T-61.5100 DIGITAALINEN KUVANKÄSITTELY/ DIGITAL IMAGE PROCESSING

Laskuharjoitukset/Exercises 2007

Contents

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 1/07

1. Oletetaan, että tasaista (x_0, y_0) -keskipisteistä aluetta valaistaan valonlähteellä, jonka intensiteettijakauma on

$$i(x, y) = Ke^{-[(x-x_0)^2+(y-y_0)^2]}.$$

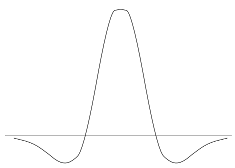
Alueen heijastavuus $r(x, y)$ on 1 ja $K = 255$. Jos kuva digitoidaan n bitin intensiteettiresoluutiolla, ja silmä pystyy havainnoimaan kahdeksan intensiteettitason suuruisen äkillisen muutoksen, millä n:n arvolla syntyy väärä ääriviiva (false contouring)?

2. Datasiirrossa mitataan siirretyn informaation määrää bitteinä sekunnissa (bit/s). Yleensä siirto suoritetaan paketteina, jotka koostuvat aloitusbitistä (start bit), tavusta (8 bittiä) informaatiota sekä lopetusbitistä (stop bit).

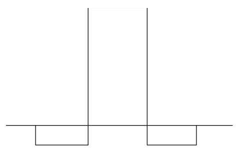
   (a) Kuinka monta minuuttia kuluu siirrettäessä 9600 bit/s nopeudella 512×512 -kokoinen kuva, jossa on 256 harmaatasoa?

   (b) Kuinka suuri pitäisi tarvittavan digitaalisen siirtokanavan kapasiteetin olla, jos halutaan välittää a-kohdan laatuista digitaalista TV-kuvaa (25 kuvaa/s) reaaliajassa?

3. Monissa kuva–analyysitehtävissä voidaan menestyksellisesti soveltaa meksikolaishattufunktiota (kuva 1a). Approksimoidaan tätä funktiota kuvan 1b funktiolla. Esitä tämän funktion avulla Machin nauhat yhdessä dimensiossa. Miten kävisi kahdessa dimensiossa mustalle pallolle valkoisella taustalla?



Kuva 1a.



Kuva 1b.

4. Käsitellään alla olevaa kuvasegmenttiä.

   (a) Olkoon $V = \{0, 1\}$. Laske D_4-, D_8- ja D_m-etäisyydet p:n ja q:n välillä.

   (b) Tee sama, kun $V = \{1, 2\}$.

	3	1	2	[1]	$\leftarrow q$
	2	2	0	2	
	1	2	1	1	
$p \rightarrow$	[1]	0	1	2	

5. Käsitellään alla olevia alikuvia S_1 ja S_2. Olkoon $V = \{1\}$. Kuinka monesta

   (a) 4-yhtenäisestä (4-connected)

   (b) 8-yhtenäisestä

   (c) m-yhtenäisestä

komponentista S_1 ja S_2 koostuvat? Ovatko S_1 ja S_2 vierekkäisiä?

	S_1				S_2				
0	0	0	0	0	0	0	1	1	0
1	0	0	1	0	0	1	0	0	1
1	0	0	1	0	1	1	0	0	0
0	0	1	1	1	0	0	1	1	1
0	0	1	1	1	0	0	1	1	1

T-61.5100 Digital image processing, Exercise 1/07

1. Suppose that a flat area with center at (x_0, y_0) is illuminated by a light source with intensity distribution

$$i(x, y) = Ke^{-[(x-x_0)^2+(y-y_0)^2]}.$$

The reflectance $r(x, y)$ of the area is 1 and $K = 255$. If the resulting image is digitized using n bits of intensity resolution, and the eye can detect an abrupt change of eight shades of intensity between adjacent pixels, what value of n will cause visible false contouring?

2. A common measure of transmission for digital data is the number of bits transmitted per second (*bit/s*). Generally, transmission is accomplished in packets consisting of a start bit, a byte (8 bits) of information, and a stop bit. Using this approximation, answer the following:

   (a) How many minutes would it take to transmit a 512×512 image with 256 gray levels at 9600 bit/s?

   (b) What should be the capacity (bit/s) of a digital transfer channel, if images described in item a. (25 images/second) are to be transferred in real time?

3. In many image analysis problems a 'Mexican-hat' function can be successfully applied (figure 1a). This function is approximated using the function in figure 1b. Demostrate Mach bands in one dimension using this function. What would happen to a black ball on white background in two dimensions?

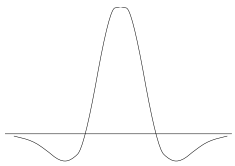


Fig. 1a.

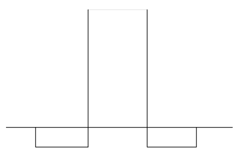


Fig 1b.

4. Consider the image segment shown below.

   (a) Let $V = \{0, 1\}$ and compute D_4-, D_8- ja D_m-distances between p and q.

   (b) Repeat for $V = \{1, 2\}$.

	3	1	2	[1]	$\leftarrow q$
	2	2	0	2	
	1	2	1	1	
$p \rightarrow$	[1]	0	1	2	

5. Consider the two image subsets S_1 and S_2 shown below. For $V = \{1\}$, determine how many

   (a) 4-connected

   (b) 8-connected

   (c) m-connected

components there are in S_1 and S_2. Are S_1 and S_2 adjacent?

	S_1				S_2				
0	0	0	0	0	0	0	1	1	0
1	0	0	1	0	0	1	0	0	1
1	0	0	1	0	1	1	0	0	0
0	0	1	1	1	0	0	1	1	1
0	0	1	1	1	0	0	1	1	1

T-61.5100 Digital image processing, Exercise 1/07

1.

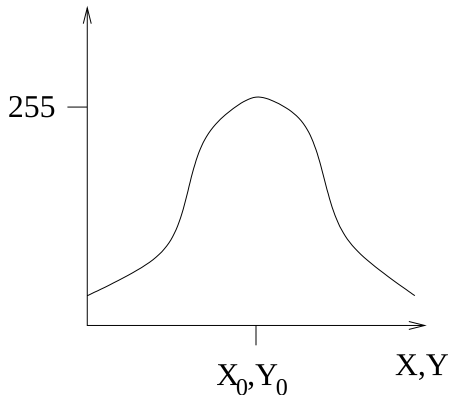
The intensity distribution is now

$$i(x,y) = 255e^{-[(x-x_0)^2+(y-y0)^2]}, 0 < i(x,y) < \infty \tag{1}$$

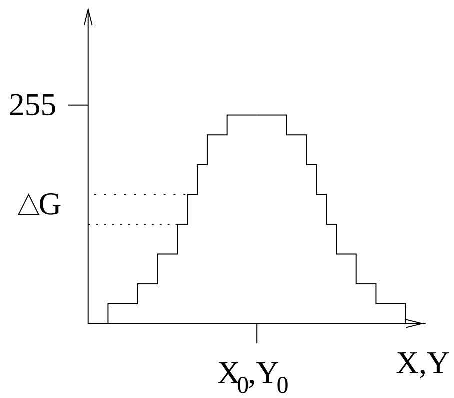
and the reflectance $r(x,y) = 1$.
The image is the product of these:

$$f(x,y) = i(x,y)r(x,y) = 255e^{-[(x-x_0)^2+(y-y0)^2]}, \tag{2}$$

and the crosssection of the image looks like this



Quantisation with n bits means that we discretise the intensity so that the smallest change is $\Delta G = \frac{255+1}{2^n}$. This is what the quantised picture looks like:



Since the eye is able to perceive a sudden change of 8 intensity levels, we see a false contour, when $\Delta G = 8$. Thus

$$8 \leq \frac{255+1}{2^n} \Leftrightarrow n \leq 5 \tag{3}$$

2.

The package consists of ten bits: a start bit, 8 bits of information (one byte) and a stop bit.

a) 9600 bit/s means that we have 960 10-bit packages in a second, i.e. 960 bytes of information in a second. 512×512 sized 8-bit image takes $512 \cdot 512 \cdot 1 = 262144$ bytes. Thus

$$\text{transfer time} = \frac{262144}{960}\ s \approx 4.55 \text{ min}$$

b) The capacity should be at least

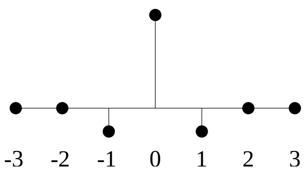
$$25 \text{ images/s} \cdot 512 \cdot 512 \text{ pixels/image} \cdot 10 \text{ bits/pixel} = 65,536,000 \text{ bit/s} = 62.5\,\text{Mib/s}$$

Where Mib means "binary" megabits, i.e. 10^{20} bits.

3.

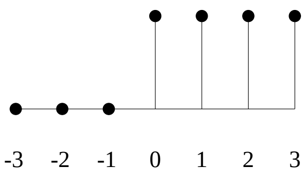
Some features of the visual system can be modeled roughly by a Mexican hat -function. For this problem we use the simplest possible hat:

$$h(i) = \begin{cases} -0.25, & \text{when } i = \pm 1 \\ 1, & \text{when } i = 0 \\ 0, & \text{otherwise} \end{cases}$$



An edge is modelled by a step function:

$$f(i) = \begin{cases} 0, & \text{when } i < 0 \\ 1, & \text{otherwise} \end{cases}$$

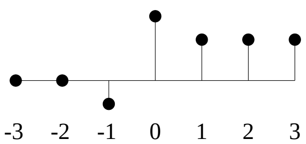


Convolution is used:

$$g(x) = \sum_{i=-\infty}^{\infty} h(i) f(x-i) = \sum_{i=-1}^{1} h(i) f(x-i)$$

And we have

$$g(-3) = 0. \quad g(-2) = 0, \quad g(-1) = -0.25, \quad g(0) = 0.75, \quad g(1) = 0.5, \quad g(2) = 0.5$$



Thus, we see a darker and a lighter bands near the edge. In two dimensions these would be seen as a moat and a wall.

4.

Distance measures:
D_4: "City-block" distance.
D_8: "Chess-board" distance.
D_m: "City-block", if possible, otherwise "Chess-board".

Note! D_4 and D_8 do not depend on V (as defined in Gonzalez-Woods)!

a) $D_4(p,q) = |x_p - x_q| + |y_p - y_q| = 3 + 3 = 6$
$D_8(p,q) = \max(|x_p - x_q|, |y_p - y_q|) = \max(3,3) = 3$
$D_m(p,q) = 5$

$$\begin{array}{ccccccc}
3 & & 1 & & 2 & & \boxed{1} \\
 & & & & & \nearrow & \\
2 & & 2 & & 0 & & 2 \\
 & & & & \uparrow & & \\
1 & & 2 & & 1 & & 1 \\
 & & & & \uparrow & & \\
\boxed{1} & \rightarrow & 0 & \rightarrow & 1 & & 2
\end{array}$$

b) D_4 and D_8 are independent of V, so they are the same as in a).
$D_m(p,q) = 6$

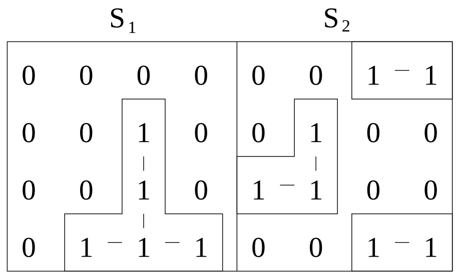
$$\begin{array}{ccccccc}
3 & & 1 & \rightarrow & 2 & \rightarrow & \boxed{1} \\
 & & \uparrow & & & & \\
2 & & 2 & & 0 & & 2 \\
 & & \uparrow & & & & \\
1 & \rightarrow & 2 & & 1 & & 1 \\
\uparrow & & & & & & \\
\boxed{1} & & 0 & & 1 & & 2
\end{array}$$

Of course, this is not the only possible path with length 6.

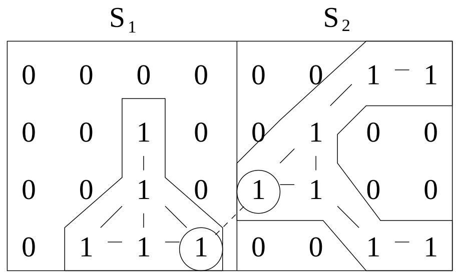
5.

Connectivity: “Are there certain pixels in the neigbourhood?”
Adjacency: “Are certain pixels adjacent?”

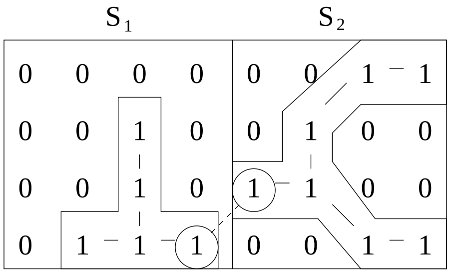
a) 4-connected components:
S_1: 1, S_2: 3



b) 8-connected components:
S_1: 1, S_2: 1



c) M-connected components:
S_1: 1, S_2: 1



The answer is the same as in b), but now there exist only one possible path connecting the pixels.

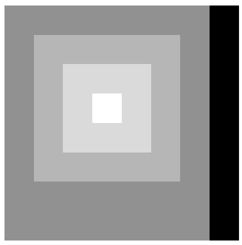
Adjacency:

a): S_1 and S_2 are not adjacent, since no pixel of S_2 that belong to V is a 4-neighour of any pixel in S_1 that belong to V.
b) and c): In both the cases S_1 and S_2 are adjacent, thanks to the pixels that have been circled in the figures.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 2/07

Spatiaalitason menetelmät kuvien parantamisessa

1. Tasoita alla olevan harmaatasoista $0, 1, \ldots, 7$ koostuvan 8×8 –kuvan histogrammi.



4	4	4	4	4	4	4	0
4	5	5	5	5	5	4	0
4	5	6	6	6	5	4	0
4	5	6	7	6	5	4	0
4	5	6	6	6	5	4	0
4	5	5	5	5	5	4	0
4	4	4	4	4	4	4	0
4	4	4	4	4	4	4	0

2. Oletetaan, että meillä on useita eri kohinaisia versioita $g_i(x, y)$ samasta kuvasta $f(x, y)$, eli
$$g_i(x, y) = f(x, y) + \eta_i(x, y),$$
missä kohina $\eta_i(x, y)$ on nollakeskiarvoista ja kaikki pisteparit ovat korreloimattomia. Silloin voidaan vähentää kohinaa ottamalla keskiarvo kaikista kuvista, eli
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^{M} g_i(x, y).$$
Todista, että pätee
$$E\{\bar{g}(x, y)\} = f(x, y)$$
ja
$$\sigma^2_{\bar{g}(x,y)} = \frac{1}{M} \sigma^2_{\eta(x,y)}$$
missä $\sigma^2_{\eta(x,y)}$ on $\eta(x, y)$:n varianssi ja $\sigma^2_{\bar{g}(x,y)}$ vastaavasti $\bar{g}(x, y)$:n varianssi.

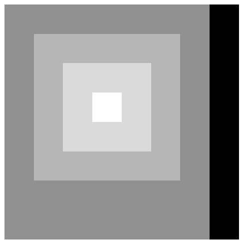
3. Kaksimuuttujaisen jatkuvan funktion Laplace-operaattori on
$$\bigtriangledown^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}.$$
Tästä seuraa, että
$$\mathcal{F}\{\bigtriangledown^2 f(x, y)\} = -(2\pi)^2(u^2 + v^2)F(u, v),$$
missä $F(u, v) = \mathcal{F}\{f(x, y)\}$. Määritä vastaava operaattori ja Fourier-muunnos diskreetissä tapauksessa. Vertaa saatua tulosta jatkuvan tapauksen tulokseen.

4. Näytä, että Laplace-muunnetun kuvan vähentäminen alkuperäisestä kuvasta on verrannollinen *unsharp masking* -operaatioon. Käytä diskreettiä Laplace-muunnosta.

T-61.5100 Digital image processing, Exercise 2/07

Image enhancement in the spatial domain

1. Equalize the histogram of the 8×8 image below. The image has grey levels $0, 1, \ldots, 7$.



$$\begin{matrix} 4 & 4 & 4 & 4 & 4 & 4 & 4 & 0 \\ 4 & 5 & 5 & 5 & 5 & 5 & 4 & 0 \\ 4 & 5 & 6 & 6 & 6 & 5 & 4 & 0 \\ 4 & 5 & 6 & 7 & 6 & 5 & 4 & 0 \\ 4 & 5 & 6 & 6 & 6 & 5 & 4 & 0 \\ 4 & 5 & 5 & 5 & 5 & 5 & 4 & 0 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 0 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 0 \end{matrix}$$

2. Assume that we have many noisy versions $g_i(x, y)$ of the same image $f(x, y)$, i.e.

$$g_i(x, y) = f(x, y) + \eta_i(x, y)$$

where the noise η_i is zero-mean and all point-pairs (x, y) are uncorrelated. Then we can reduce noise by taking the mean of all the noisy images

$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^{M} g_i(x, y).$$

Prove that

$$E\{\bar{g}(x, y)\} = f(x, y)$$

and

$$\sigma^2_{\bar{g}(x,y)} = \frac{1}{M} \sigma^2_{\eta(x,y)}$$

where $\sigma^2_{\eta(x,y)}$ is the variance of η and $\sigma^2_{\bar{g}(x,y)}$ the variance of $\bar{g}(x, y)$.

3. A two-variable continuous function's Laplace-operator is

$$\bigtriangledown^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}.$$

From this, it follows that

$$\mathcal{F}\{\bigtriangledown^2 f(x, y)\} = -(2\pi)^2(u^2 + v^2)F(u, v),$$

where $F(u, v) = \mathcal{F}\{f(x, y)\}$. Determine the corresponding operator and the Fourier transform in the discrete case. Compare the result obtained to the continuous case.

4. Show that subtracting the Laplacian from an image is proportional to unsharp masking. Use the definition for the Laplacian in the discrete case.

T-61.5100 Digital image processing, Exercise 2/07

1.

