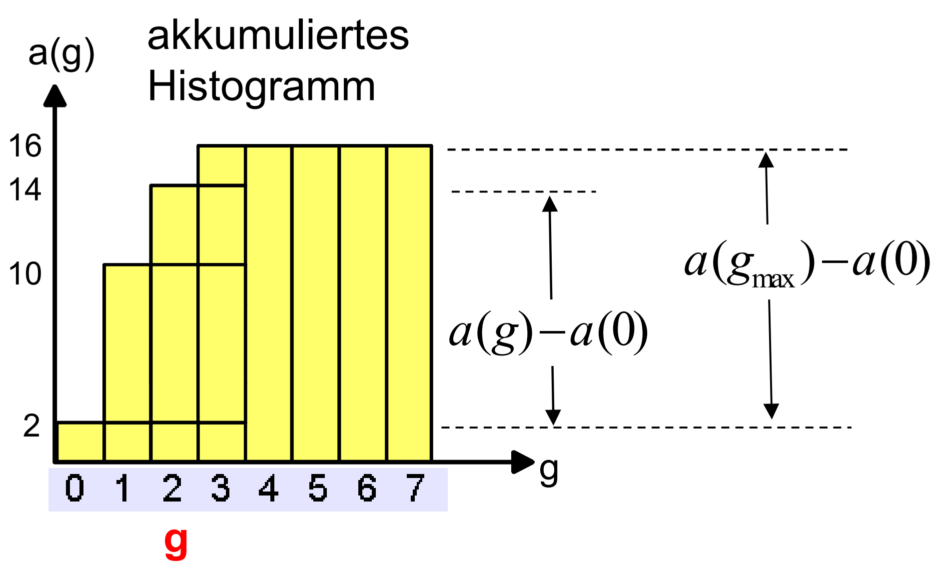
Schritt 2: Normierung des akkumulierten Histogramms (Wertebereich 0...1)

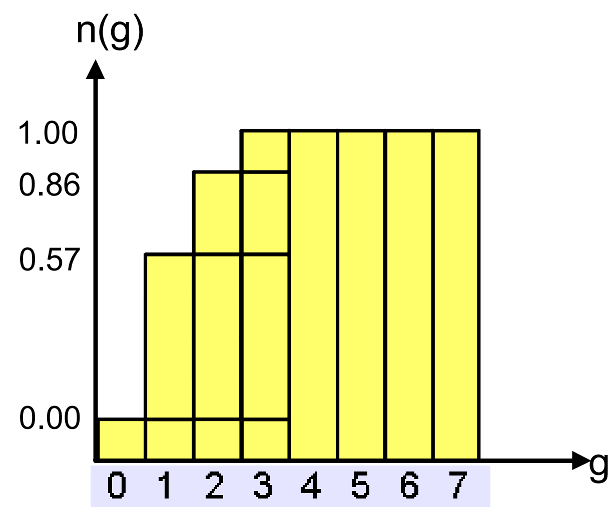
Durch die Normierung der Werte des akkumulierten Histogramms sind die Werte unabhängig von der Bildgröße.

$$n(g) = \frac{a(g) - a(0)}{a(g_{\max}) - a(0)}$$

Anm.: g_{Max} = max. möglicher Grauwert

$a(g_{\max})$ = Gesamtzahl der Bildpunkte

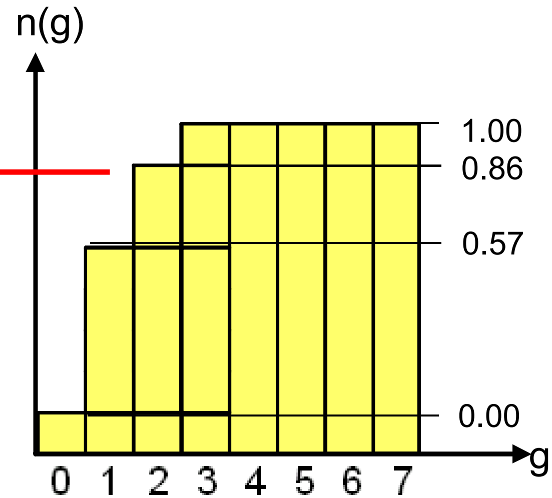




Schritt 3: Transformation der Quellgrauwerte auf die Zielgrauwerte

Mit Hilfe der Werte des normierten akkumulierten Histogramms werden die Grauwerte g des Quellbildes in die Grauwerte s des Zielbildes transformiert.

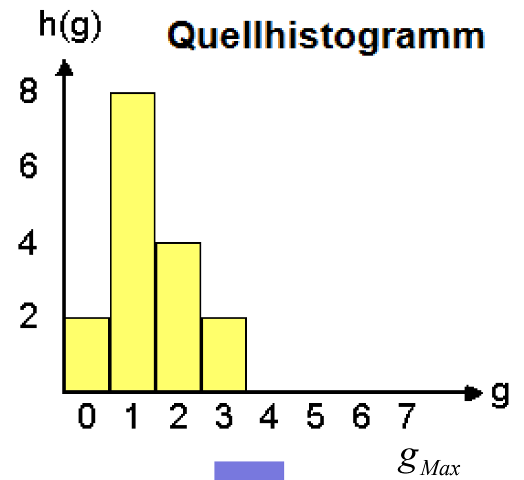
s ← $s = n(g) \cdot g_{max}$ ←

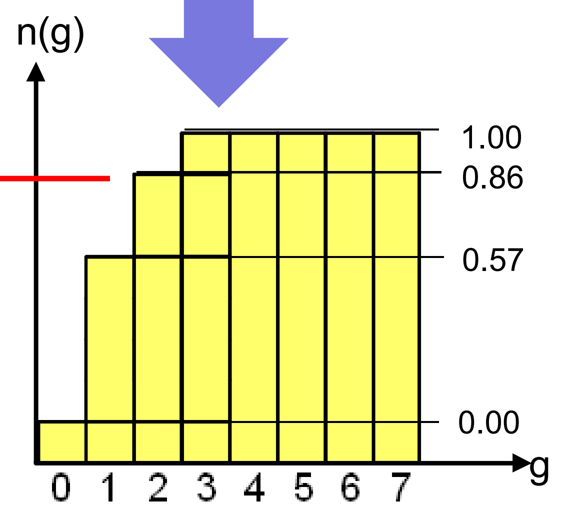


Quellgrauwert g

Zahlenbeispiel: Histogrammausgleich

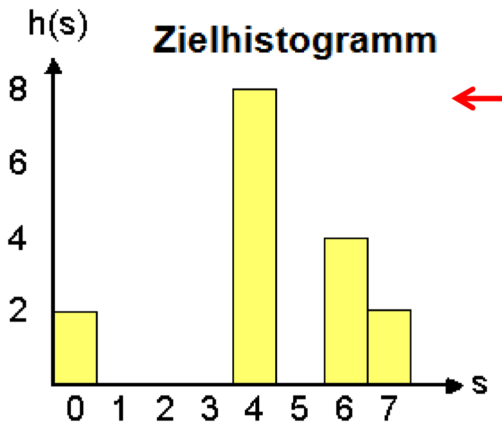
g	n(g)	s
0	0.00	0
1	0.57	4
2	0.86	6
3	1.00	$7 = g_{Max}$