Histogram equalization spreads the histogram of an image so that it will span a wider range of gray scale values. This usually results in enhancing the contrast, which is often useful to improve the visibility of details in images. The discrete equalization is given by

$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n},$$

where r_k is the normalized gray level, n_k is the number of pixels having gray level k, and n is the total number of pixels.

First we calculate the gray level histogram:

k	0	1	2	3	4	5	6	7
r_k	0	0.143	0.286	0.429	0.571	0.714	0.857	1.0
n_k	8	0	0	0	31	16	8	1
$p_r(r_k) = n_k/n$	0.125	0	0	0	0.484	0.250	0.125	0.016

Then we form the cumulative distribution function:

s_k	0.125	0.125	0.125	0.125	0.609	0.859	0.984	1.000

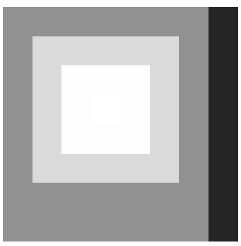
Finally we round the obtained values for the actual gray levels r_k:

s'_k	0.143	0.143	0.143	0.143	0.571	0.857	1.000	1.000

So, the histogram equalization is done by mapping the gray levels k into the new gray levels k':

k	0	1	2	3	4	5	6	7
k'	1	1	1	1	4	6	7	7

Resulting image:



4	4	4	4	4	4	4	1
4	6	6	6	6	6	4	1
4	6	7	7	7	6	4	1
4	6	7	7	7	6	4	1
4	6	7	7	7	6	4	1
4	6	6	6	6	6	4	1
4	4	4	4	4	4	4	1
4	4	4	4	4	4	4	1

NOTE: This solution is the “official” one for the course, however, it has some problems. For instance if one tries to equalize an image where the histogram is already fully equalized (i.e. 8 pixels for each level) the resulting histogram will not remain the same. Alternative solutions and discussion were presented in exercise session 4.

2.

The noisy image is now given as

$$\underbrace{g_i(x,y)}_{\text{noisy image}} = \underbrace{f(x,y)}_{\text{original image}} + \underbrace{\eta_i(x,y)}_{\text{noise}},$$

where noise has zero mean and it is uncorrelated.
When we average the noisy images

$$\bar{g}(x,y) = \frac{1}{M}\sum_{i=1}^{M} g_i(x,y)$$

we get for the expectation

$$E\{\bar{g}(x,y)\} = E\left\{\frac{1}{M}\sum_{i=1}^{M} g_i(x,y)\right\} = E\left\{\frac{1}{M}\sum_{i=1}^{M} f(x,y) + \frac{1}{M}\sum_{i=1}^{M}\eta_i(x,y)\right\} =$$
$$\frac{1}{M}\sum_{i=1}^{M} E\{f(x,y)\} + \frac{1}{M}\sum_{i=1}^{M}\underbrace{E\{\eta_i(x,y)\}}_{=0 \text{ (zero mean)}} = f(x,y)$$

and for the variance

$$\sigma^2_{\bar{g}(x,y)} = E\left\{(\bar{g} - E\{\bar{g}\})^2\right\} = E\left\{\left(\frac{1}{M}\sum_{i=1}^{M}(f+\eta_i) - f\right)^2\right\} = E\left\{\left(\frac{1}{M}\sum_{i=1}^{M}\eta_i\right)^2\right\} =$$
$$\frac{1}{M^2}E\left\{\left(\sum_{i=1}^{M}\eta_i\right)^2\right\} = \frac{1}{M^2}E\left\{\sum_{i=1}^{M}\left(\eta_i^2 + \sum_{j=1,\, j\neq i}^{M}\eta_i\eta_j\right)\right\} = \frac{1}{M^2}\left[\sum_{i=1}^{M}\left(\underbrace{E\{\eta_i^2\}}_{=\sigma_\eta^2} + \sum_{j=1,\, j\neq i}^{M}\underbrace{E\{\eta_i\eta_j\}}_{=0}\right)\right]$$
$$= \frac{1}{M}\sigma_\eta^2,$$

since the noise had zero mean and it was uncorrelated.

3.

We approximate the derivative in discrete case:

$$\frac{\partial f(x,y)}{\partial x} \simeq f(x,y) - f(x-1,y)$$

By 'derivating' again, we have

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

This is equivalent to filtering with a mask $[1,-2,1]$. The mask is most practically used symmetrically so that -2 is in the middle, rather than as in the formula above.
Thus

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \simeq f(x-1,y) + f(x+1,y) + f(x,y-1) + f(x,y+1) - 4f(x,y),$$

which corresponds to mask $h(x, y)$:

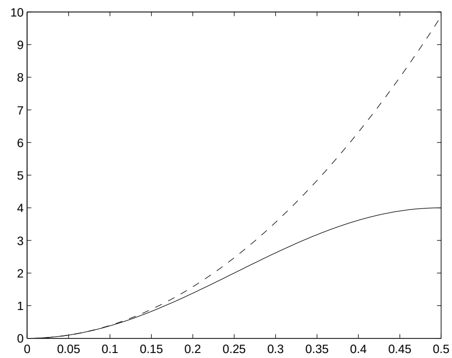
	1	
1	−4	1
	1	

Next we calculate the Fourier transform of the mask:

$$H(u,v) = \frac{1}{N}\sum_{x=-\infty}^{\infty}\sum_{y=-\infty}^{\infty} h(x,y)e^{-j2\pi(ux+vy)/N} = \frac{1}{N}(e^{j2\pi u/N} + e^{j2\pi v/N} + e^{-j2\pi u/N} + e^{-j2\pi v/N} - 4)$$
$$= \frac{1}{N}\left[2\cos(2\pi u/N) + 2\cos(2\pi v/N) - 4\right]$$

And finally

$$\mathcal{F}\{\bigtriangledown^2 f[x,y]\} = \frac{1}{N}(2\cos(2\pi u/N) + 2\cos(2\pi v/N) - 4)\cdot F(u,v),$$



In the figure there is $|H(u,0)|$ in discrete and continuous case. Solid line: discrete case, i.e. $|H(u,0)| = 2 - 2\cos(2\pi u)$. Dotted line: continuous case, i.e. $|H(u,0)| = (2\pi)^2u^2$.

4.

Consider the following equation:

$$\begin{aligned} f(x,y) - \bigtriangledown^2 f(x,y) &= f(x,y) - [f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)] \\ &= 6f(x,y) - [f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) + f(x,y)] \\ &= 5\{1.2f(x,y) - \frac{1}{5}[f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) + f(x,y)]\} \\ &= 5[1.2f(x,y) - \bar{f}(x,y)] \end{aligned}$$

where $\bar{f}(x,y)$ denotes the average of $f(x,y)$ in a predefined neighborhood that is centered at (x,y) and includes the center pixel and its four immediate neighbors. Treating the constants in the last line of the above equation as proportionality factors, we may write

$$f(x,y) - \bigtriangledown^2 f(x,y) \sim f(x,y) - \bar{f}(x,y).$$

The right side of this equation is recognized as the definition of unsharp masking given in Eq. (3.7-7). Thus, it has been demonstrated that subtracting the Laplacian from an image is proportional to unsharp masking.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 3/07

Taajuustason menetelmät kuvien parantamisessa I

1. Totea, että ortogonaalisuusehto

$$\sum_{x=0}^{N-1} e^{j2\pi rx/N} e^{-j2\pi ux/N} = \begin{cases} N, & \text{kun } r-u=kN \\ 0, & \text{muuten} \end{cases}$$

   pätee, kun r, u, N ja k ovat kokonaislukuja.

2. Laske sekvenssin $f(0) = 2, f(1) = 3, f(2) = 4, f(3) = 4$ Fourier–muunnos. Käänteismuunna saatu tulos ja vertaa alkuperäisiin arvoihin.

3. Osoita, että kahden funktion konvoluution Fourier-muunnos saadaan kertomalla funktioiden Fourier-muunnokset keskenään. Oleta yksinkertaisuuden vuoksi funktioiden olevan yhden muuttujan funktioita.

4. Gaussinen alipäästösuodin taajuustasossa on muotoa

$$H(u,v) = Ae^{-(u^2+v^2)/2\sigma^2}.$$

   Osoita, että vastaava spatiaalitason suodin on muotoa

$$h(x,y) = A2\pi\sigma^2 e^{-2\pi^2\sigma^2(x^2+y^2)}.$$

5. Tarkastellaan kuvantasoitusmenetelmää, jossa lasketaan keskiarvo kunkin pisteen neljästä lähimmästä naapurista, mutta itse piste jätetään pois keskiarvon laskennasta.

   (a) Muodosta vastaava taajuustason suodin $H(u,v)$.

   (b) Osoita, että saatu $H(u,v)$ on alipäästösuodin.

   (c) Tutki myös vaihevastetta $\phi(u,v) = \arg H(u,v)$.

T-61.5100 Digital image processing, Exercise 3/07

Image enhancement in the frequency domain I

1. Show that the orthogonality condition

$$\sum_{x=0}^{N-1} e^{j2\pi rx/N} e^{-j2\pi ux/N} = \begin{cases} N, & \text{when } r-u=kN \\ 0, & \text{otherwise} \end{cases}$$

   is correct, when r, u, N and k are integers.

2. Fourier transform the sequence $f(0)=2, f(1)=3, f(2)=4, f(3)=4$. Then calculate the inverse Fourier transform and compare the result with the original sequence.

3. Show that the Fourier transform of the convolution of two functions is the product of their Fourier transforms. For simplicity, assume 1-D functions.

4. A Gaussian lowpass filter in the frequency domain has the transfer function

$$H(u,v) = Ae^{-(u^2+v^2)/2\sigma^2}.$$

   Show that the corresponding filter in the spatial domain has the form

$$h(x,y) = A2\pi\sigma^2 e^{-2\pi^2\sigma^2(x^2+y^2)}.$$

5. Suppose that you form a lowpass spatial filter that averages the 4-neighbors of a point (x,y), but excludes the point (x,y) itself.

   (a) Find the equivalent filter $H(u,v)$ in the frequency domain.

   (b) Show that $H(u,v)$ is a lowpass filter.

   (c) Consider also the phase response $\phi(u,v) = \arg H(u,v)$.

T-61.5100 Digital image processing, Exercise 3/07

1.

These will be needed:

$$\sum_{x=0}^{N-1} q^x = \frac{1-q^N}{1-q}, \quad e^{jx} = \cos x + j \sin x$$

First $r = u + kN$:

$$\sum_{x=0}^{N-1} e^{j2\pi rx/N} e^{-j2\pi ux/N} = \sum_{x=0}^{N-1} e^{j2\pi(r-u)x/N} = \sum_{x=0}^{N-1} e^{j2\pi kx} = \sum_{x=0}^{N-1} \underbrace{\cos 2\pi kx}_{1} + j \underbrace{\sin 2\pi kx}_{0} = N$$

Next $r \neq u + kN$:

$$\sum_{x=0}^{N-1} e^{j2\pi(r-u)x/N} = \sum_{x=0}^{N-1} \left(e^{j2\pi(r-u)/N}\right)^x$$

This is a geometric sum. Thus

$$= \frac{1-\left(e^{j2\pi(r-u)/N}\right)^N}{1-\left(e^{j2\pi(r-u)/N}\right)} = \frac{1-e^{j2\pi(r-u)}}{1-e^{j2\pi(r-u)/N}} = \frac{1-\overbrace{\cos 2\pi(r-u)}^{1} - j\overbrace{\sin 2\pi(r-u)}^{0}}{\underbrace{1-e^{j2\pi(r-u)/N}}_{\neq 0 \text{ always, provided that } r-u\neq kN,\ k\in Z}} = 0$$

2.

The discrete Fourier-transform and the inverse transform:

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{-j2\pi ux/N}, \quad f(x) = \sum_{u=0}^{N-1} F(u)e^{j2\pi ux/N}$$

The sequence is now $f(0) = 2, f(1) = 3, f(2) = 4, f(3) = 4$. The Fourier-transform is done using 4 points $\Longrightarrow$ $N = 4$:

$$F(0) = \frac{1}{4}\sum_{x=0}^{3} f(x)e^0 = \frac{1}{4}[f(0)+f(1)+f(2)+f(3)] = \frac{1}{4}(2+3+4+4) = 3.25$$

$$\begin{aligned}F(1) &= \frac{1}{4}\sum_{x=0}^{3} f(x)e^{-j2\pi x/4} = \frac{1}{4}(2e^0 + 3e^{-j\pi/2} + 4e^{-j\pi} + 4e^{-j3\pi/2}) \\ &= \frac{1}{4}[2 + 3(\cos(\pi/2) - j\sin(\pi/2)) + 4(\cos(\pi) - j\sin(\pi)) + 4(\cos(3\pi/2) - j\sin(3\pi/2))] \\ &= \frac{1}{4}[2 + 3(-j) + 4(-1) + 4(j)] = \frac{1}{4}(-2+j)\end{aligned}$$

$$F(2) = \frac{1}{4}\sum_{x=0}^{3} f(x)e^{-j2\pi 2x/4} = -\frac{1}{4}(1+j0) = -\frac{1}{4}$$

$$F(3) = \frac{1}{4}\sum_{x=0}^{3} f(x)e^{-j2\pi 3x/4} = -\frac{1}{4}(2+j)$$

Then we do the inverse transform:

$$\begin{aligned}
f(0) &= \sum_{u=0}^{3} F(u)e^{0} = \frac{13}{4} + \frac{1}{4}(-2+j) - \frac{1}{4} - \frac{1}{4}(2+j) = 2 \\
f(1) &= \sum_{u=0}^{3} F(u)e^{j2\pi u/4} = \frac{13}{4} + \frac{1}{4}(-2+j)e^{j\pi/2} - \frac{1}{4}e^{j\pi} - \frac{1}{4}(2+j)e^{j3\pi/2} = 3 \\
f(2) &= \sum_{u=0}^{3} F(u)e^{j2\pi 2u/4} = \frac{13}{4} + \frac{1}{4}(-2+j)(-1) - \frac{1}{4} - \frac{1}{4}(2+j)(-1) = 4 \\
f(3) &= \sum_{u=0}^{3} F(u)e^{j2\pi 3u/4} = \frac{13}{4} + \frac{1}{4}(-2+j)(-j) - \frac{1}{4}(-1) - \frac{1}{4}(2+j)j = 4
\end{aligned}$$

The result is thus the original sequence.

3.

Convolution:

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\alpha)g(x-\alpha)\,d\alpha$$

The Fourier transform of a delay:

$$\mathcal{F}\{g(x-\alpha)\} = G(u)\cdot e^{-j2\pi u\alpha}$$

By direct substitution into the definition, we get

$$\mathcal{F}\{f(x)*g(x)\} = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(\alpha)g(x-\alpha)\,d\alpha\right]e^{-j2\pi ux}\,dx$$

$$= \int_{-\infty}^{\infty} f(\alpha)\left[\int_{-\infty}^{\infty} g(x-\alpha)e^{-j2\pi ux}\,dx\right]d\alpha = G(u)\int_{-\infty}^{\infty} f(\alpha)e^{-j2\pi u\alpha}\,d\alpha = G(u)F(u)$$

This is a very practical result, since it is often easier to calculate separately the transforms of f and g by FFT, multiply them and calculate the inverse transform than to calculate the convolution straight from the definition.

4.

We start with only one variable and show first that, if

$$H(u) = e^{-u^2/2\sigma^2}$$

then

$$h(x) = \int_{-\infty}^{\infty} e^{-u^2/2\sigma^2}e^{j2\pi ux}du = \sqrt{2\pi}\sigma e^{-2\pi^2x^2\sigma^2}.$$

We can express the integral in the preceding equation as

$$h(x) = \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[u^2-j4\pi\sigma^2ux]}du.$$

We now make use of the identity

$$e^{-\frac{(2\pi)^2x^2\sigma^2}{2}}e^{\frac{(2\pi)^2x^2\sigma^2}{2}} = 1.$$

Inserting this identity in the preceding integral yields

$$h(x) = e^{-\frac{(2\pi)^2x^2\sigma^2}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[u^2 - j4\pi\sigma^2 ux - (2\pi)^2\sigma^4 x^2]} du = e^{-\frac{(2\pi)^2x^2\sigma^2}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[u - j2\pi\sigma^2 x]^2} du.$$

Next we make the change of variable $r = u - j2\pi\sigma^2 x$. Then, $dr = du$ and the above integral becomes

$$h(x) = e^{-\frac{(2\pi)^2x^2\sigma^2}{2}} \int_{-\infty}^{\infty} e^{-\frac{r^2}{2\sigma^2}} dr.$$

Finally, we multiply and divide the right side of this equation by $\sqrt{2\pi}\sigma$:

$$h(x) = \sqrt{2\pi}\sigma e^{-\frac{(2\pi)^2x^2\sigma^2}{2}} [\frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} e^{-\frac{r^2}{2\sigma^2}} dr].$$

The expression inside the brackets is recognized as a Gaussian probability density function, whose integral from $-\infty$ to ∞ is 1. Therefore,

$$h(x) = \sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2 x^2}.$$

With this result as background, we now move into two-dimensional case. By substituting directly into definition of the inverse Fourier transform we have:

$$h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} Ae^{-(u^2+v^2)/2\sigma^2} e^{j2\pi(ux+vy)} dudv = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} Ae^{(-\frac{u^2}{2\sigma^2} + j2\pi ux)} du \right] e^{(-\frac{v^2}{2\sigma^2} + j2\pi vy)} dv$$

By using the obtained result for the one-dimensional case, we now have the final result:

$$h(x,y) = (A\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2 x^2})(\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2 y^2}) = A2\pi\sigma^2 e^{-2\pi^2\sigma^2(x^2+y^2)}.$$

5.

The spatial average is

$$g(x,y) = \frac{1}{4}\left[f(x,y+1) + f(x+1,y) + f(x,y-1) + f(x-1,y)\right].$$

We need the following property of the Fourier transform:

$$\mathcal{F}\left\{f(x-x_0, y-y_0)\right\} = \exp\left[-j2\pi(ux_0 + vy_0)/N\right] F(u,v)$$

a)

$$G(u,v) = \frac{1}{4}\left[e^{-j2\pi v/N} + e^{-j2\pi u/N} + e^{j2\pi v/N} + e^{j2\pi u/N}\right] F(u,v) = H(u,v)F(u,v).$$

Because

$$e^{-jx} = \cos x - j\sin x \quad \text{and} \quad e^{jx} = \cos x + j\sin x$$

we get

$$H(u,v) = \frac{1}{2}\left[\cos\frac{2\pi u}{N} + \cos\frac{2\pi v}{N}\right]$$

that is the filter transfer function in the frequency domain. ($H(u,v)$ is real because the filter is symmetric!)

b) $|H(0,0)| = 1$ and $|H|$ decreases as a function of distance from the origin. This is the characteristic of a lowpass filter. In fact this happens monotonically in the directions of u and v axis only; in the diagonal directions also the higher frequencies are emphasized:

$$\left| H(\pm\frac{N}{2}, \pm\frac{N}{2}) \right| = |-1| = 1$$

Look at the images!

c) The phase response can be calculated from $H(u,v)$. Because $\text{Im}\{H(u,v)\} = 0$, then the phase response is 0 when $\text{Re}\{H(u,v)\} > 0$, and π where $\text{Re}\{H(u,v)\} < 0$:

$$\arg H(u,v) = \begin{cases} 0, & \text{if } (|u| + |v|)/N \leq 0.5 \\ \pi, & \text{if } (|u| + |v|)/N > 0.5 \end{cases}$$

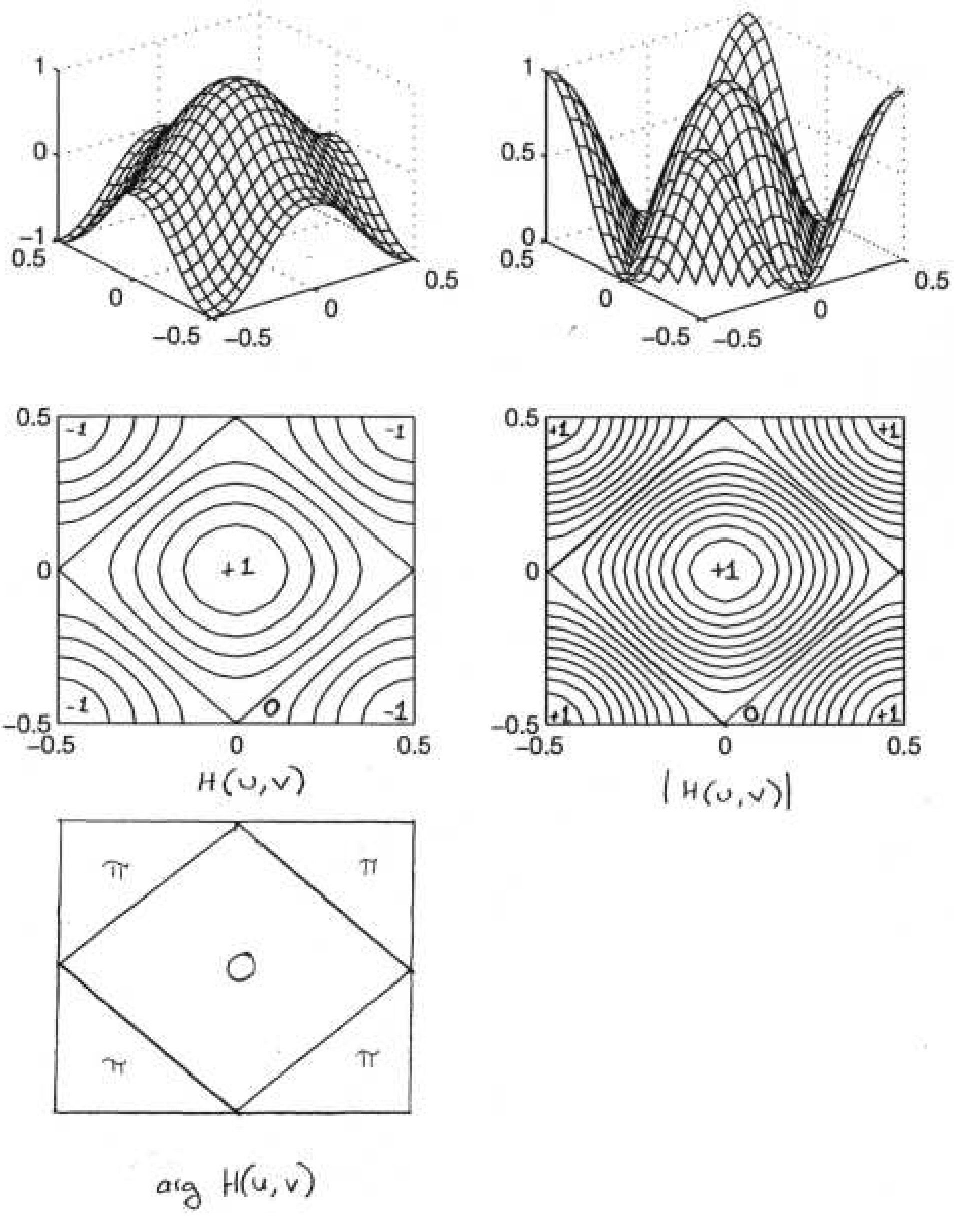
Look at the images!

So H retains (but turns) “the chessboard” ($u = v = N/2$) but averages the horizontal and vertical lines in images.

```
                              < M A T L A B (R) >
                (c) Copyright 1984-97 The MathWorks, Inc.
                              All Rights Reserved
                               Version 5.1.0.421
                                  May 25 1997

  To get started, type one of these commands: helpwin, helpdesk, or demo.
  For information on all of the MathWorks products, type tour.

>> u=-0.5:.05:0.5;
>> v=u;
>> [X,Y]=meshdom(u,v);
>> Z=0.5*(cos(2*pi.*X)+cos(2*pi.*Y));
>> subplot(3,2,1), mesh(u,v,Z);
>> subplot(3,2,2), mesh(u,v,abs(Z));
>> subplot(3,2,3), contour(u,v,Z);
>> subplot(3,2,4), contour(u,v,abs(Z));
>> subplot(3,2,5), mesh(u,v,atan2(imag(Z),real(Z)));
>> print -dps
```



T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 4/07

Taajuustason menetelmät kuvien parantamisessa II

1. Milloin Butterworth–alipäästösuodin

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$

(missä $D(u,v) = \sqrt{u^2 + v^2}$) on ideaalinen alipäästösuodin?

2. Tutkitaan, millaiselta Butterworth–alipäästösuotimet näyttävät spatiaalitasossa. Olkoon rajataajuus $D_0 = N/6$. Muodosta asteluvuilla $n = 1$ ja $n = 2$ spatiaalisuotimet, jotka ovat kokoa 3×3 ja 5×5.

3. Kuvanparannuksessa käytetään yleisesti menetelmää, joka yhdistää korkeitten taajuuksien korostuksen ja histogrammin tasoituksen. Tällä menetelmällä saadaan terävöitettyä reunoja ja parannettua kontrastia.

   (a) Tutki, onko toimenpiteitten suoritusjärjestyksellä väliä.

   (b) Mikäli järjestyksellä on merkitystä, anna peukalosääntö siitä, kumpi kannattaa suorittaa ensin.

4. Ajatellaan, että meillä on joukko tähtitaivasta esittäviä kuvia. Jokaisessa kuvassa on joukko kirkkaita, laajalle levinneitä pisteitä, jotka vastaavat tähtiä tietyssä suunnassa maailmankaikkeutta. Ongelmana on se, että tähdet ovat tuskin nähtävissä ilmakehän dispersion aiheuttaman ylimääräisen valaisun vuoksi. Tämän tyyppisiä kuvia voidaan kuitenkin yrittää mallittaa vakiona pysyvän valaistuskomponentin ja impulssijoukon tulona. Muodosta tähän mallitukseen perustuva homomorfinen suodatusmenetelmä, jolla tähdet saadaan näkyviin.

5. Käsitellään elektronimikroskoopilla otettuja kuvia. Kuvien tulkinnan parantamiseksi päätetään käyttää digitaalista kuvanparannusta. Tutkittaessa edustavaa joukkoa näytekuvia havaitaan seuraavanlaisia ongelmia:

   (a) kirkkaita erillään olevia pisteitä, jotka eivät ole mielenkiintoisia kuvan tulkinnan kannalta

   (b) riittämätön terävyys

   (c) joidenkin kuvien kontrasti ei ole riittävä

   (d) siirtymiä keskimääräisessä harmaatasoarvossa silloin, kun arvon pitäisi olla K joidenkin intensiteettimittausten oikeanlaiseksi esittämiseksi.

   Kuvanparannusta halutaan käyttää näiden virheiden korjaamiseen. Lisäksi väritetään vakioarvoisella punaisella kaikki harmaatasot I_1:stä I_2:een (muiden harmaatasojen sävy pysyy samana). Esitä sarja prosessointiaskelia halutun lopputuloksen aikaansaamiseksi.

T-61.5100 Digital image processing, Exercise 4/07

Image enhancement in the frequency domain II

1. Under what conditions does the Butterworth lowpass filter

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$

   (where $D(u,v) = \sqrt{u^2 + v^2}$) become an ideal lowpass filter?

2. Investigate what the Butterworth low-pass filters look like in the spatial domain. Let the cutoff frequency be $D_0 = N/6$. Generate the spatial masks of size 3×3 and 5×5 of order $n = 1$ ja $n = 2$.

3. A popular procedure for image enhancement combines high-frequency emphasis and histogram equalization to achieve edge sharpening and contrast enhancement.

   (a) Prove whether or not it matters which process is applied first.

   (b) If the order does matter, give a rationale for using one or the other method first.

4. Suppose that you are given a set of images generated by an experiment dealing with the analysis of stellar events. Each image contains a set of bright, widely scattered dots corresponding to stars in a sparsely occupied section of the universe. The problem is that the stars are barely visible, owing to superimposed illumination resulting from athmospheric dispersion. If these images are modeled as the product of a constant illumination component with a set of impulses, give an enhancement procedure based on homomorphic filtering designed to bring out the image components due to the stars themselves.

5. Images generated by an electronic microscope are being inspected. In order to simplify the inspection task, digital image enhancement is used. When a representative set of images is examined, following problems are found:

   (a) bright, isolated dots that are of no interest

   (b) lack of sharpness

   (c) not enough contrast in some images

   (d) shifts in the average grey-level value, when this value should be K to perform correctly certain intensity measurements.

   Image enhancement will be used to correct all these problems. In addition, all grey levels in the band between I_1 and I_2 will be colored in constant red, while keeping normal tonality in the remaining grey levels. Propose a sequence of processing steps to achieve the desired goal.

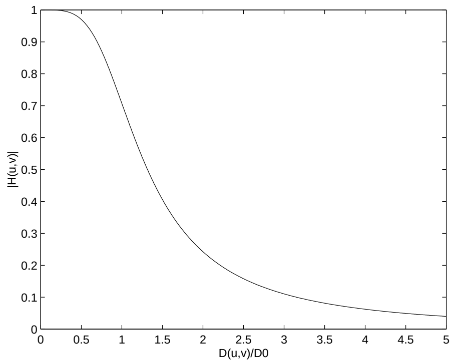
T-61.5100 Digital image processing, Exercise 4/07

1.

The frequency response of the Butterworth filter is flat in both the passband and the stopband. The transition band is usually quite wide, and the filter order must be higher than in other common filters if we wish to fulfill the required specifications.
The Butterworth lowpass filter is given as

$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}},$$

where $D(u,v) = \sqrt{u^2 + v^2}$. Its frequency response looks like that:



<u>The answer:</u> Never. The Butterworth filter approaches the ideal filter when $n \to \infty$, because

$$\lim_{n\to\infty} |H(u,v)| = \begin{cases} 1, \text{ when } D(u,v) < D_0 \\ 0, \text{ when } D(u,v) > D_0 \end{cases}$$

In fact $H(u,v) = 1/2$ always if $D(u,v) = D_0$, so the filter does not *exactly* approach the ideal filter.

2.

Let $h(x,y)$ be an $m \times m$-sized spatial mask. Let $\mathbf{h}$ be the mask ordered as an m^2-sized vector, e.g.,

$$\mathbf{h}_{(x-1)m+y} = h(x,y).$$

The Fourier transform is linear. Let the complex transformation matrix be $\mathbf{F}$, and the transformed mask (N^2-sized complex vector) be $\hat{\mathbf{H}}$:

$$\hat{\mathbf{H}} = \mathbf{F}\mathbf{h}.$$

Now we want $\hat{\mathbf{H}}$ to be as close as possible to the Butterworth filter $\mathbf{H}$ (also N^2-sized complex vector). In effect we want to minimize the error function,

$$\mathcal{E}(\mathbf{h}) = \|\hat{\mathbf{H}} - \mathbf{H}\|^2 = \|\mathbf{F}\mathbf{h} - \mathbf{H}\|^2.$$

The minimum can be found where the partial derivative vanishes:

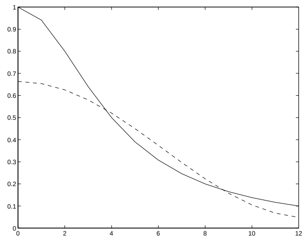
$$\frac{\partial \mathcal{E}(\mathbf{h})}{\partial \mathbf{h}} = 2\mathbf{F}^T(\mathbf{F}\mathbf{h} - \mathbf{H}) = 0.$$

This is solved when $\mathbf{F}\mathbf{h} - \mathbf{H} = 0$, i.e.

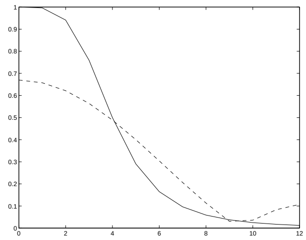
$$\mathbf{h} = \mathbf{F}^\dagger \mathbf{H},$$

where $\mathbf{F}^\dagger = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T$ is the pseudo-inverse of $\mathbf{F}$.
The Matlab-code that generates the spatial mask for the Butterworth filter can be found in `http://www.cis.hut.fi/Opinnot/T-61.5100/spatial.m`

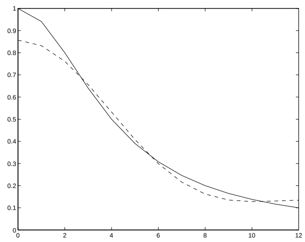
Results:



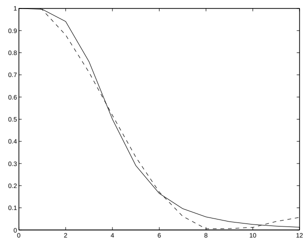
0.0404	0.0736	0.0404
0.0736	0.2075	0.0736
0.0404	0.0736	0.0404



0.0576	0.0800	0.0576
0.0800	0.1196	0.0800
0.0576	0.0800	0.0576



0.0067	0.0130	0.0156	0.0130	0.0067
0.0130	0.0404	0.0736	0.0404	0.0130
0.0156	0.0736	0.2075	0.0736	0.0156
0.0130	0.0404	0.0736	0.0404	0.0130
0.0067	0.0130	0.0156	0.0130	0.0067



0.0107	0.0244	0.0318	0.0244	0.0107
0.0244	0.0576	0.0800	0.0576	0.0244
0.0318	0.0800	0.1196	0.0800	0.0318
0.0244	0.0576	0.0800	0.0576	0.0244
0.0107	0.0244	0.0318	0.0244	0.0107

Solid line: the frequency response of the Butterworth filter.
Dotted line: the frequency response of the spatial mask.
Left column: $n = 1$. Right column: $n = 2$.
Notice that in both cases the 3×3 mask is included in the 5×5 mask. This is a general property that comes from the orthogonality. If the mask size is increased, the obtained frequency response approaches the frequency response of the Butterworth filter.

3.

a) The filtered image is given by

$$g(x, y) = h(x, y) * f(x, y),$$

where $h(x, y)$ is the spatial filter (inverse Fourier transform of the frequency-domain filter) and f is the input image. Histogram processing this result yields

$$g'(x, y) = T[g(x, y)] = T[h(x, y) * f(x, y)],$$

where T denotes the histogram equalization transformation. If we histogram-equalize first, then

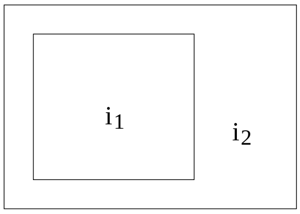
$$g(x, y) = T[f(x, y)]$$

and

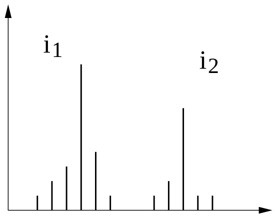
$$g'(x, y) = h(x, y) * T[f(x, y)].$$

In general, T is a nonlinear function determined by the nature of the pixels in the image from which it is computed. Thus, in general $T[h(x, y) * f(x, y)] \neq h(x, y) * T[f(x, y)]$ and the order does matter.

b) As indicated in Section 4.4, highpass filtering severely diminishes the contrast of an image. Although high-frequency emphasis helps some, the improvement is usually not dramatic. Thus, if an image is histogram equalized first, the gain in contrast improvement will essentially be lost in the filtering process. Therefore, the procedure in general is *to filter first and histogram-equalize the image after that.*
A small example: Let us assume that the following image has two areas with even intensities, i_1 and i_2.

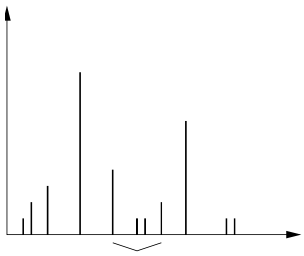


Image



Histogram of the imag

These areas are shown in the histogram as high peaks and in the frequency spectrum as low frequencies. Changes in intensities (high frequencies) in these areas and in their edges are typically shown as lower peaks around the high peaks. If we now emphasize the high frequencies, we in fact emphasize those lower peaks, and thus we can make more use of these high frequencies in histogram equalization. If we do the histogram equalization first, it is more based on the high peaks in the histogram. Then we might lose some contrast in the final image:



4.