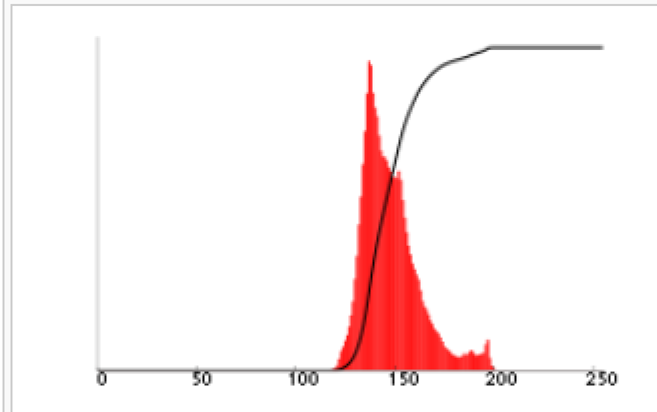


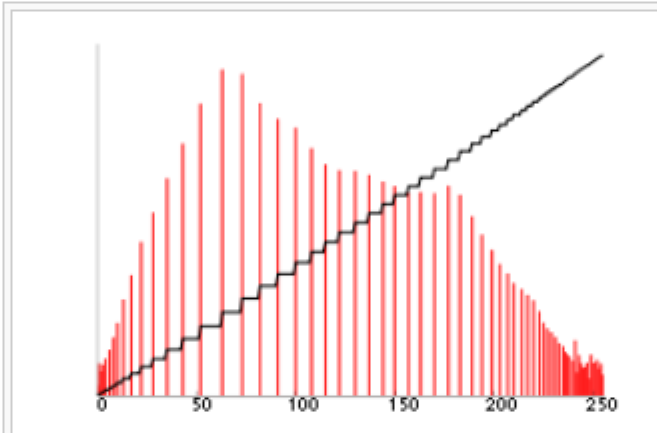
$$s = n(g) \cdot g_{\max}$$

$$g_{Max} = 7$$



Beispiel: Histogrammausgleich eines Landschaftsbildes





s. Wikipedia

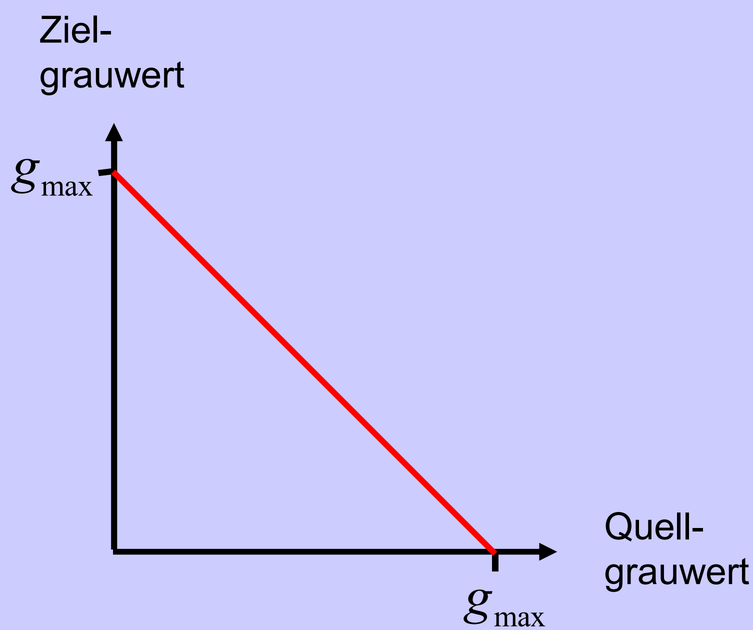
3.1.2 Grauwerttransformation mit Look-up-Tabellen

Grundgedanke: Es wird eine Tabelle erstellt und angewendet, mit deren Hilfe Quellgrauwerte auf Zielgrauwerte abbildet werden.

→ Vorteil: keine Berechnungen zur Laufzeit (sehr schnell)

Beispiel: Negieren des Bildes.

Quell-grauwert	Ziel-grauwert
0	15
1	14
2	13
.... u.s.w	....u.s.w
....	....
13	2
14	1
15	0



ÜBUNG: Operationen mit Look-up-Tabellen

Geben sei ein 4-bit-Grauwertbild
→ Graustufen 0….15.

Geben Sie die Look-up-Tabellen für folgende Operationen an:

a) Binarisierung des Bildes bei Grauwert 10.

b) Reduktion auf 4 Grauwerte (0, 5, 10, 15) in 4 äquidistanten Stufen.

c) Anhebung des Kontrastes im dunklen Grauwertbereich auf Kosten des hellen Bereiches.

d) Hervorheben von Bitebene 0 (= ungerade Grauwerte).