Let us assume that there is only a single star that is modeled as an impulse $\delta(x - x_0, y - y_0)$ where (x_0, y_0) are the coordinates of the star. (A discrete impulse is one at the origin, zero elsewhere.) K is the illumination and ϵ the reflectance of the sky. Then the image model is

$$f(x, y) = i(x, y)r(x, y) = K(\delta(x - x_0, y - y_0) + \epsilon)$$

Homomorphic filtering:

$$f(x,y) \rightarrow \boxed{\ln} \rightarrow \boxed{\mathcal{F}} \rightarrow \boxed{H(u,v)} \rightarrow \boxed{\mathcal{F}^{-1}} \rightarrow \boxed{\exp} \rightarrow g(x,y)$$

So, first we take the logarithm from the image:

$$z(x, y) = \ln f(x, y) = \ln K + \ln(\delta(x - x_0, y - y_0) + \epsilon) = \ln K + \ln \epsilon + \delta(x - x_0, y - y_0)(\ln(1 + \epsilon) - \ln \epsilon).$$

Then we Fourier transform:

$$\begin{aligned}\mathcal{F}\{z(x, y)\} &= \mathcal{F}\{\ln K\epsilon\} + \mathcal{F}\{\delta(x - x_0, y - y_0)(\ln(1 + \epsilon) - \ln \epsilon)\} \\ &= C_1\delta(u, v) + C_2 e^{-j2\pi(\frac{ux_0}{M} + \frac{vy_0}{N})},\end{aligned}$$

where $C_1 = \ln K\epsilon$ ja $C_2 = \ln(1 + \epsilon) - \ln \epsilon$.

From this result, it is evident that the contribution of illumination is an impulse at the origin of the frequency plane. It can be canceled by high-pass filtering the image (e.g. using a notch filter). Extension of this development to multiple impulses (stars) is straightforward. The filter will be the same.

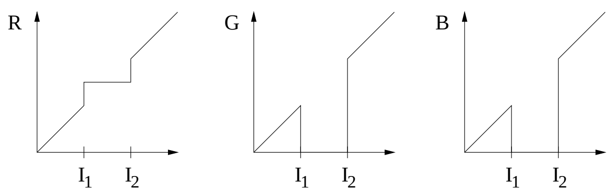
5.

a) Perform a median filtering operation – an isolated bright dot won't greatly affect the median value.

b) Sharpness can be enhanced by high-frequency emphasis. Then areas with high frequencies (edges) will be emphasized, but areas with low frequencies (even areas) are canceled.

c) Contrast can be enhanced by histogram equalization.

d) Compute the average gray level, K_0. Add the quantity $(K - K_0)$ to all pixels.
Coloring is achieved with the following color transformation:



So only the red color remains in $I_1 \ldots I_2$. Here R, G, and B are the color components of an RGB color monitor.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 5/07

Kuvien entistäminen

1. Käsitellään lineaarista paikkainvarianttia kuvanvääristymissysteemiä (image degradation system), jonka impulssivaste on
$$h(x,y) = e^{-\left(x^2+y^2\right)}$$
Olkoon systeemin sisääntulona kuva, joka koostuu äärettömän ohuesta viivasta kohdassa $x = a$ ja jonka malli on $f(x,y) = \delta(x-a)$. Mikä on systeemin ulostulokuva $g(x,y)$, kun oletetaan, ettei kohinaa esiinny?

2. Kuvaamisen aikana kohde liikkuu ensin tasaisesti pystysuunnassa ajan T_1 ja sitten vaakasuunnassa ajan T_2. Oletetaan, että liikesuunnan vaihtamiseen kuluva aika on merkityksetön samoin kuin kameran sulkimen avautumiseen ja sulkeutumiseen kuluvat ajatkin. Määrää syntyvän häiriön siirtofunktio $H(u,v)$.

3. Cannon esitteli vuonna 1974 yhdenmukaisen tehospektrin entistämissuotimen (power spectrum equalization filter) $R(u,v)$, jonka ideana oli pakottaa entistetyn kuvan tehospektri samaksi kuin alkuperäisen kuvan tehospektri:
$$S_{\hat{f}}(u,v) = |R(u,v)|^2 S_g(u,v) = S_f(u,v).$$
Määrää entistämissuotimen taajuusvasteen itseisarvo $|R(u,v)|$.

4. Laske vahvistukset (taajuusvasteen itseisarvot)
   (a) käänteissuotimelle
   (b) yhdenmukaisen tehospektrin suotimelle
   (c) Wiener-suotimelle

   taajuusalueen pisteissä (u,v), joissa signaalin tehospektrillä $S_f(u,v)$, kohinan tehospektrillä $S_n(u,v)$ sekä pisteen leviämisfunktion taajuusvasteen $H(u,v)$ itseisarvolla on seuraavat arvot:

$\lvert H(u,v)\rvert$	$S_f(u,v)$	$S_n(u,v)$	
0	0	N	• pisteenleviämisfunktion taajuusvasteen nolla
0	S	0	
0	S	N	
H	0	N	• signaalin teho on nollassa
H	S	0	• kohinaa ei ole
1.0	3000.0	0.01	• uv-origon tuntumassa
0.7	0.7	0.01	• matalilla taajuuksilla
0.01	0.005	0.01	• korkeilla taajuuksilla

5. (a) Osoita, että 3×3-kokoisen, keskiarvon laskevan maskin käyttäminen voidaan korvata 1×3- ja 3×1-kokoisten maskien käyttämisellä peräkkäin. Vertaa tarvittavien yhteenlaskuoperaatioiden lukumäärää kummassakin tapauksessa.
   (b) Vastaavasti, kuinka suhtautuvat toisiinsa tarvittavien yhteen- ja kertolaskujen lukumäärät yleisessä tapauksessa, jossa alkuperäinen N×N-kokoinen maski on hajoitettu 1×N- ja N×1-kokoisiksi maskeiksi ja kertoimet maskeissa poikkeavat ykkösestä?
   (c) Esitä 3×3-kokoiset Sobelin gradienttimaskit. Osoita toiselle Sobel-maskille, että sen toiminta voidaan keskiarvomaskin tapaan jakaa kahdeksi peräkkäiseksi yksiulotteiseksi maskioperaatioksi.
   (d) Tutki, voidaanko myös 3×3-kokoinen diskreetti Laplace-operaattori toteuttaa kahtena peräkkäisenä yksiulotteisena operaationa.

T-61.5100 Digital image processing, Exercise 5/07

Image restoration

1. Consider a linear, position invariant image degradation system with impulse response

$$h(x, y) = e^{-\left(x^2+y^2\right)}$$

   Suppose the input to the system is an image consisting of a line of infinitesimal width located at $x = a$, and modeled by $f(x, y) = \delta(x - a)$. Assuming no noise, what is the output image $g(x, y)$?

2. During acquisition, an image undergoes uniform linear motion in the vertical direction for a time T_1. The direction of motion then switches to the horizontal direction for a time interval T_2. Assuming that the time it takes the image to change directions is negligible, and the shutter opening and closing times are negligible also, give and expression for the blurring function, $H(u, v)$.

3. Cannon [1974] suggested the power spectrum equalization filter $R(u, v)$ based on the premise of forcing the power spectrum of the restored image to equal the power spectrum of the original image:

$$S_{\hat{f}}(u, v) = |R(u, v)|^2 S_g(u, v) = S_f(u, v).$$

   Find $|R(u, v)|$, the magnitude response of the restoration filter.

4. Find magnitude responses for the

   (a) inverse filter

   (b) power spectrum equalization filter

   (c) Wiener filter

   in the points (u, v) of frequency domain, where signal power spectrum $S_f(u, v)$, noise $S_n(u, v)$ power spectrum and magnitude response of point spread function (PSF) $H(u, v)$ have the following values:

$\lvert H(u,v)\rvert$	$S_f(u,v)$	$S_n(u,v)$	
0	0	N	• zero of PSF
0	S	0	
0	S	N	
H	0	N	• signal power zero
H	S	0	• no noise
1.0	3000.0	0.01	• close to uv-origin
0.7	0.7	0.01	• low frequencies
0.01	0.005	0.01	• high frequencies

5. (a) Show that the application of a 3×3-sized local mean mask can be replaced by 1×3 and 3×1 masks applied sequentially. Compare the amount of additions that are needed in both cases.

   (b) Compare the amounts of additions and multiplications that are needed in a general case, where a N×N mask is replaced by 1×N and N×1 masks and masks' coefficients are not equal to ones.

   (c) Depict the 3×3 Sobel gradient masks. Show for one of the Sobel masks that it can be separated as above into two one-dimensional masks.

   (d) Is it possible to separate the 3×3 discrete Laplace-operator?

T-61.5100 Digital image processing, Exercise 5/07

1.

From Eq. (5.5-13),

$$g(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(\alpha,\beta)h(x-\alpha,y-\beta)d\alpha d\beta,$$

where

$$f(\alpha,\beta)=\delta(\alpha-a)$$

and

$$h(x-\alpha,y-\beta)=e^{-[(x-\alpha)^2+(y-\beta)^2]}.$$

Then we get

$$\begin{aligned}g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(\alpha-a)e^{-[(x-\alpha)^2+(y-\beta)^2]}d\alpha d\beta\\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(\alpha-a)e^{-[(x-\alpha)^2]}e^{-[(y-\beta)^2]}d\alpha d\beta\\ &= \int_{-\infty}^{\infty}\delta(\alpha-a)e^{-[(x-\alpha)^2]}d\alpha\int_{-\infty}^{\infty}e^{-[(y-\beta)^2]}d\beta\\ &= e^{-[(x-a)^2]}\int_{-\infty}^{\infty}e^{-[(y-\beta)^2]}d\beta,\end{aligned}$$

where we used the fact that the integral of the impulse is nonzero only when $\alpha=a$. Next, we note that

$$\int_{-\infty}^{\infty}e^{-[(y-\beta)^2]}d\beta=\int_{-\infty}^{\infty}e^{-[(\beta-y)^2]}d\beta$$

which is in the form of a constant times a Gaussian density with variance $\sigma^2=1/2$ or standard deviation $\sigma=1/\sqrt{2}$. In other words,

$$e^{[-(y-\beta)^2]}=\sqrt{2\pi}\sqrt{1/2}\left[\frac{1}{\sqrt{2\pi}\sqrt{1/2}}e^{[-\frac{(\beta-y)^2}{2\,(1/2)}]}\right].$$

The integral from minus to plus infinity of the quantity inside the brackets is 1, so

$$g(x,y)=e^{[-(x-a)^2]}\sqrt{2\pi}\sqrt{1/2}=\sqrt{\pi}e^{[-(x-a)^2]}.$$

This is the blurred version of the original image.

2.

Because the motion in the x- and y-directions are independent (motion is in the vertical (x) direction only at first, and then switching to motion only in the horizontal (y) direction) this problem can be solved in two steps. The first step is identical to the analysis that resulted in Eq. (5.6-10), which gives the blurring function due to vertical motion only:

$$H_1(u,v)=\frac{T_1}{\pi ua}\sin(\pi ua)e^{-j\pi ua},$$

where we are representing linear motion by the equation $x_0(t)=at/T_1$. The function $H_1(u,v)$ would give us a blurred image in the vertical direction. That blurred image is the image that would then start moving in the horizontal direction and to which horizontal blurring would be applied. This is nothing more than applying a second filter with transfer function

$$H_2(u,v)=\frac{T_2}{\pi vb}\sin(\pi vb)e^{-j\pi vb},$$

where we assumed the form $y_0(t) = bt/T_2$ for motion in the y-direction. Therefore, the overall blurring transfer function is given by the product of these two functions

$$H(u,v) = \frac{T_1 T_2}{(\pi ua)(\pi vb)} \sin(\pi ua) \sin(\pi vb) e^{-j\pi ua} e^{-j\pi vb},$$

and the overall blurred image is

$$g(x,y) = \mathcal{F}^{-1}[H(u,v)F(u,v)]$$

where $F(u,v)$ is the Fourier transform of the input image.

3.

The power spectrum of the restored image $\hat{f}$ is given by

$$S_{\hat{f}}(u,v) = |R(u,v)|^2 S_g(u,v).$$

The restoration filter should force the power spectrum of the restored image to equal the power spectrum of the original image:

$$S_{\hat{f}}(u,v) = S_f(u,v).$$

We start with the above equations:

$$\begin{aligned} S_f(u,v) &= |R(u,v)|^2 S_g(u,v) = |R(u,v)|^2 |G(u,v)|^2 \\ &= |R(u,v)|^2 \left[|H(u,v)|^2 |F(u,v)|^2 + |N(u,v)|^2 + \right. \\ &\quad \left. H(u,v)N^*(u,v)F(u,v) + H^*(u,v)F^*(u,v)N(u,v) \right] \\ &= |R(u,v)|^2 \left[|H(u,v)|^2 S_f(u,v) + S_\eta(u,v) \right], \end{aligned}$$

where the cross-terms vanish because the image and the noise are uncorrelated. $S_\eta(u,v)$ is the power spectrum of the noise. Then we solve for $|R(u,v)|$:

$$|R(u,v)| = \sqrt{\frac{1}{|H(u,v)|^2 + \frac{S_\eta(u,v)}{S_f(u,v)}}}$$

4.

<u>Inverse filter:</u>

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}$$

So the magnitude response of the inverse filter is

$$|H_{\text{INV}}(u,v)| = \frac{1}{|H(u,v)|}.$$

<u>Power spectrum equalization filter:</u>

$$|H_{\text{PSE}}(u,v)| = \left[\frac{1}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)} \right]^{1/2}$$

<u>Wiener-filter:</u>

$$\hat{F}(u,v) = \left[\frac{H^\star(u,v)}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)} \right] G(u,v),$$

where

$$|H_{\text{WIENER}}(u,v)| = \frac{|H(u,v)|}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}$$

Let us substitute the given values into these equations:

$\lvert H(u,v)\rvert$	$S_f(u,v)$	$S_\eta(u,v)$	$\lvert H_{\text{INV}}(u,v)\rvert$	$\lvert H_{\text{PSE}}(u,v)\rvert$	$\lvert H_{\text{WIENER}}(u,v)\rvert$
0	0	N	∞	0	0
0	S	0	∞	∞	0/0, usually 0
0	S	N	∞	$\sqrt{S/N}$	0
H	0	N	$1/H$	0	0
H	S	0	$1/H$	$1/H$	$1/H$
1.0	3000	0.01	1.0	≈ 1.0	≈ 1.0
0.7	0.7	0.01	1.43	≈ 1.41	≈ 1.38
0.01	0.005	0.01	100.0	≈ 0.71	≈ 0.005

5.

a) The 3×3-sized local mean mask is (scaling is omitted for simplicity)

1	1	1
1	1	1
1	1	1

The part of an image that falls under the mask is given as

a	b	c
d	e	f
g	h	i

The mask response in position e is $e^* = 1\cdot a + 1\cdot b + 1\cdot c + 1\cdot d + 1\cdot e + 1\cdot f + 1\cdot g + 1\cdot h + 1\cdot i$ $= a+b+c+d+e+f+g+h+i$. If we are using the mask [1 | 1 | 1], the responses in positions b, e, and h are $b' = a+b+c$, $e' = d+e+f$, and $h' = g+h+i$. When we then apply the mask [1 | 1 | 1]^T, the response in position e is $e'' = b'+e'+h' = (a+b+c)+(d+e+f)+(g+h+i) = e^*$.

With a 3×3 mask we have 8 additions for each mask position. With a 1×3 mask we have 2 additions for each position. Thus using 1×3 and 3×1 masks takes a total of 4 additions for each position which is half the number of additions needed with a 3×3 mask.

b) In a general case we have N^2-1 additions and N^2 multiplications with a N×N mask. Using the separate masks takes $2(N-1)$ additions and $2N$ multiplications.

c) The 3×3 Sobel gradient masks are

$G_x =$

-1	-2	-1
0	0	0
1	2	1

$G_y =$

-1	0	1
-2	0	2
-1	0	1

G_x measures horizontal edges and G_y vertical edges.

Let us use the G_x mask. The response in position e is $e^* = (g+2h+i)-(a+2b+c)$. With a one-dimensional difference mask [-1 | 0 | 1]^T the responses in positions d, e, and f are $d' = g-a$, $e' = h-b$, and $f' = i-c$. When we then apply the mask [1 | 2 | 1] the response in position e is $e'' = (g-a)+2(h-b)+(i-c) = (g+2h+i)-(a+2b+c) = e^*$. G_y mask can be used in the same way.

d) The 3×3 discrete Laplace-operator is

0	−1	0
−1	4	−1
0	−1	0

In order to separate the 3×3 discrete Laplace-operator, we must find two 3×1 vectors $\begin{bmatrix} a & b & c \end{bmatrix}^T$ and $\begin{bmatrix} d & e & f \end{bmatrix}^T$ whose outer product

ad	bd	cd
ae	be	ce
af	bf	cf

were the Laplace mask. For example, $ad = 0$. If we choose $a = 0$ then also $ae = 0$ which is not valid. If $d = 0$ then $bd = 0$ which is also not valid. So, we cannot separate the mask into two one-dimensional vectors.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 6/07

Morfologinen kuvankäsittely

1. (a) Kehitä algoritmi, joka muuttaa 8-riippuvaisen binaarisen ääriviivan m-riippuvaiseksi ääriviivaksi (katso Section 2.5.2). Voit olettaa ääriviivan yhtenäiseksi ja yhden pikselin levyiseksi.