In	Out
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

In	Out
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

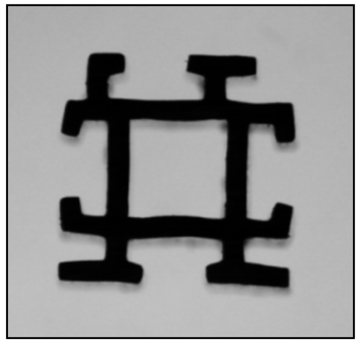
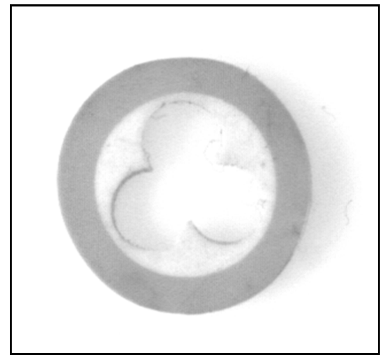
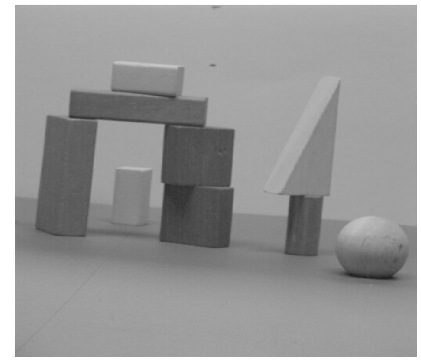
In	Out
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

In	Out
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

3.1.3 Thresholding = Binarisierung

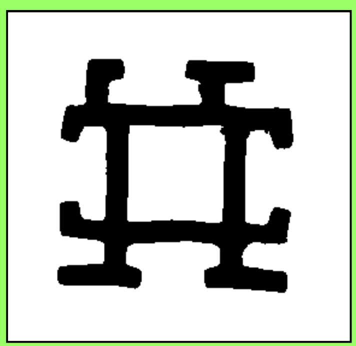
3.1.3.1 Grundgedanke

In manchen Anwendungsfällen werden Objekt und Hintergrund in unterschiedlichen Helligkeitsbereichen abgebildet.

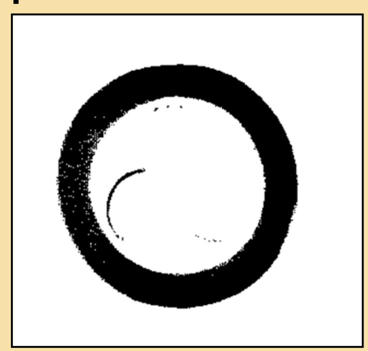
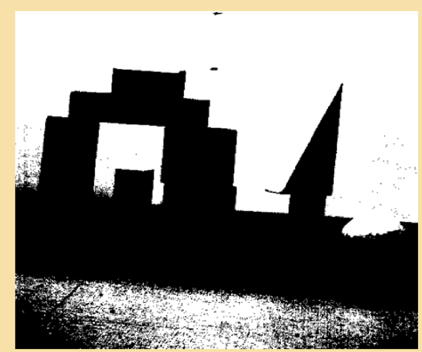




In solchen (einfachen) Fällen kann durch eine geeignete Grauwertschwelle Objekt und Hintergrund voneinander getrennt werden (→ *Segmentierung*).

möglich

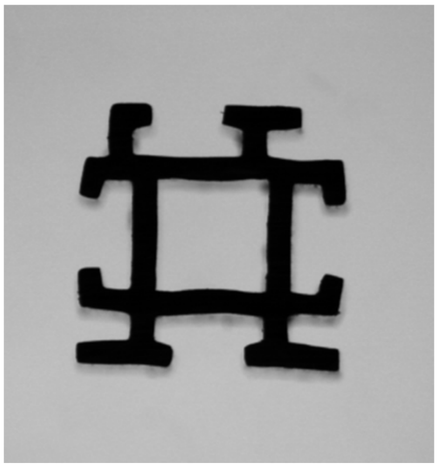
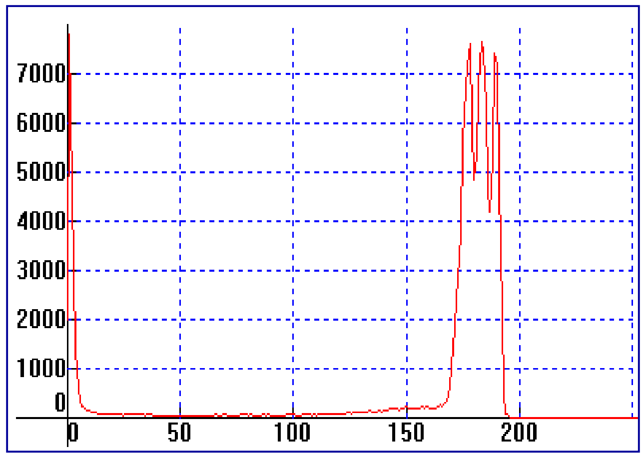


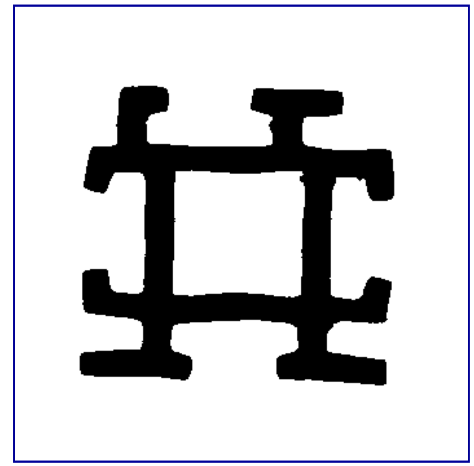
problematisch

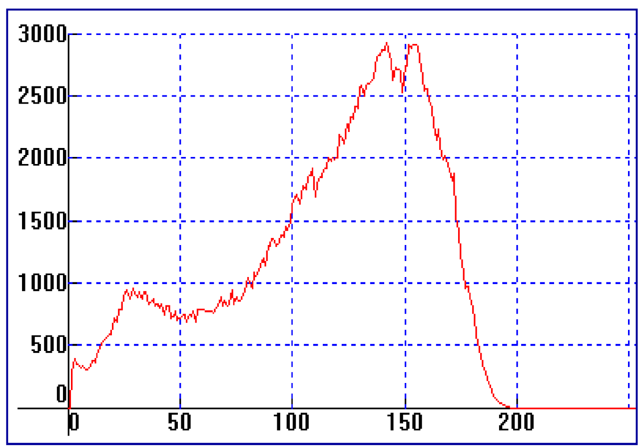



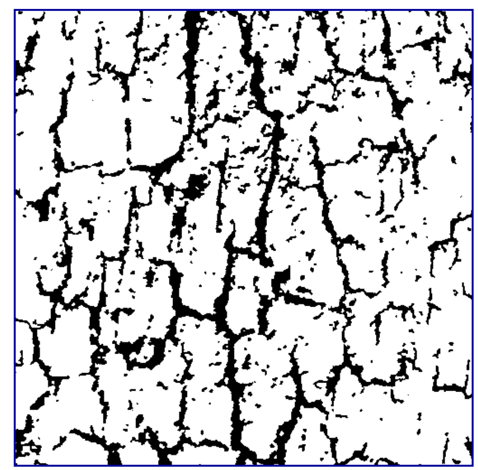
3.1.3.2 Automatische globale Schwellwertbestimmung

In einfachen Fällen ist das Histogramm „*bimodal*“, d.h. es besteht aus 2 signifikanten Maxima.