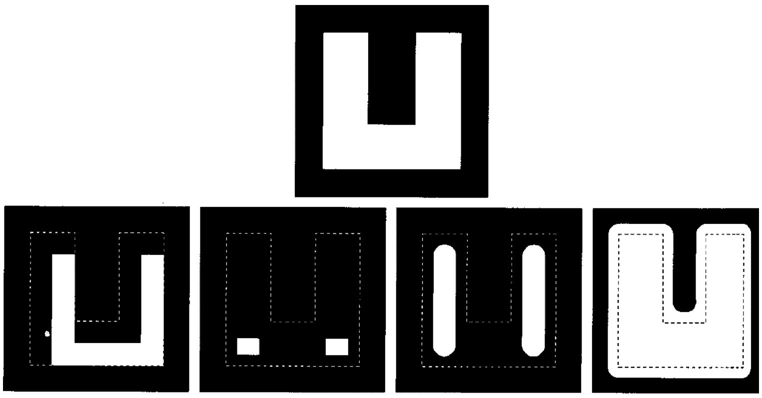
   (b) Tarvitseeko algoritmisi enemmän kuin yhden iteraation / strukturointielementti?

   (c) Onko algoritmisi toiminta riippumaton eri strukturointielementtien suoritusjärjestyksestä? Jos vastaus on kyllä, niin todista se; muuten anna esimerkki, joka kuvaa suoritusjärjestyksen vaikutusta.

2. Miten saat oheisen kuvan kohteen siistityksi tunnistusta varten käyttäen morfologisia suotimia? (Kuvassa olevan nollan reunaviiva tulisi saada yhtenäiseksi.)

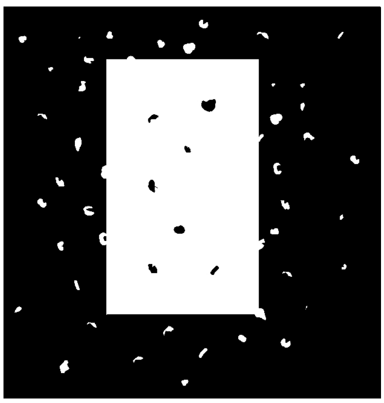
0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	0
0	0	1	0	1	0	0	1	0
0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0

3. Anna strukturointielementti ja morfologiset operaatiot, joilla on saatu aikaan kuvat (a)-(d). Merkitse selvästi strukturointielementin keskusta. Katkoviivat näyttävät vain alkuperäisen kuvan ääriviivat, eivätkä siis kuulu tuloskuvaan. Huomaa, että (d):ssä kaikki nurkat ovat pyöristetyt.



(a) (b) (c) (d)

4. Hahmottele, miltä tulosjoukot C, D, E, F näyttävät, kun seuraavat operaatiot tehdään oheiselle joukolle (kuvalle): $C = A \ominus B$; $D = C \oplus B$; $E = D \oplus B$; $F = E \ominus B$, missä B on strukturointielementti. Alkuperäisessä joukossa A kaikki siihen kuuluvat pikselit ovat valkoisia. Huomaa, että annetut operaatiot ensin avaavat A:n B:llä ja sitten sulkevat saadun tuloksen B:llä. Voit olettaa, että B on pyöreä ja juuri tarpeeksi suuri peittämään kuvan kohinakomponentit.



T-61.5100 Digital image processing, Exercise 6/07

Morphological image processing

1. (a) Give a morphological algorithm for converting an 8-connected binary boundary to an m-connected boundary (see Section 2.5.2). You may assume that the boundary is fully connected and that it is one pixel thick.

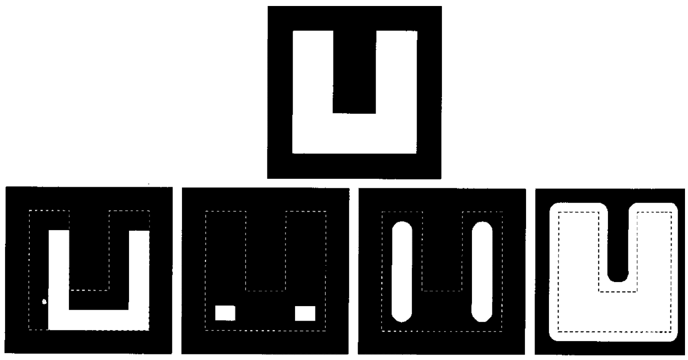
   (b) Does the operation of your algorithm require more than one iteration with each structuring element? Explain your reasoning.

   (c) Is the performance of your algorithm independent of the order in which the structuring elements are applied? If your answer is yes, prove it; otherwise give an example that illustrates the dependence of your procedure on the order of application of the structuring elements.

2. How can the given object be cleaned up by using morphological operations? (The outline of the zero in the image should be closed.)

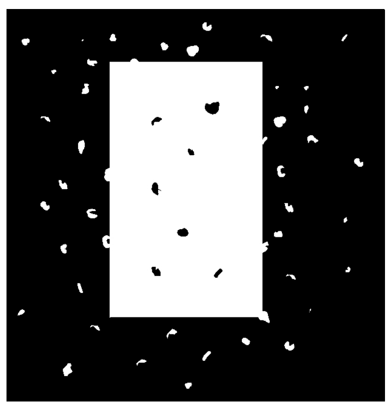
0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	0
0	0	1	0	1	0	0	1	0
0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0

3. Give the structuring element and morphological operation(s) that produced each of the results shown in images (a) through (d). Show the origin of each structuring element clearly. The dashed lines show the boundary of the original set and are included only for reference. Note that in (d) all corners are rounded.



(a) (b) (c) (d)

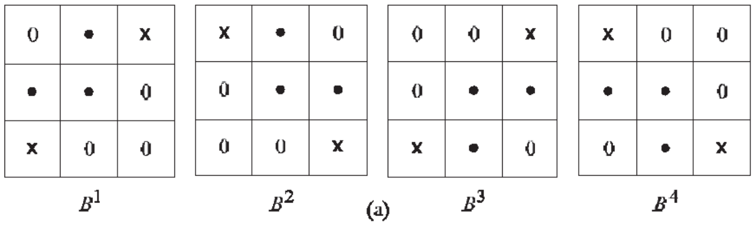
4. Sketch what the sets C, D, E, F would look like when the following sequence of operations is applied to a given image: $C = A \ominus B$; $D = C \oplus B$; $E = D \oplus B$; $F = E \ominus B$, where B is the structuring element. The initial set A consists of all the image components shown in white. Note that this sequence of operations is simply the opening of A by B, followed by the closing of that opening by B. You may assume that B is round and just large enough to enclose each of the noise components.



T-61.5100 Digital image processing, Exercise 6/07

1.

a) With reference to the discussion in Section 2.5.2, m-connectivity is used to avoid multiple paths that are inherent in 8-connectivity. In one-pixel-thick, fully connected boundaries, these multiple paths manifest themselves in the four basic patterns shown here:



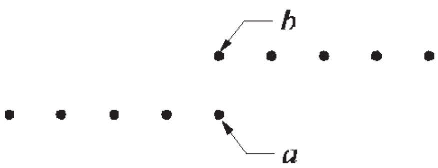
The solution to the problem is to use the hit-or-miss transform to detect the patterns and then to change the center pixel to 0, thus eliminating the multiple paths. A basic sequence of morphological steps to accomplish this is as follows:

$$
\begin{aligned}
X_1 &= A \circledast B^1 \\
Y_1 &= A \cap X_1^c \\
X_2 &= Y_1 \circledast B^2 \\
Y_2 &= Y_1 \cap X_2^c \\
X_3 &= Y_2 \circledast B^3 \\
Y_3 &= Y_2 \cap X_3^c \\
X_4 &= Y_3 \circledast B^4 \\
Y_4 &= Y_3 \cap X_4^c
\end{aligned}
$$

where A is the input image containing the boundary.

b) Only one pass is required. Application of the hit-or-miss transform using a given B^i finds all instances of occurrence of the pattern described by that structuring element.

c) The order does matter. For example, consider the sequence of points shown in next figure and assume that we are traveling from left to right. If B^1 is applied first, point a will be deleted and point b will remain after application of all other structuring elements. If, on the other hand, B^3 is applied first, point b will be deleted and point a will remain. Thus, we would end up with different (but of course, acceptable) m-paths.



2.

The morphological operation closing is used to close the outline of the zero in the image A given as

0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	0
0	0	1	0	1	0	0	1	0
0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0

Closing is dilation followed by erosion. Dilation of A by B is the set of all points z such that $\hat{B}$ (B's reflection about its origin) and A overlap by at least one element,

$$A \oplus B = \{z | [(\hat{B})_z \cap A] \subseteq A\}.$$

Erosion of A by B is the set of all points z such that B is contained in A,

$$A \ominus B = \{z | (B)_z \subseteq A\}.$$

Closing is defined as

$$A \bullet B = (A \oplus B) \ominus B.$$

The structuring element chosen for B is

×	•
•	•

B's reflection about its origin, $\hat{B}$, is then

•	•
•	×

The result of dilation of the image A by B is

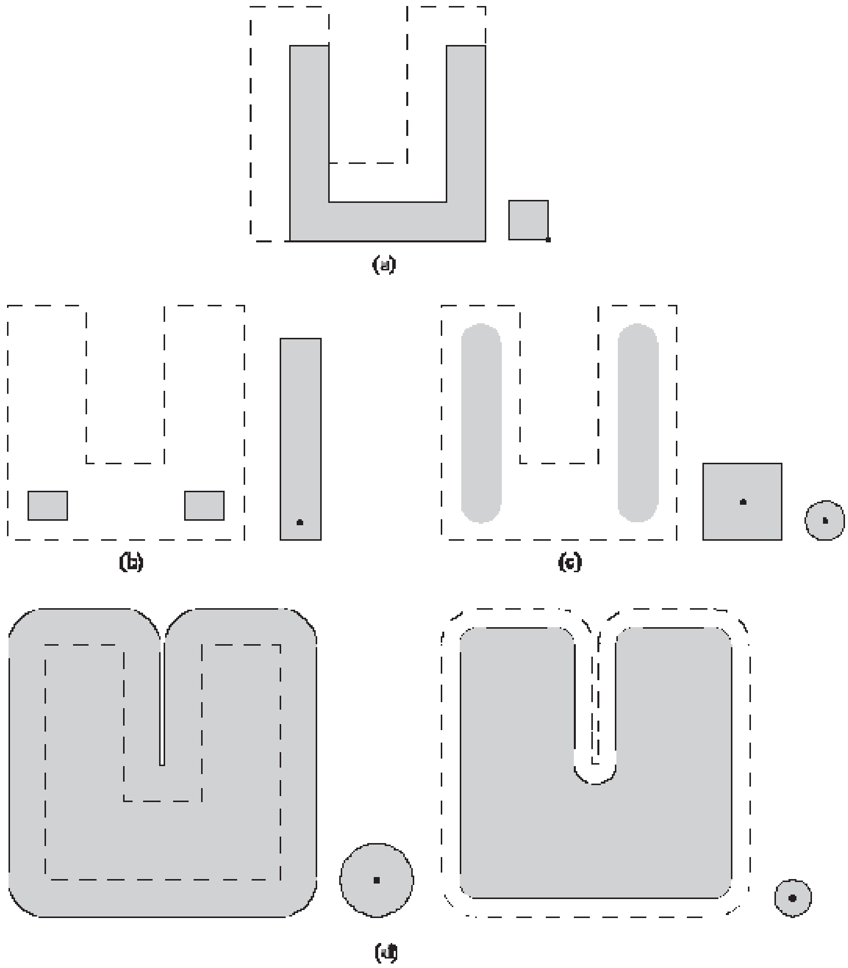
0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1
0	0	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0

The previous image after erosion is

0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	0
0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0

3.

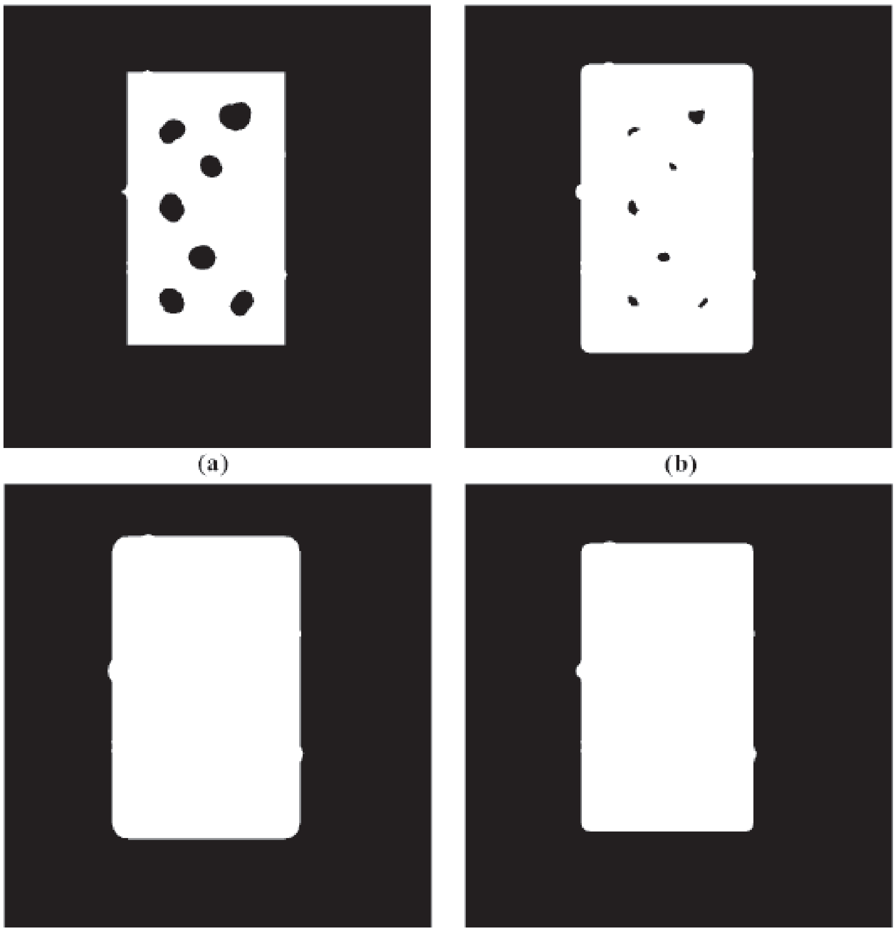
The center of each structuring element is shown as a black dot.
Solution (a) was obtained by eroding the original set (shown dashed) with the structuring element shown (note that the origin is at the bottom, right).
Solution (b) was obtained by eroding the original set with the tall rectangular structuring element shown.
Solution(c) was obtained by first eroding the image shown down to two vertical lines using the rectangular structuring element. This result was then dilated with the circular structuring element. **NOTE:** This solution is not entirely correct, after the initial eroding there is also a horizontal line connecting the two vertical ones.
Solution (d) was obtained by first dilating the original set with the large disk shown. Then dilated image was then eroded with a disk of half the diameter of the disk used for dilation.



4.

The solution is shown in the next figure. Although the images shown could be sketched by hand, they were done in MATLAB. The size of the original is 647 x 624 pixels. A disk structuring element of radius 11 was used. This structuring element was just large enough to encompass all noise elements, as given in the problem statement.
The images shown in the figure are: (a) erosion of the original, (b) dilation of the result, (c) another dilation, and finally (d) an erosion.
The first erosion (leftmost image) should take out all noise elements that do not touch the rectangle, should increase the size of the noise elements completely contained within the rectangle,

and should decrease the size of the rectangle. If worked by hand, the student may or may not realize that some imperfections are left along the boundary of the object. We do not consider this an important issue because it is scale-dependent, and nothing is said in the problem statement about this.
The first dilation (next image) should shrink the noise components that were increased in erosion, should increase the size of the rectangle, and should round the corners.
The next dilation should eliminate the internal noise components completely and further increase the size of the rectangle.
The final erosion (last image on the right) should then decrease the size of the rectangle. The rounded corners in the final answer are an important point that should be recognized by the student.



(a) (b) (c) (d)

T-61.51000 Digitaalinen kuvankäsittely, Harjoitus 7/07

Aallokkeet

1. Muodosta täysinäinen approksimaatiopyramidi ja vastaava ennustusvirhepyramidi oheiselle kuvalle:

$$f(x,y) = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}.$$

   Käytä 2×2-kokoista keskiarvoistavaa approksimaatiofiltteriä. Älä käytä interpolaatiofiltteriä (katso sivu 351 kirjasta).

2. Laske Haar-muunnos $\mathbf{T} = \mathbf{HFH}^{\mathrm{T}}$ oheiselle 2×2-kokoiselle kuvalle

$$\mathbf{F} = \begin{bmatrix} 3 & -1 \\ 6 & 2 \end{bmatrix}.$$

   Laske myös käänteinen Haar-muunnos $\mathbf{F} = \mathbf{H}^{\mathrm{T}}\mathbf{TH}$ saamallesi tuloskuvalle.

3. Laske 2-ulotteinen aallokemuunnos edellisen tehtävän 2×2-kokoiselle kuvalle $\mathbf{F}$ käyttäen Haar-aallokkeita. Piirrä tarvittava suodinpankki ja nimeä kaikki tulot ja lähdöt sopivin taulukoin.

4. Piirrä aalloke $\psi_{3,3}(x)$ Haarin aallokefunktiolle. Kirjoita lisäksi $\psi_{3,3}(x)$ Haarin skaalausfunktioiden avulla.

5. Laske 1-ulotteinen diskreetti aallokemuunnos (DWT) funktiolle $f(0) = 1$, $f(1) = 4$, $f(2) = -3$, and $f(3) = 0$ käyttäen aloitusskaalaa $j_0 = 1$. Käänteismuunna sitten saatu tulos.

T-61.5100 Digital image processing, Exercise 7/07

Wavelets

1. Construct a fully populated approximation pyramid and corresponding prediction residual pyramid for the image

$$f(x,y) = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}.$$

   Use 2×2 block neighborhood averaging for the approximation filter and omit the interpolation filter (see pp. 351 in the textbook).

2. Compute the Haar transform $\mathbf{T} = \mathbf{HFH}^{\mathrm{T}}$ of the 2×2 image

$$\mathbf{F} = \begin{bmatrix} 3 & -1 \\ 6 & 2 \end{bmatrix}.$$

   Compute also the inverse Haar transform $\mathbf{F} = \mathbf{H}^{\mathrm{T}}\mathbf{TH}$ of the obtained result.

3. Compute the two-dimensional wavelet transform with respect to Haar wavelets of the 2×2 image $\mathbf{F}$ in previous exercise. Draw the required filter bank and label all inputs and outputs with the proper arrays.

4. Draw wavelet $\psi_{3,3}(x)$ for the Haar wavelet function. Write an expression for $\psi_{3,3}(x)$ in terms of the Haar scaling function.

5. Compute the one-dimensional discrete wavelet transform (DWT) of function $f(0) = 1$, $f(1) = 4$, $f(2) = -3$, and $f(3) = 0$ with starting scale $j_0 = 1$. Then compute the inverse transform.

T-61.5100 Digital image processing, Exercise 7/07

1.

A mean approximation pyramid is formed by forming 2×2 block averages. Since the starting image is of size 4×4, $J = 2$, and $f(x, y)$ is placed in level 2 of the mean approximation pyramid. The level 1 approximation is (by taking 2×2 block averages over $f(x, y)$ and subsampling):

$$\begin{bmatrix} 3.5 & 5.5 \\ 11.5 & 13.5 \end{bmatrix}$$

and the level 0 approximation is similarly [8.5]. The completed mean approximation pyramid is

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \begin{bmatrix} 3.5 & 5.5 \\ 11.5 & 13.5 \end{bmatrix} [8.5] .$$

Since no interpolation filtering is specified, pixel replication is used in the generation of the mean prediction residual pyramid levels. Level 0 of the prediction residual pyramid is the lowest resolution approximation, [8.5]. The level 2 prediction residual is obtained by upsampling the level 1 approximation and subtracting it from the level 2 (original image). Thus, we get

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} - \begin{bmatrix} 3.5 & 3.5 & 5.5 & 5.5 \\ 3.5 & 3.5 & 5.5 & 5.5 \\ 11.5 & 11.5 & 13.5 & 13.5 \\ 11.5 & 11.5 & 13.5 & 13.5 \end{bmatrix} = \begin{bmatrix} -2.5 & -1.5 & -2.5 & -1.5 \\ 1.5 & 2.5 & 1.5 & 2.5 \\ -2.5 & -1.5 & -2.5 & -1.5 \\ 1.5 & 2.5 & 1.5 & 2.5 \end{bmatrix} .$$

Similarly, the level 1 prediction residual is obtained by upsampling the level 0 approximation and subtracting it from the level 1 approximation to yield

$$\begin{bmatrix} 3.5 & 5.5 \\ 11.5 & 13.5 \end{bmatrix} - \begin{bmatrix} 8.5 & 8.5 \\ 8.5 & 8.5 \end{bmatrix} = \begin{bmatrix} -5 & -3 \\ 3 & 5 \end{bmatrix} .$$

The mean prediction residual pyramid is therefore

$$\begin{bmatrix} -2.5 & -1.5 & -2.5 & -1.5 \\ 1.5 & 2.5 & 1.5 & 2.5 \\ -2.5 & -1.5 & -2.5 & -1.5 \\ 1.5 & 2.5 & 1.5 & 2.5 \end{bmatrix} \begin{bmatrix} -5 & -3 \\ 3 & 5 \end{bmatrix} [8.5] .$$

2.

The 2×2 Haar transformation matrix is

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} .$$

Then we get

$$\mathbf{T} = \mathbf{HFH}^{\mathrm{T}} = (\frac{1}{\sqrt{2}})^2 \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ 6 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 5 & 4 \\ -3 & 0 \end{bmatrix} .$$

Next we compute the inverse Haar transform. First, the transpose of the 2×2 Haar transformation matrix is computed:

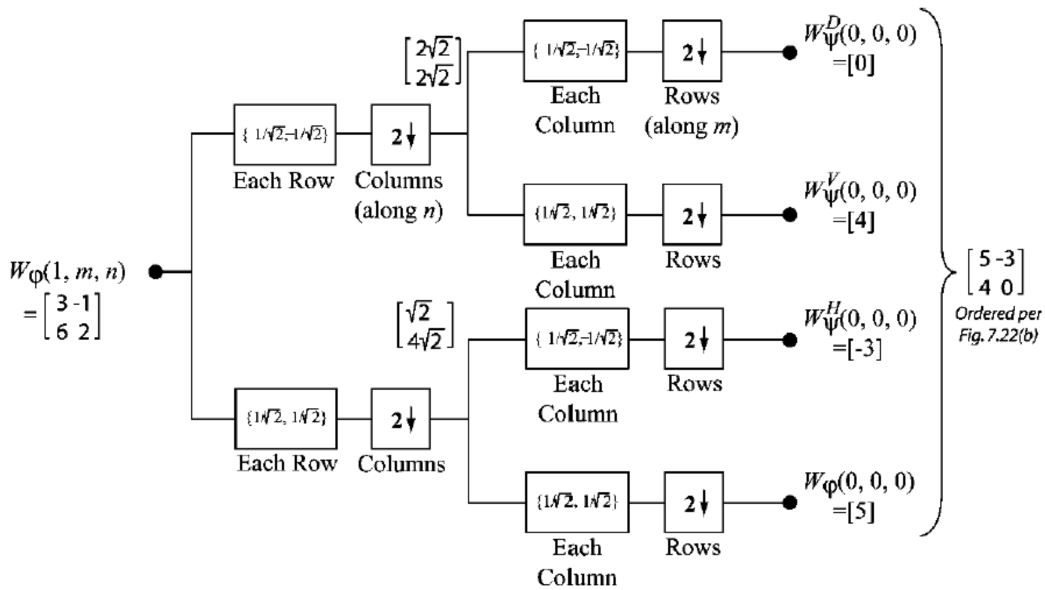
$$\mathbf{H}^{\mathrm{T}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \mathbf{H}.$$

Then,

$$\mathbf{F} = \mathbf{H}^{\mathrm{T}}\mathbf{TH} = (\frac{1}{\sqrt{2}})^2 \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 5 & 4 \\ -3 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 3 & -1 \\ 6 & 2 \end{bmatrix} .$$

3.

One pass through the FWT (fast wavelet transform) 2-d filter bank is all that is required:



4.

The set $\{\psi_{j,k}(x)\}$ of wavelets is defined as

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k).$$

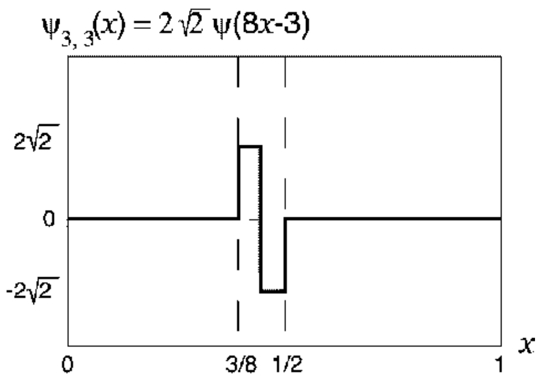
From this definition we obtain

$$\psi_{3,3}(x) = 2^{3/2}\psi(2^3 x - 3) = 2\sqrt{2}\psi(8x - 3).$$

Using the Haar wavelet function definition,

$$\psi(x) = \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & \text{elsewhere,} \end{cases}$$

we obtain the following plot.



To express $\psi_{3,3}(x)$ as a function of scaling functions, we notice that any wavelet function can be expressed as a sum of shifted, double-resolution scaling functions,

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\varphi(2x - n).$$

Employing this result and the Haar wavelet vector defined in the textbook in Example 7.6 ($h_\psi(0) = 1/\sqrt{2}$ and $h_\psi(1) = -1/\sqrt{2}$), we get

$$\psi(8x-3) = \sum_n h_\psi(n)\sqrt{2}\varphi(2(8x-3)-n) = \frac{1}{\sqrt{2}}\sqrt{2}\varphi(16x-6)+\frac{-1}{\sqrt{2}}\sqrt{2}\varphi(16x-7) = \varphi(16x-6)-\varphi(16x-7).$$

Then, since $\psi_{3,3} = 2\sqrt{2}\psi(8x-3)$,

$$\psi_{3,3} = 2\sqrt{2}\psi(8x-3) = 2\sqrt{2}\varphi(16x-6) - 2\sqrt{2}\varphi(16x-7).$$

5.

The DWT transform pair is given as

$$W_\varphi(j_0,k) = \frac{1}{\sqrt{M}}\sum_x f(x)\varphi_{j_0,k}(x)$$
$$W_\psi(j,k) = \frac{1}{\sqrt{M}}\sum_x f(x)\psi_{j,k}(x),$$

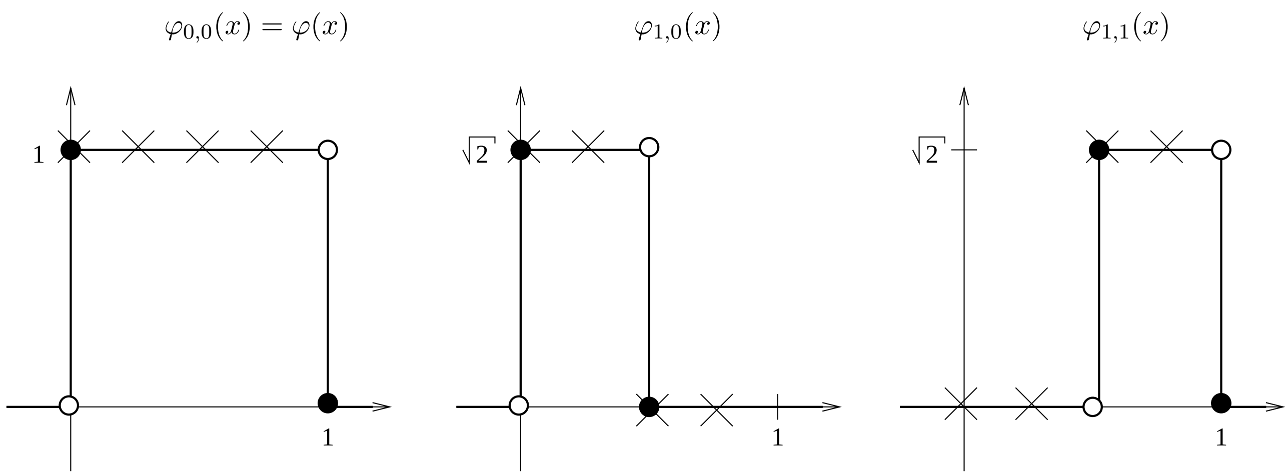
and the inverse transform as

$$f(x) = \frac{1}{\sqrt{M}}\sum_k W_\varphi(j_0,k)\varphi_{j_0,k}(x) + \frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_k W_\psi(j,k)\psi_{j,k}(x).$$

Now $M = 4$, $J = 2$, and $j_0 = 1$, so the summations in the above formulas are performed over $x = 0,1,2,3$, $j = 1$, and $k = 0,1$. Using Haar functions and assuming that they are distributed over the range of the input sequence, we get for the scaling functions

$$\begin{aligned} \varphi_{1,0}(x) &= \sqrt{2}\varphi(2x) \\ \varphi_{1,1}(x) &= \sqrt{2}\varphi(2x-1), \\ \text{where} \quad \varphi(x) &= \begin{cases} 1 & 0 \le x < 1 \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

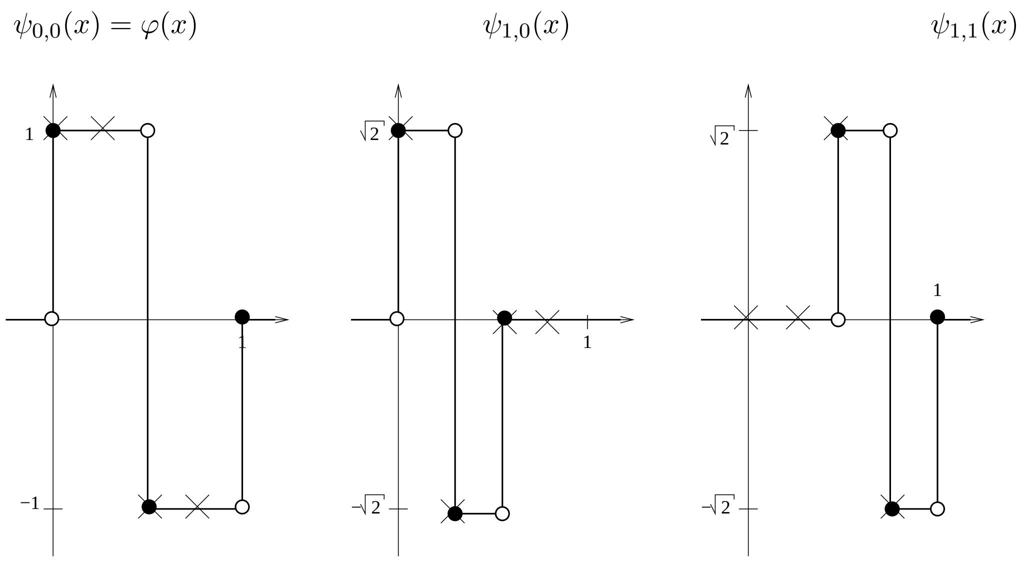
and below is shown the three functions with the four sampling points, i.e. from where we should read values for $x = 0,1,2,3$:



Similarly for the wavelet functions

$$\begin{aligned}
\psi_{1,0}(x) &= \sqrt{2}\psi(2x) \\
\psi_{1,1}(x) &= \sqrt{2}\psi(2x-1), \\
\text{where} \quad \psi(x) &= \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

and the functions with the four sampling points:



Now we can calculate the transform,

$$\begin{aligned}
W_\varphi(1,0) &= \frac{1}{2}[f(0)\varphi_{1,0}(0) + f(1)\varphi_{1,0}(1) + f(2)\varphi_{1,0}(2) + f(3)\varphi_{1,0}(3)] \\
&= \frac{1}{2}[(1)(\sqrt{2}) + (4)(\sqrt{2}) + (-3)(0) + (0)(0)] = \frac{5\sqrt{2}}{2} \\
W_\varphi(1,1) &= \frac{1}{2}[f(0)\varphi_{1,1}(0) + f(1)\varphi_{1,1}(1) + f(2)\varphi_{1,1}(2) + f(3)\varphi_{1,1}(3)] \\
&= \frac{1}{2}[(1)(0) + (4)(0) + (-3)(\sqrt{2}) + (0)(\sqrt{2})] = \frac{-3\sqrt{2}}{2} \\
W_\psi(1,0) &= \frac{1}{2}[f(0)\psi_{1,0}(0) + f(1)\psi_{1,0}(1) + f(2)\psi_{1,0}(2) + f(3)\psi_{1,0}(3)] \\
&= \frac{1}{2}[(1)(\sqrt{2}) + (4)(-\sqrt{2}) + (-3)(0) + (0)(0)] = \frac{-3\sqrt{2}}{2} \\
W_\psi(1,1) &= \frac{1}{2}[f(0)\psi_{1,1}(0) + f(1)\psi_{1,1}(1) + f(2)\psi_{1,1}(2) + f(3)\psi_{1,1}(3)] \\
&= \frac{1}{2}[(1)(0) + (4)(0) + (-3)(\sqrt{2}) + (0)(-\sqrt{2})] = \frac{-3\sqrt{2}}{2}
\end{aligned}$$

so that the DWT is $5\sqrt{2}/2$, $-3\sqrt{2}/2$, $-3\sqrt{2}/2$, $-3\sqrt{2}/2$.

The inverse transform is then calculated:

$$f(x) = \frac{1}{2}[W_\varphi(1,0)\varphi_{1,0}(x) + W_\varphi(1,1)\varphi_{1,1}(x) + W_\psi(1,0)\psi_{1,0}(x) + W_\psi(1,1)\psi_{1,1}(x)],$$

so we get

$$\begin{aligned}
f(0) &= \frac{\sqrt{2}}{4}[(5)(\sqrt{2}) + (-3)(0) + (-3)(\sqrt{2}) + (-3)(0)] = \frac{2(\sqrt{2})^2}{4} = 1\\
f(1) &= \frac{\sqrt{2}}{4}[(5)(\sqrt{2}) + (-3)(0) + (-3)(-\sqrt{2}) + (-3)(0)] = \frac{8(\sqrt{2})^2}{4} = 4\\
f(2) &= \frac{\sqrt{2}}{4}[(5)(0) + (-3)(\sqrt{2}) + (-3)(0) + (-3)(\sqrt{2})] = \frac{-6(\sqrt{2})^2}{4} = -3\\
f(3) &= \frac{\sqrt{2}}{4}[(5)(0) + (-3)(\sqrt{2}) + (-3)(0) + (-3)(-\sqrt{2})] = \frac{0(\sqrt{2})^2}{4} = 0
\end{aligned}$$

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 8/07

Kuvien kompressointi

1. Poistetaan 8 pikselin mittaisesta harmaatasokuvan rivistä {12, 12, 13, 13, 10, 13, 57, 54} redundanttia dataa. Data on kvantisoitu 6-bitin tarkkuudella. Muodosta rivin

   (a) 3 bitin IGS-koodi ja

   (b) mahdollisimman lyhyt koodi.

   Vertaa koodien keskimääräisiä pituuksia suoraan binaarikoodiin, kun harmaatasoja on yhteensä 64. Miten voitaisiin vähentää pikseleiden välistä redundanssia?

2. Lähdeaakkostoon kuuluu 8 symbolia $a_i, i = 1, \ldots, 8$, joiden todennäköisyydet ovat 0.6, 0.2, 0.08, 0.06, 0.02, 0.02, 0.01 ja 0.01. Muodosta aakkostolle

   (a) Huffman-koodi,

   (b) B_2-koodi ja

   (c) S_2-koodi (siirto-koodi, jossa lohkon koko on 2).