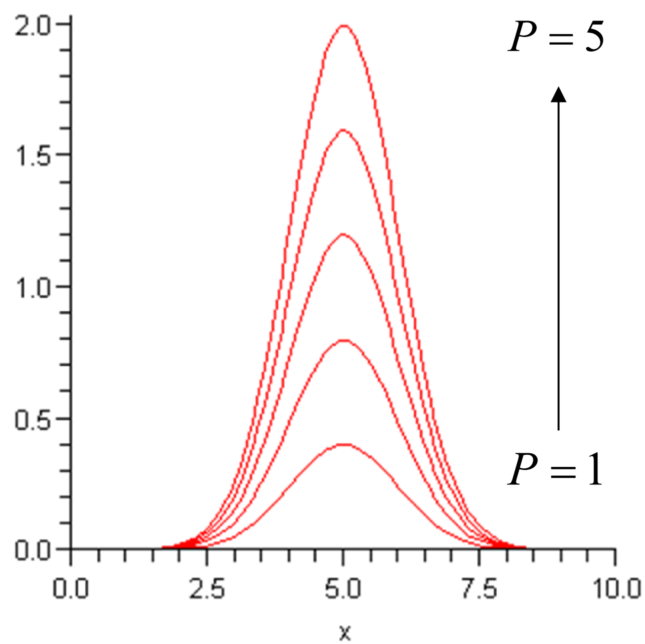




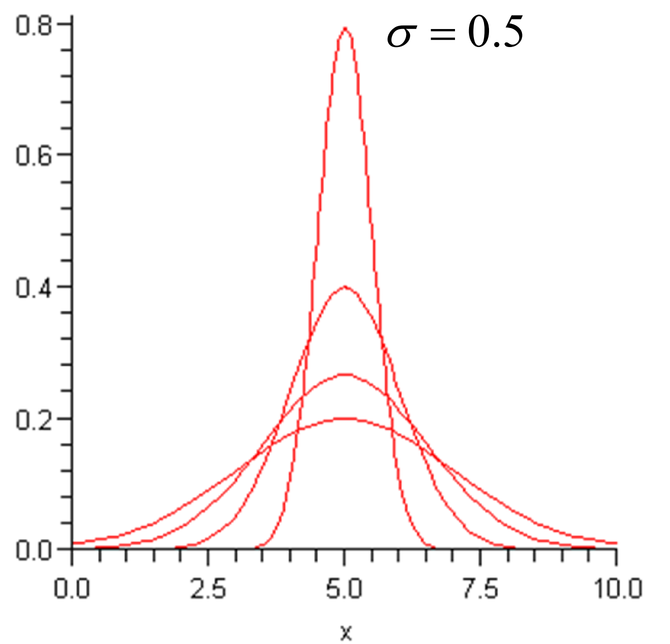


Kurze Erläuterung: Parameter der Gauss-Funktion

$$G(x) = \frac{P}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

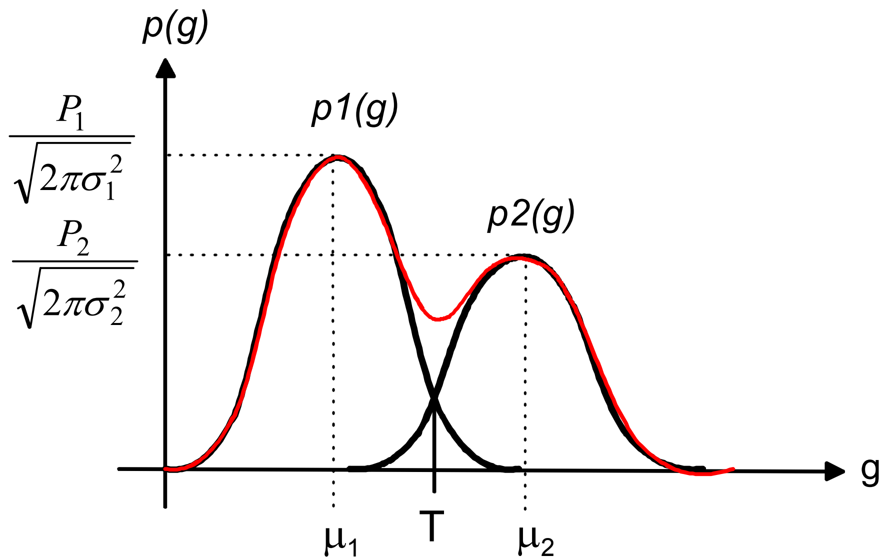


$P = 1..5, \quad \mu = 5, \quad \sigma = 1$



$P = 1, \quad \mu = 5, \quad \sigma = 0.5.....2.5$

Bestimmung der optimalen Binarisierungsschwelle T:



Annahmen :

- Es gibt 2 Häufungen $p_1(g)$, $p_2(g)$,
- die Häufungen sind Gauss-Funkt. mit μ: Mittelwert, σ: Varianz
- vereinfachend werden die Varianzen der beiden Häufungen als gleich angenommen → $\sigma_1 = \sigma_2 = \sigma$

→ Die optimale Schwelle g=T ist dort, wo gilt: $p_1(g) = p_2(g)$.

$$\underbrace{\frac{P_1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(g-\mu_1)^2}{2\sigma^2}}}_{p_1(g)} = \underbrace{\frac{P_2}{\sqrt{2\pi\sigma^2}} e^{-\frac{(g-\mu_2)^2}{2\sigma^2}}}_{p_2(g)}$$

⇒ nach g auflösen

Durch Umstellung nach g erhält man die optimale Schwelle T:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \cdot \ln\left(\frac{P_2}{P_1}\right)$$

Randbedingungen:

- zwei gaussförmige Häufungen
- mit gleichem σ
- Häufungen können unterschiedliche Höhe haben

Aber: Die Berechnung erfordert die Parameter μ_1, μ_2, P_1, P_2, und σ der beiden Häufungen. Woher?

In manchen Anwendungsfällen (Schriftstücke binarisieren, Bildmesstechnik, ...) können evtl. näherungsweise Annahmen zu P_1, P_2, und σ gemacht werden.

Wenn diese Möglichkeit nicht besteht, können die Parameter iterativ angenähert werden (s.u.).

Sonderfall 1: Für beide Häufungen gilt (näherungsweise)
a) gaussförmig,
b) gleiche P ($P_1=P_2$)

$$T = \frac{\mu_1 + \mu_2}{2}$$

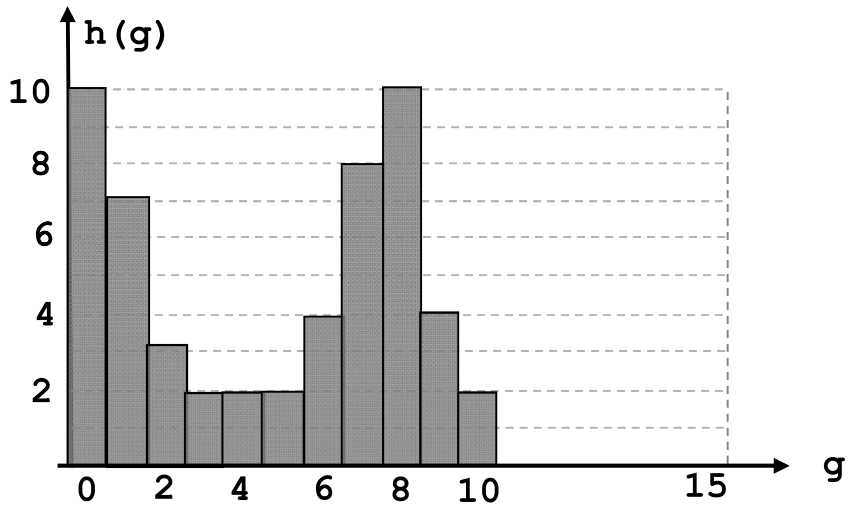
μ_1, μ_2 können dann iterativ mit folgendem Algorithmus berechnet werden (k-Means-Clustering):

1. **begin** Startwert für T (opt. Schwellwert) festlegen
2. **do** μ_1, μ_2 der Teilhistogramme berechnen
3. T daraus neu berechnen
4. **until** T ändert sich nicht mehr
5. **end**

$$\mu = \frac{\sum_{i=1}^{n} g_i \cdot h(g_i)}{\sum_{i=1}^{n} h(g_i)}$$

Beispiel: Adaptiven Schwellwert iterativ berechnen für Sonderfall 1

Gegeben ist das folgende Histogramm (Anm.: sehr kleines Bild):



$$P_1 \approx P_2$$

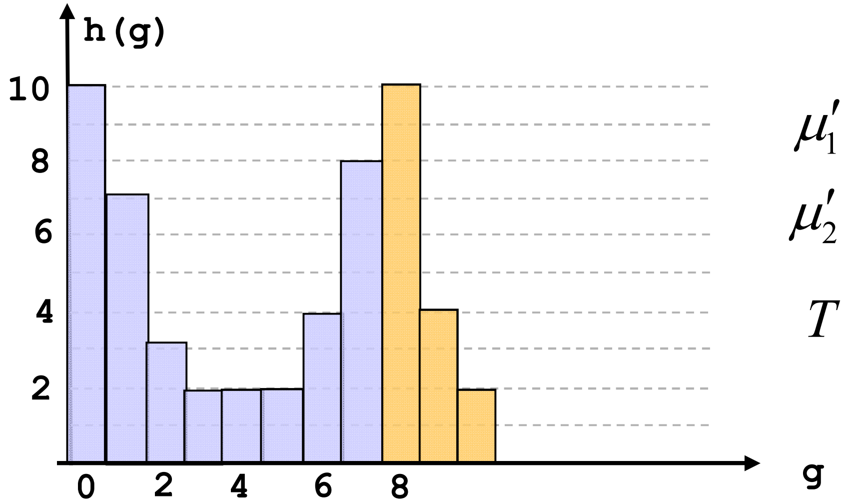
$$\boxed{T = \frac{\mu_1 + \mu_2}{2}}$$

$$\mu = \frac{\sum_{i=1}^{n} g_i \cdot h(g_i)}{\sum_{i=1}^{n} h(g_i)}$$

- Der Grauwertbereich geht von 0 … 15.
- Der Startwert T werde in der Mitte des Grauwertbereiches angenommen (T=8).