   Laske myös entropia ja vertaa keskimääräisiä sananpituuksia siihen.

3. 64 pikselin levyinen binaarikuva on koodattu yksidimensioisella WBS-koodilla, jossa blokit ovat neljän pikselin mittaisia, ja valkoiset blokit merkitty bitillä 0. Esimerkkirivin WBS-koodi on

   $$01100100000010000100100000000,$$

   missä 0 edustaa mustaa pikseliä.

   (a) Dekoodaa rivi.

   (b) Suunnittele yksidimensioinen WBS-proseduuri, joka alkaa valkoisen rivin (64 pikselin mittainen blokki) etsinnällä ja jakaa rekursiivisesti blokin kahtia kunnes päästään neljän pikselin mittaiseen blokkiin.

   (c) Koodaa algoritmillasi esimerkkirivi. Koodi saattaisi olla nyt lyhyempi.

4. Suunnittele algoritmi, jolla voit dekoodata seuraavan LZW-koodatun rivin (esimerkki 8.12 kirjassa):

   $$39\ 39\ 126\ 126\ 256\ 258\ 260\ 259\ 257\ 126$$

   Koska koodaamisessa käytetty sanasto ei ole saatavilla, koodikirja täytyy muodostaa dekoodaamisen yhteydessä.

5. Käytetään ennustavaa koodausta (kuva 8.21) harmaatasojonoon

   $$30\ 29\ 29\ 28\ 20\ 15\ 12\ 10\ 9\ 8\ 9\ 10\ 11\ 11\ 11\ 11\ 11$$

   Muodosta deltamodulaatiokoodi (DM), kun $\alpha = 1$ ja virhe koodataan arvoiksi ± 2. Vertaa tarvittavaa bittimäärää vastaavaan informaation säilyttävään koodaukseen. Mitä huonoja puolia deltamodulaatiolla on?

T-61.5100 Digital image processing, Exercise 8/07

Image compression

1. Consider an 8-pixel line of gray-scale data, {12, 12, 13, 13, 10, 13, 57, 54}, which has been uniformly quantized with 6-bit accuracy. Redundant information should be removed. Construct its

   (a) 3-bit IGS code

   (b) shortest possible code

   Compare average code lengths to the plain binary code, when there are 64 gray levels. How could redundancy between pixels be reduced?

2. The source alphabet has 8 symbols a_i, $i = 1, ..., 8$, whose probabilities are 0.6, 0.2, 0.08, 0.06, 0.02, 0.02, 0.01 and 0.01. Construct for the alphabet its

   (a) Huffman code

   (b) B_2 code

   (c) S_2 code (shift code, block size 2)

   Calculate also entropy and compare average word lengths to it.

3. A 64-bit wide binary image has been coded with one-dimensional WBS code with blocks of four pixels and white blocks are marked with a 0-bit. The WBS code for the example line is

   $$0110010000001000010010000000,$$

   where 0 is used to mark black pixels.

   (a) Decode the line.

   (b) Create a 1-D recursive WBS procedure that begins by looking for all white lines (a 64-pixel block) and successively halves nonwhite intervals until four pixel blocks are reached.

   (c) Use your algorithm to code the previously decoded line. It should require fewer bits.

4. Devise an algorithm for decoding the following LZW encoded line (Example 8.12 in the textbook):

   39 39 126 126 256 258 260 259 257 126

   Since the dictionary that was used during encoding is not available, the code book must be reproduced as the output is decoded.

5. Predictive coding (Fig 8.21) for the gray level line

   30 29 29 28 20 15 12 10 9 8 9 10 11 11 11 11 11

   Create delta modulation code (DM), when $\alpha = 1$ and error is coded using values ± 2. Compare required number of bits to corresponding non-lossy coding. What are the disadvantages of DM?

T-61.5100 Digital image processing, Exercise 8/07

1.

a) IGS (Improved Gray-Scale Quantization) coding adds pseudorandom noise to an image. The noise is generated from the low-order bits of pixel's gray-level values. This added noise reduces false contouring in decoded images.

orig.	*32*	*16*	*8*	*4*	*2*	*1*
12	0	0	1	1	0	0
12	0	0	1	1	0	0
13	0	0	1	1	0	1
13	**0**	**0**	**1**	**1**	**0**	**1**
10	0	0	1	0	1	0
13	0	0	1	1	0	1
57	1	1	1	0	0	1
54	1	1	0	1	1	0

32	*16*	*8*	*4*	*2*	*1*	*IGS*
0	0	1	1	0	0	8
0	1	0	0	0	0	16
0	0	1	**1**	**0**	**1**	8
0	1	0	0	1	0	16
0	0	1	1	0	0	8
0	1	0	0	0	1	16
1	1	1	0	0	1	56
1	1	0	1	1	1	48

On the left are the original gray-levels and their binary representations. On the right are the quantized gray-levels and their binary representations. We start by copying the first row of bits from left to right. Then we add the 3 lowest-order bits to the bits on the second row on the left. This results in the bits on the second row on the right. This procedure is then continued, e.g. the bolded bits on the third row are added to the bolded bits on the fourth row, and thus we get the bits on the fourth row on the right.
If the higher-order bits on the left are full ('111'), then we copy the whole row to the right like in the seventh row. After we are done, the three higher-order bits on the right are the IGS code.

b) The probabilities for the gray levels are:

$$p(13) = 3/8 \qquad p(12) = 2/8 \qquad p(10) = p(54) = p(57) = 1/8$$

The most probable gray-levels are coded with less bits than the rare ones. The code must also be unambiguous. We can choose e.g.

Gray-level	p(Gray-level)	Code
13	3/8	1
12	2/8	01
10	1/8	001
54	1/8	0000
57	1/8	0001

The average code lengths are obtained with Eq. 8.1-4:
a) 3
b) $1 \cdot 3/8 + 2 \cdot 2/8 + 3 \cdot 1/8 + 4 \cdot 1/8 + 4 \cdot 1/8 = 2.25$

The plain binary code: 6.

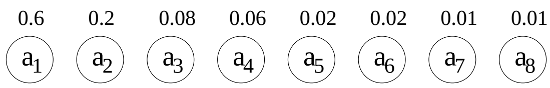
The redundancy between pixels could be reduced with e.g. run-length coding. We could also use differences between neighboring pixels. The difference row would now be

$$\{12,\ 0,\ 1,\ 0,\ -3,\ 3,\ 44,\ -3\}.$$

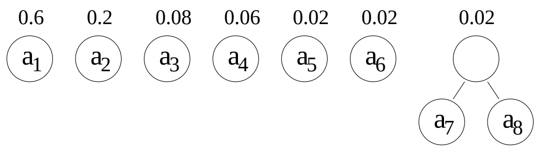
Differences are usually gathered around the origin. However, when we are dealing with such a short and varying sample, we do not gain any advantage when compared to a normal coding.

2.

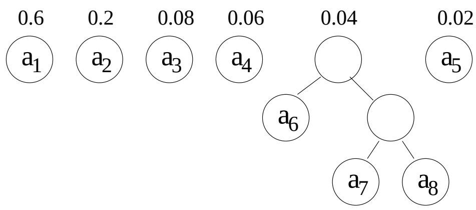
a) The Huffman code is constructed as follows. In the beginning, each symbol is its own tree whose weight is the probability of the symbol. Then with each step we combine two 'lightest' trees (if there are several possibilities, we can choose any of them). The weight of the new tree is the sum of the weights of the combined trees. This procedure is repeated until we end up with only one tree.



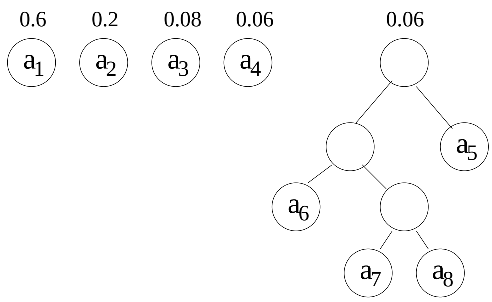
Beginning



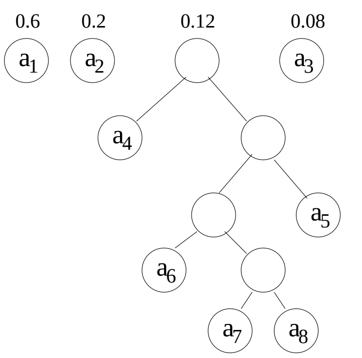
Step 1.



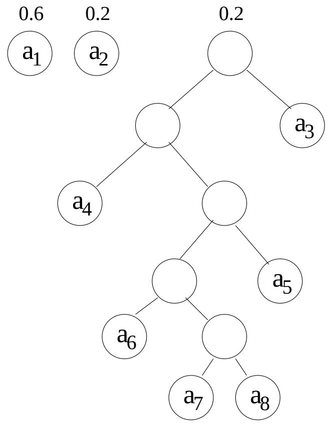
Step 2.



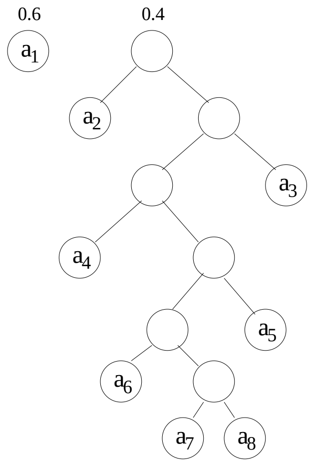
Step 3.



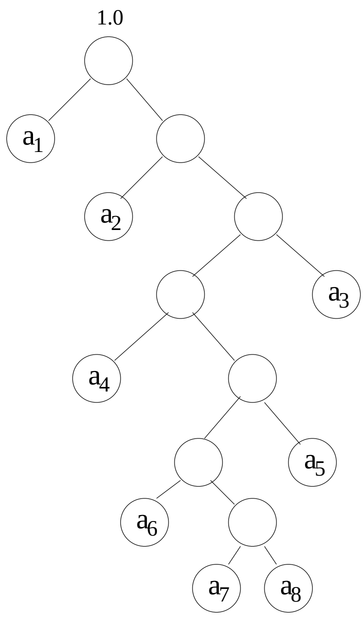
Step 4.



Step 5.



Step 6.



The resulting tree.

Now the binary code of each symbol is the path from the root node to the symbol node. The path to the left is marked with '0 and to the right with '1'. So we obtain:

$$\begin{aligned} a_1 &= 0 \\ a_2 &= 10 \\ a_3 &= 111 \\ a_4 &= 1100 \\ a_5 &= 11011 \\ a_6 &= 110100 \\ a_7 &= 1101010 \\ a_8 &= 1101011 \end{aligned}$$

b) In B_2 code we have an additional bit C. It tells us whether we continue with the current symbol or with the next symbol.

$$\begin{aligned} a_1 &= C00 \\ a_2 &= C01 \\ a_3 &= C10 \\ a_4 &= C11 \\ a_5 &= C00\,C00 \\ a_6 &= C00\,C01 \\ a_7 &= C00\,C10 \\ a_8 &= C00\,C11 \end{aligned}$$

c) In S_2 code we reserve a block of two bits to tell us if we will continue with the current symbol.

$$\begin{aligned} a_1 &= 00 \\ a_2 &= 01 \\ a_3 &= 10 \\ a_4 &= 11\ 00 \\ a_5 &= 11\ 01 \\ a_6 &= 11\ 10 \\ a_7 &= 11\ 11\ 00 \\ a_8 &= 11\ 11\ 01 \end{aligned}$$

The entropy is

$$H = -\sum_{i=1}^{8} P(a_i)\log_2 P(a_i) = -[0.6\log_2 0.6 + \cdots + 0.01\log_2 0.01] = 1.80 \text{ bits/symbol}$$

and the average word lengths are

a) $0.6 \cdot 1 + 0.2 \cdot 2 + 0.08 \cdot 3 + 0.06 \cdot 4 + 0.02 \cdot 5 + 0.02 \cdot 6 + 0.01 \cdot 7 + 0.01 \cdot 7 = 1.84$ bits/symbol.

b) $3 \cdot (0.6 + 0.2 + 0.08 + 0.06) + 6 \cdot (0.02 + 0.02 + 0.01 + 0.01) = 3.18$ bits/symbol.

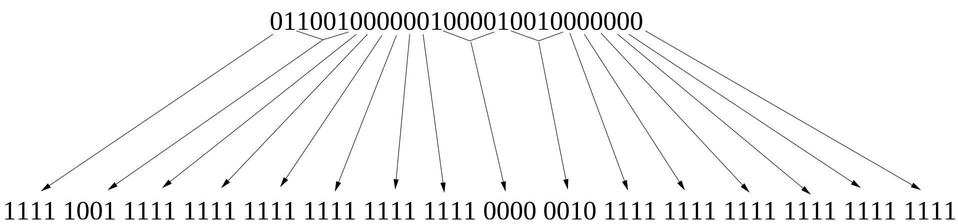
c) $2 \cdot (0.6 + 0.2 + 0.08) + 4 \cdot (0.06 + 0.02 + 0.02) + 6 \cdot (0.01 + 0.01) = 2.28$ bits/symbol.

Huffman code uses less bits (on the average) for each symbol so it can be considered optimal in that sense. As we can see from the results, the word length of Huffman code is very close to entropy. B_2 code is not well suited for this exercise since it uses at least three bits for each symbol. Since we are now having only 8 symbols, they can always be coded with three bits (the plain binary code).

3.

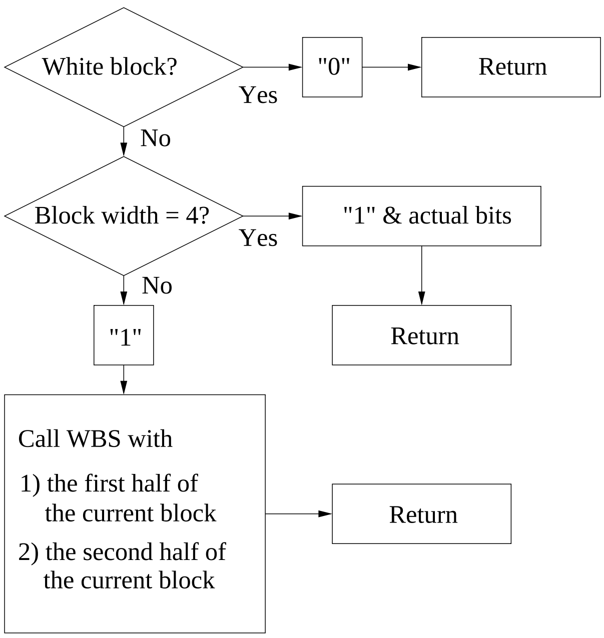
In WBS (White block skipping) code the most common white pixel lines are marked with a short code, e.g. '0'. For other lines we need to have a prefix, e.g. '1'.

a)

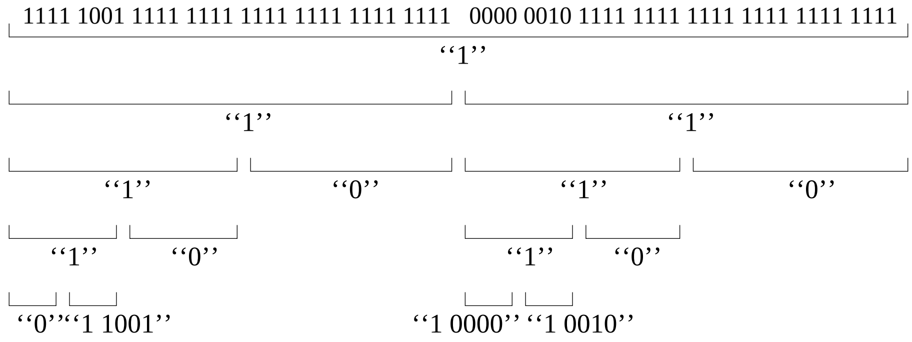


'1' stands for a white pixel in the decoded string and '0' for a black pixel.

b) A recursive program WBS, whose argument is the block that has to be encoded:



c) The given string is encoded:



The encoded result is now:

1111011001001111100001001000

The code length is 27 bits. The original code had 28 bits, so the new code is a little shorter.

4.

The input to the LZW decoding algorithm is

39 39 126 126 256 258 260 259 257 126

The starting dictionary, to be consistent with the coding itself, contains 512 locations - with the first 256 corresponding to gray level values 0 through 255. The decoding algorithm begins by getting the first encoded value, outputting the corresponding value from the dictionary, and setting the "recognized sequence" to the first value. For each additional encoded value, we (1) output the dictionary entry for the pixel value(s), (2) add a new dictionary entry whose content is the "recognized sequence" plus the first element of the encoded value being processed, and (3) set the "recognized sequence" to the encoded value being processed. For the encoded output in Example 8.12, the sequence of operations is:

Recognized	Encoded Value	Pixels	Dict. Address	Dict. Entry
	39	39		
39	39	39	256	39-39
39	126	126	257	39-126
126	126	126	258	126-126
126	256	39-39	259	126-39
256	258	126-126	260	39-39-126
258	260	39-39-126	261	126-126-39
260	259	126-39	262	39-39-126-126
259	257	39-126	263	126-39-39
257	126	126	264	39-126-126

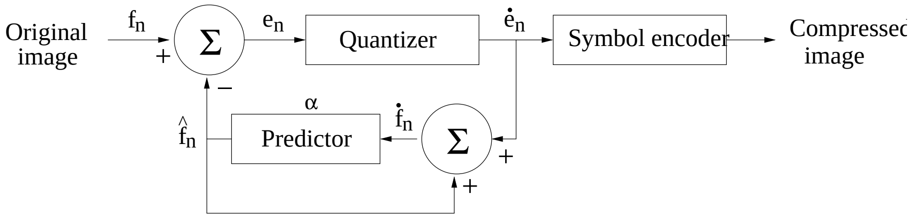
Note, for example, in row 5 of the table that the new dictionary entry for location 259 is 126-39, the concatenation of the currently recognized sequence, 126, and the first element of the encoded value being processed - the 39 from the 39-39 entry in dictionary location 256. The output is then read from the third column of the table to yield

39	39	126	126
39	39	126	126
39	39	126	126
39	39	126	126

where it is assumed that the decoder knows or is given the size of the image that was received. Note that the dictionary is generated as the decoding is carried out.