Startwert: $T = 8$

1. Iteration

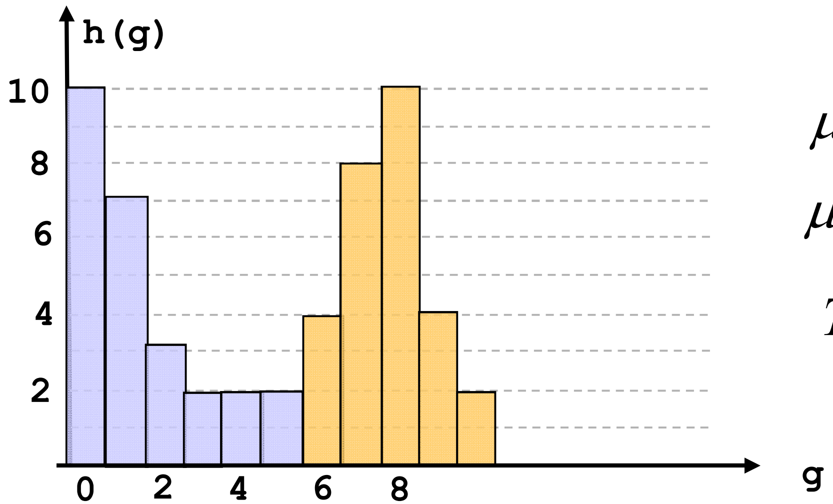


$$\mu_1' = 3.1$$

$$\mu_2' = 8.5$$

$$T' \approx 6$$

2. Iteration

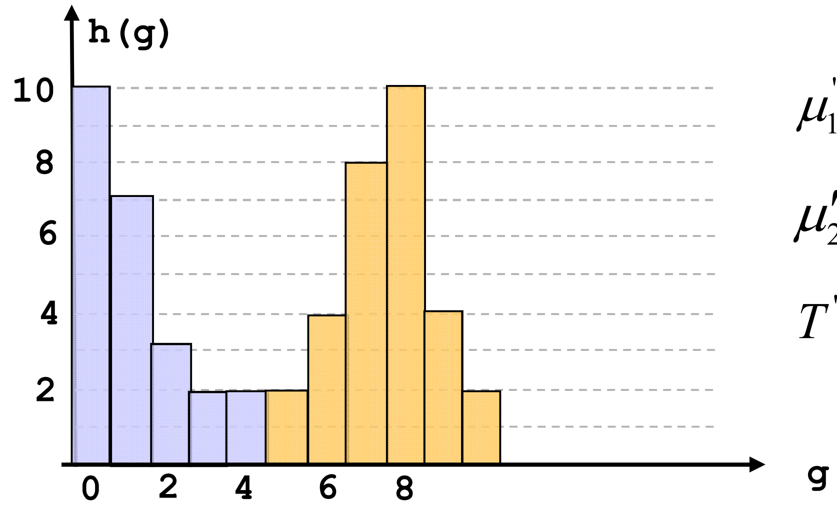


$$\mu_1'' = 1.4$$

$$\mu_2'' = 7.7$$

$$T'' \approx 5$$

3. Iteration

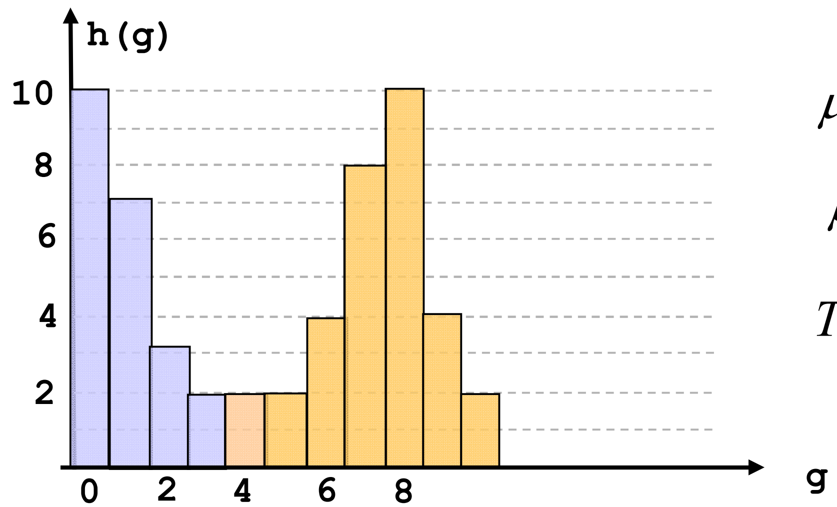


$\mu_1''' = 1.1$

$\mu_2''' = 7.5$

$T''' = 4$

4. Iteration



$\mu_1'''' = 0.9$

$\mu_2'''' = 7.3$

$T''' = 4$

$\boxed{T_{End} = 4}$

Sonderfall 2: Für beide Häufungen gilt (näherungsweise)
a) gaussförmig,
b) gleiche σ

Problem: Die Parameter P_1, P_2, σ, μ_1, μ_2 der beiden Verteilungen unbekannt.

Die Parameter können dann iterativ mit folgendem Algorithmus berechnet werden (k-Means-Clustering):

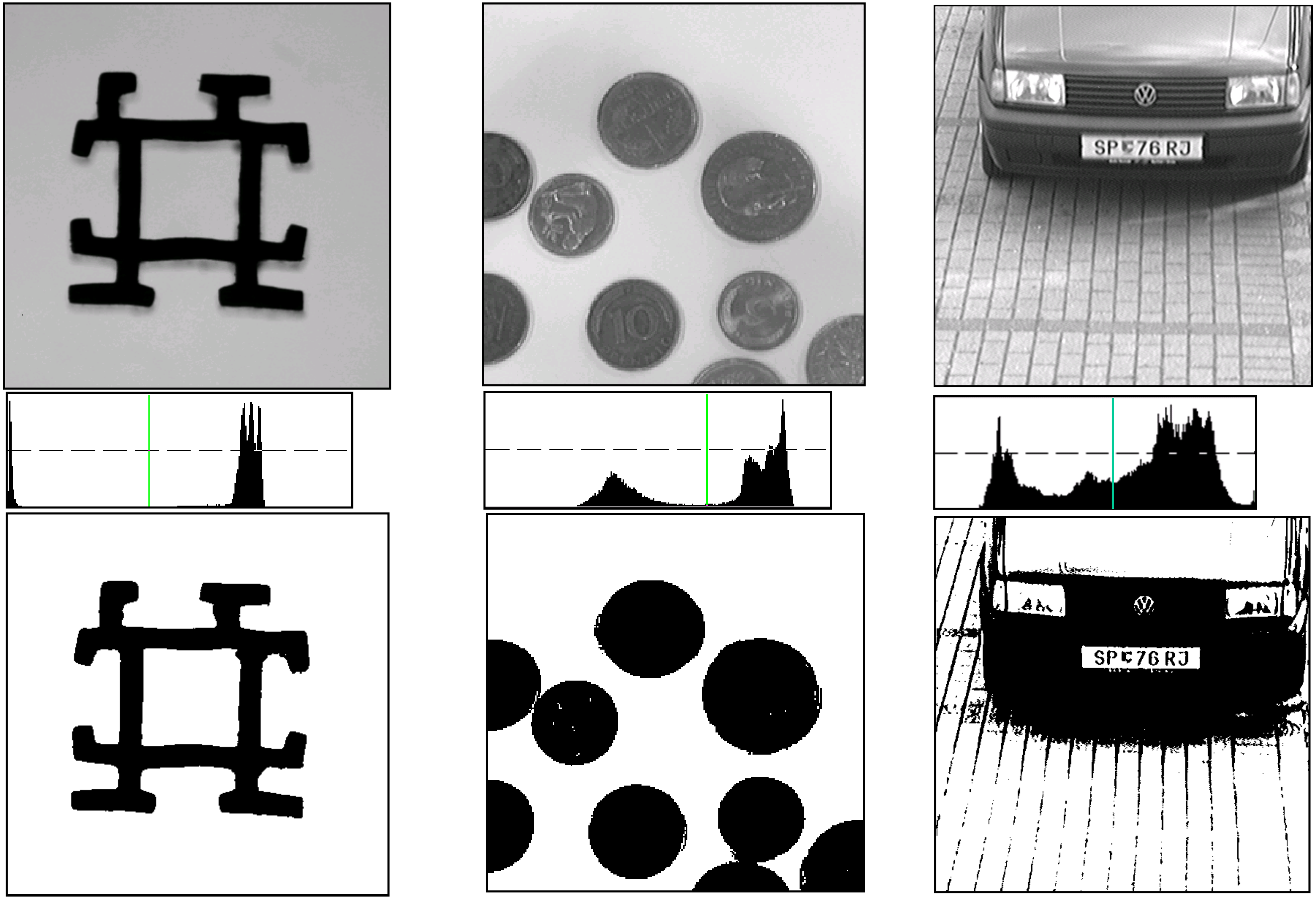
1. **begin** Startwert für T (opt. Schwellwert) festlegen
2. **do** μ_1, μ_2, σ, P_1, P_2 der Teilhistogramme berechnen
3. T daraus neu berechnen
4. **until** T ändert sich nicht mehr
5. **end**

$$\mu = \frac{\sum_{i=1}^{n} g_i \cdot h(g_i)}{\sum_{i=1}^{n} h(g_i)}$$

$$\sigma^2 = \frac{\sum_{i=1}^{n} (g_i - \mu)^2 h(g_i)}{\sum_{i=1}^{n} h(g_i)}$$

$$P = h(\mu) \cdot \sqrt{2\pi\sigma^2}$$

Beispiele: Adaptive Binarisierung



3.2 Arithmetisch-logische Bildpunktoperationen

3.2.1. Bildsubtraktion

$$f_{out}(x,y) = f_2(x,y) - f_1(x,y)$$

Zu beachten:

Das Ergebnis der Subtraktion kann auch negativ sein. Abhängig vom Anwendungsfall muss damit geeignet umgegangen werden, z.B. :

a) $f_{out}(x,y) = | f_2(x,y) - f_1(x,y) |$

b) $f_{out}(x,y) = f_2(x,y) - f_1(x,y) \quad wenn\, Erg. > 0$

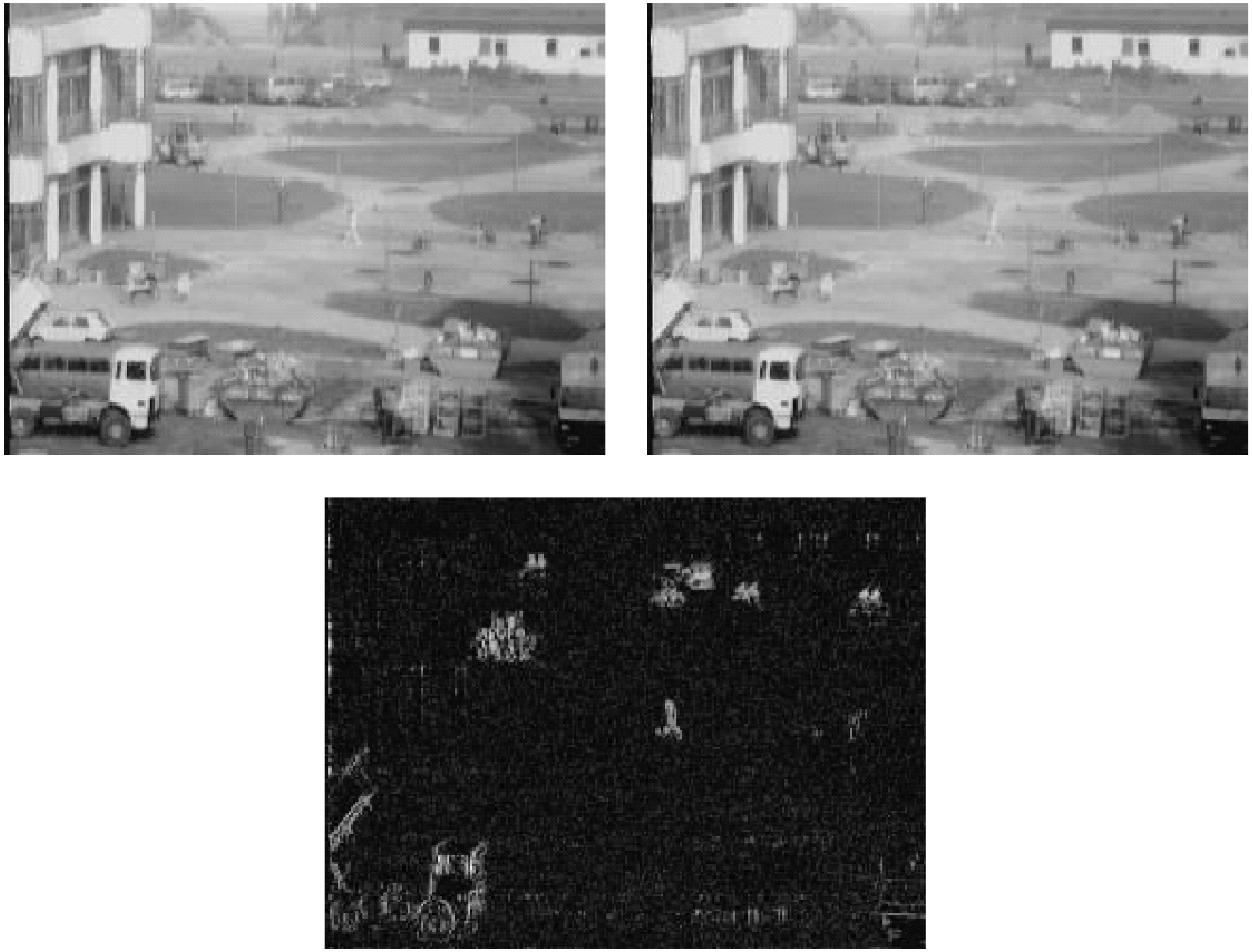
$\quad = 0 \quad sonst$

Anwendungsbeispiele:

- Hintergrundkompensation
- Vignettierungskorrektur
- Änderungsdetektion (z.B. Bewegungsmelder)

Beispiel : Bewegungsanalyse durch Differenzbildung einer Bildsequenz

aus B. Jähne: Digitale Bildverarbeitung



3.2.2 Averaging (Mittelung über mehrere Bilder)

$$\bar{f}(x,y) = \frac{1}{K}\sum_{i=1}^{K} f_i(x,y)$$

von http://www.djcash.demon.co.uk/astro/webcam/HowItsDone.htm

Anwendungsbeispiele:

- Rauschunterdrückung in Bildern
- Kompensation thermischer Schwankungen, z.B. astronom. Bilder („*stacking*“)

3.2.3 Bitweise logische Operationen zwischen Bildern

$$f_{out}(x,y) = f_2(x,y)\ AND\ \ f_1(x,y)$$

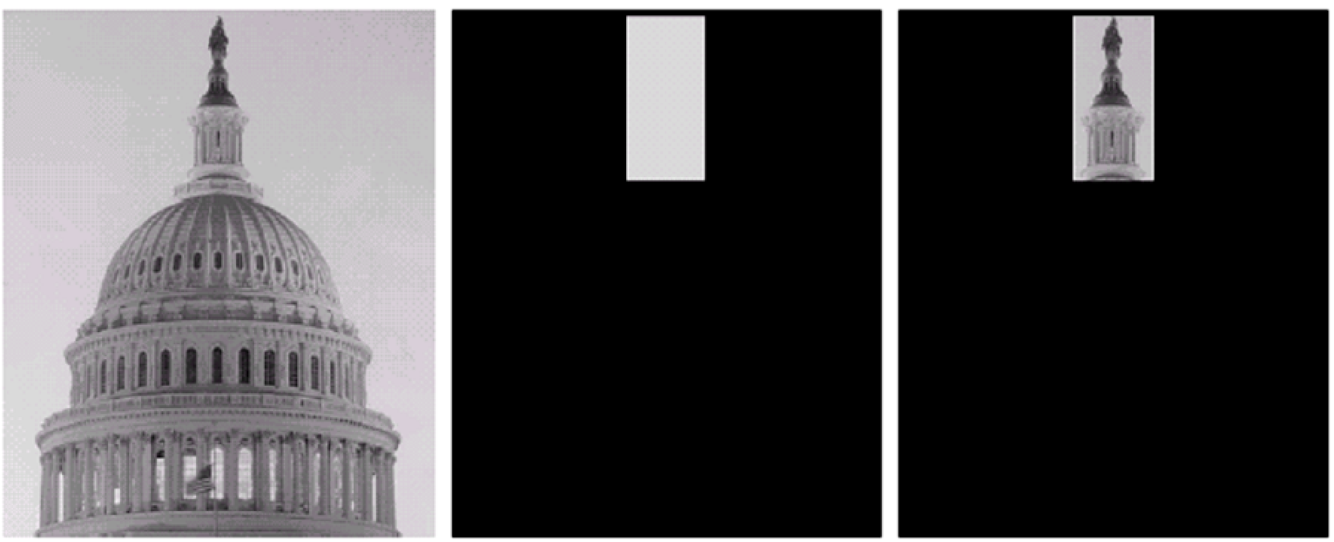
$$f_{out}(x,y) = f_2(x,y)\ OR\ \ f_1(x,y)$$

$$f_{out}(x,y) = f_2(x,y)\ XOR\ \ f_1(x,y)$$

Anm.: bitweise AND / OR / XOR

Anwendungsbeispiele:

- Ausmaskieren von Bildbereichen (AND)
- Überlagern von Bildern, z.B. Einblenden von Rastern, Fadenkreuzen ((X)OR)



aus "Digital Image Processing", Gonzalez/Woods

ÜBUNG: Logische Operationen mit und zwischen Bildern

Wie lassen sich mit log. Operationen folgende Funktionen realisieren?

a) Graustufenreduktion auf 2, 4, 8 und 16 Graustufen (s.u.).

b) Ein Binärbild (0, 255) in ein Grauwertbild einblenden (s.u.) .

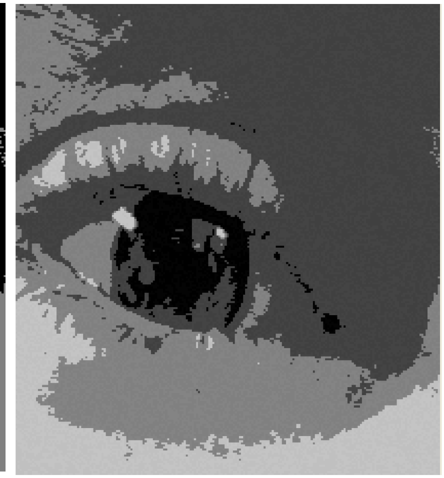
c) Nur die Teile eines Bildes anzeigen, die in einem Binärbild (*Maskierung*) den Wert 255 haben (s.u.).

d) Die linke Hälfte von Bild 1 und die rechte Hälfte von Bild 2 zwei in einem Bild darstellen (s.u.).

Beispielbilder zur Übung

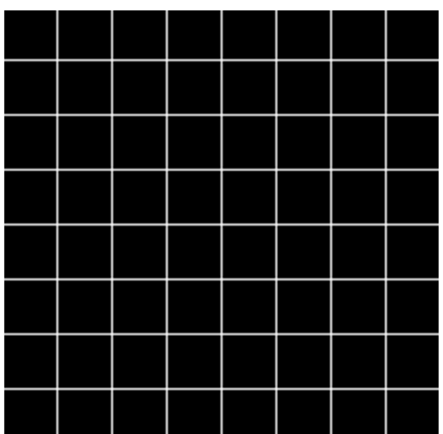
Originalbild (8-bit)

AND 10000000



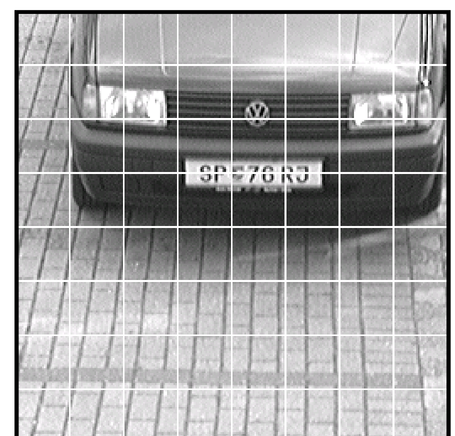
AND 11000000

AND 11100000



OR

=



Beispielbilder zur Übung