5.

The encoder:



The predictor:

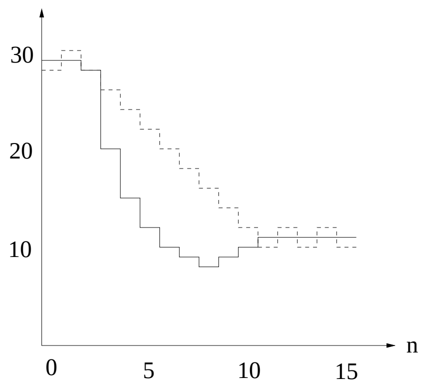
$$\hat{f}_n = \alpha \dot{f}_{n-1}, \quad \alpha = 1$$

The quantizer:

$$\dot{e}_n = \begin{cases} +2, & \text{when } e_n \geq 0 \\ -2, & \text{when } e_n < 0 \end{cases}$$

The initial values are $f_0 = \dot{f}_0 = 30$.

f_n	29	29	28	20	15	12	10	9	8	9	10	11	11	11	11	11
$\hat{f}_n$	30	28	30	28	26	24	22	20	18	16	14	12	10	12	10	12
e	−1	+1	−2	−8	−11	−12	−12	−11	−10	−7	−4	−1	+1	−1	+1	−1
$\dot{e}$	−2	+2	−2	−2	−2	−2	−2	−2	−2	−2	−2	−2	+2	−2	+2	−2
$\dot{f}$	28	30	28	26	24	22	20	18	16	14	12	10	12	10	12	10



It can be seen from the above image that DM cannot follow rapid changes and it will produce noise even in flat areas. The encoding will thus loose information.

In non-lossy or information preserving coding the error must be encoded exactly. If the predictor is given as

$$\hat{f}_n = \alpha f_{n-1} = f_{n-1}$$

then the error is

$$e_n = f_n - \hat{f}_n = f_n - f_{n-1}.$$

The largest change will determine the code length. In the example the largest change is $20-28 = -8$. So we need 4 bits to encode the change if it is assumed that the change can also happen in another direction to $+7$. In DM only one bit is required.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 9/07

Kuvien kompressointi / kuvien segmentointi

1. Optimaalinen kvantisoija muodostetaan asettamalla rekonstruktiotasot eli koodikirjan arvot t_i sellaisiin paikkoihin, että keskimääräinen neliöllinen virhe $\sum_{i=1}^{N} \int_{s_{i-1}}^{s_i} (s - t_i)^2 p(s) ds$ minimoituu. (Tässä $s_0 = -\infty$, $s_N = \infty$, ja $s_i = (t_i + t_{i-1})/2$, kun $i \neq 1$ ja $i \neq N$.) Mihin kaksi rekonstruktiotasoa pitäisi asettaa, jos tiheysfunktiona on

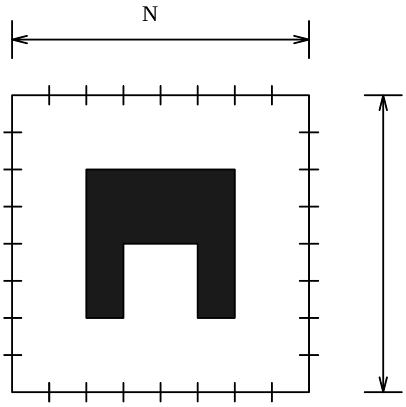
$$p(s) = \begin{cases} s+1, & \text{kun } -1 < s < 0, \\ -s+1, & \text{kun } 0 \leq s < 1, \\ 0 & \text{muutoin} \end{cases}$$

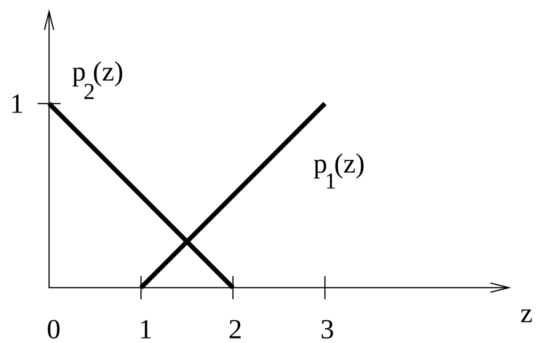
2. Tarkastellaan kirjan kuvassa 8.28(a) esitettyä koodaajaa ja jätetään symbolikooderiosa huomiotta, koska se ei aiheuta mitään virhettä. Olkoon $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ koodattava osakuva. Muodostetaan muunnettu kuva $\boldsymbol{y}$ suorittamalla ensin $\boldsymbol{x}$:lle jokin lineaarinen ortonormaali muunnos (esim. Fourier-muunnos) $\boldsymbol{y}' = \boldsymbol{Ax}$ ja ottamalla sitten $\boldsymbol{y}$:ksi $\boldsymbol{y}'$-vektorin m ensimmäistä komponenttia ja välittämällä nämä kvantisoijalle. Määritellään kuvauksessa syntyvä keskimääräinen neliöllinen katkaisuvirhe $e_m^2 = E\left\{\|\boldsymbol{y} - \boldsymbol{y}'\|^2\right\}$, kvantisointivirhe $e_q^2 = E\left\{\|\boldsymbol{v} - \boldsymbol{y}\|^2\right\}$ ja kokonaisvirhe $e_T^2 = E\left\{\|\boldsymbol{v} - \boldsymbol{y}'\|^2\right\}$, missä $\boldsymbol{v}$ on kvantisoijan ulostulo. Oletetaan, että katkaisu- ja kvantisointivirheet ovat keskenään korreloimattomia ja siten additiivisia.

   (a) Osoita, että katkaisuvirhe on yhtä suuri kuin pois jätettyjä $\boldsymbol{y}'$:n komponentteja vastaava energia, ts. $e_m^2 = \sum_{i=m+1}^{n} E\{y_i'^2\}$.

   (b) Osoita katkaisu- ja kvantisointivirheiden korreloimattomuudesta seuraavan, että $e_T^2 = e_m^2 + e_q^2$.

3. Segmentoi alla (vasemmalla) oleva kuva käyttäen jakamisen ja yhdistämisen tekniikkaa (split-and-merge). Oletetaan $P(R_1)$:n arvon olevan TRUE, jos kaikilla pikseleillä R_1:ssä on sama intensiteetti. Muodosta myös segmentointia vastaava nelipuu.





4. Olkoon kuvan intensiteettijakauma yllä (oikealla) olevan kuvaajan kaltainen, missä $p_1(z)$ vastaa objektien intensiteettiä ja $p_2(z)$ taustan intensiteettiä. Etsi optimaalinen kynnysarvo objektin ja taustan erottamiseksi. Oleta, että $P_1 = P_2$.

5. Oletetaan kuvan koostuvan taustasta ja pienistä toisiaan peittämättömistä kuplista, joiden keskimääräinen harmaataso on $m_1 = 150$ ja varianssi $\sigma_1^2 = 400$. Taustan keskimääräinen harmaataso on $m_2 = 25$ ja varianssi $\sigma_2^2 = 625$. Kuplat peittävät noin 20 % kuvan alasta. Oletetaan sekä taustan että kuplien harmaatasot normaalijakautuneiksi. Kehitä kynnystykseen perustuva menetelmä, jolla kuplat saadaan segmentoitua esiin kuvasta.

T-61.5100 Digital image processing, Exercise 9/07

Image compression / Image segmentation

1. The optimal quantizer is created by setting the reconstruction levels (codebook values) t_i so that the mean square error $\sum_{i=1}^{N}\int_{s_{i-1}}^{s_i}(s-t_i)^2 p(s)ds$ is minimized. (Here $s_0=-\infty$, $s_N=\infty$, and $s_i=(t_i+t_{i-1})/2$, kun $i\neq 1$ ja $i\neq N$.) Where should the two reconstruction levels be placed, if the propability density function is

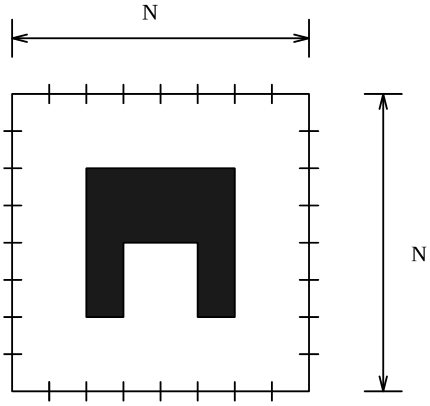
$$p(s)=\begin{cases} s+1, & \text{when } -1<s<0,\\ -s+1, & \text{when } 0\leq s<1,\\ 0 & \text{otherwise}\end{cases}$$

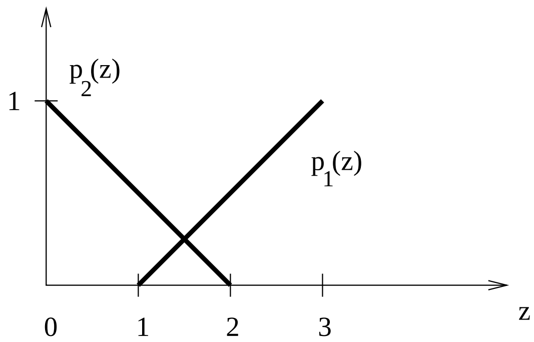
2. Inspect the encoder in Figure 8.28(a) in the textbook and ignore the symbol encoder part, because it causes no error. Let $\boldsymbol{x}=(x_1,x_2,\ldots,x_n)$ be a subimage to be encoded. Transformed image $\boldsymbol{y}$ will be formed from $\boldsymbol{x}$ so that first some linear orthonormed transformation is applied (for example, Fourier transformation) $\boldsymbol{y}'=\boldsymbol{A}\boldsymbol{x}$. After this, $\boldsymbol{y}$ will be replaced by m first components of $\boldsymbol{y}'$ and the result is given to the quantizer. Let the average quadratic truncation error $e_m^2=E\left\{\|\boldsymbol{y}-\boldsymbol{y}'\|^2\right\}$, the quantization error $e_q^2=E\left\{\|\boldsymbol{v}-\boldsymbol{y}\|^2\right\}$, and the total error $e_T^2=E\left\{\|\boldsymbol{v}-\boldsymbol{y}'\|^2\right\}$ be defined as where $\boldsymbol{v}$ is the output of the quantizer. Let us assume that truncation and quantization errors are uncorrelated and thus additive.

   (a) Show that the truncation error equals to the energy corresponding to the excluded components of $\boldsymbol{y}'$ $e_m^2=\sum_{i=m+1}^{n}E\{y_i'^2\}$

   (b) Show that from the fact that truncation and quantization errors are uncorrelated follows $e_T^2=e_m^2+e_q^2$

3. Segment the image shown below (left) using the split and merge procedure. Let $P(R_1)$=TRUE if all pixels in R_1 have the same intensity. Show the quadtree corresponding to your segmentation.





4. Suppose that an image has the intensity distribution as shown above (right), where $p_1(z)$ corresponds to the intensity of the objects and $p_2(z)$ corresponds to the intensity of the background. Assuming that $P_1=P_2$, find the optimal threshold between object and background pixels.

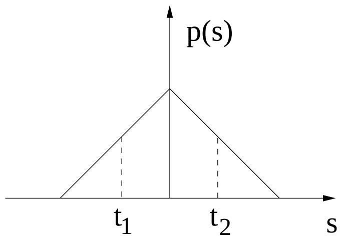
5. Assume that the image consists of small, non-overlapping bubbles, which have a mean grayscale value of $m_1=150$ and a variance $\sigma_1^2=400$. The background has mean $m_2=25$ and variance $\sigma_2^2=625$. The bubbles take up about 20 % of the image. Show a method based on thresholding which separates the bubbles from the background.

T-61.5100 Digital image processing, Exercise 9/07

1.

It is given that

$$p(s) = \begin{cases} s+1, & \text{when } -1 < s < 0 \\ 1-s, & \text{when } 0 \le s < 1 \\ 0, & \text{otherwise} \end{cases}$$



Because $p(s)$ is symmetric around some point (here it is 0), the reconstruction levels of the quantizer $t_i,\ i = 1, 2$ are also set symmetrically around 0.
Minimizing the mean-square error function is equivalent with setting the reconstruction level to its center of gravity. Let us solve t_2 from the following equation:

$$\int_0^{\infty} (s - t_2)p(s)\, ds = 0.$$

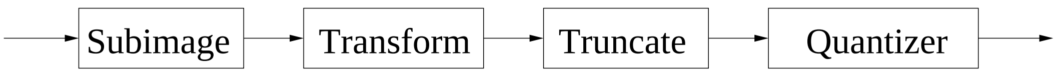
$$\int_0^{\infty} (s-t_2)p(s)\, ds = \int_0^1 (s-t_2)(1-s)\, ds = \int_0^1 (-s^2 + (1+t_2)s - t_2)\, ds = -\frac{1}{3} + \frac{1+t_2}{2} - t_2 = 0$$

$$\Leftrightarrow \quad t_2 = 2(-\frac{1}{3} + \frac{1}{2}) = \frac{1}{3}$$

t_1 is set symmetrically to $t_1 = -1/3$.

2.

The encoder is the following:



$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{y}' = \begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_n \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{bmatrix}$$

a) The average truncation (mapping) error:

$$e_m^2 = E\left\{\|\mathbf{y} - \mathbf{y}'\|^2\right\} = E\left\{\sum_{i=1}^{n} (y_i - y'_i)^2\right\} = \sum_{i=1}^{n} E\left\{(y_i - y'_i)^2\right\}$$

$$= \sum_{i=1}^{m} E\{\underbrace{(y_i - y'_i)}_{0}{}^2\} + \sum_{i=m+1}^{n} E\{\underbrace{(0 - y'_i)}_{(\star)}{}^2\} = \sum_{i=m+1}^{n} E\left\{{y'_i}^2\right\}.$$

$(\star)$ The components that were left out were set as zeros.

b)

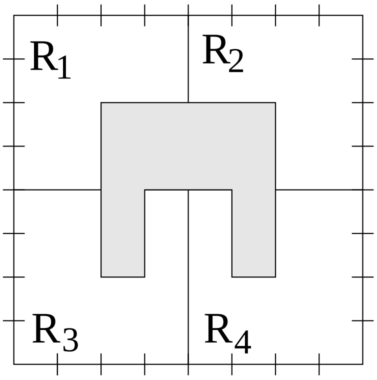
$$\begin{aligned} e_T^2 &= E\left\{\|\mathbf{v}-\mathbf{y}'\|^2\right\} = E\left\{\|(\mathbf{v}-\mathbf{y})+(\mathbf{y}-\mathbf{y}')\|^2\right\} \\ &= E\left\{\|\mathbf{v}-\mathbf{y}\|^2 + 2(\mathbf{v}-\mathbf{y})^T(\mathbf{y}-\mathbf{y}') + \|\mathbf{y}-\mathbf{y}'\|^2\right\} \\ &= E\{\|\mathbf{v}-\mathbf{y}\|^2\} + 2E\{(\mathbf{v}-\mathbf{y})^T(\mathbf{y}-\mathbf{y}')\} + E\{\|\mathbf{y}-\mathbf{y}'\|^2\} \end{aligned}$$

$(\mathbf{v}-\mathbf{y})$ is the quantization error and $(\mathbf{y}-\mathbf{y}')$ the truncation (mapping) error. These are now uncorrelated, so $E\{(\mathbf{v}-\mathbf{y})^T(\mathbf{y}-\mathbf{y}')\} = 0$. Thus

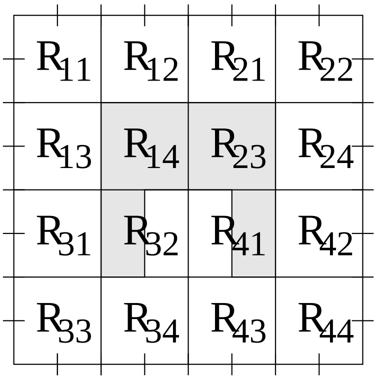
$$e_T^2 = E\{\|\mathbf{v}-\mathbf{y}\|^2\} + E\{\|\mathbf{y}-\mathbf{y}'\|^2\} = e_q^2 + e_m^2$$

3.

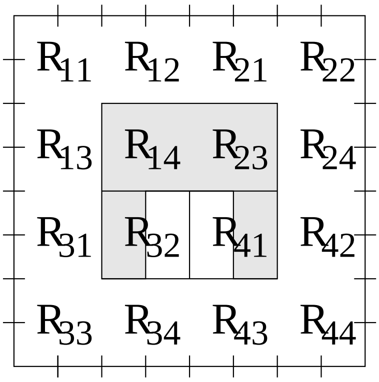
The given image is first divided into four regions:



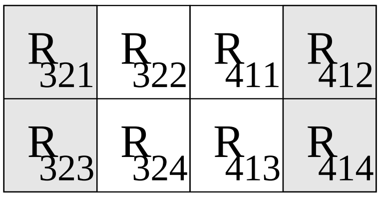
Each region is still non-homogeneous, so each region is further divided into four regions:



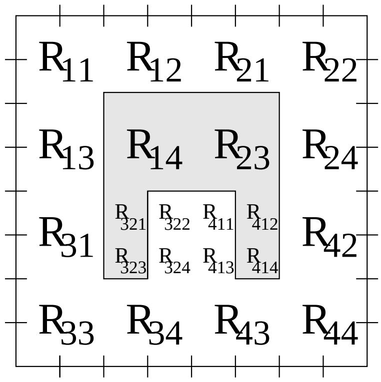
Now the border regions (R_{11}, R_{12}, ...) are all homogenous with the same intensity (i.e. white) and can therefore be merged into one region. The regions R_{14} and R_{23} are also homogenous with the same gray-scale value and are merged as well:



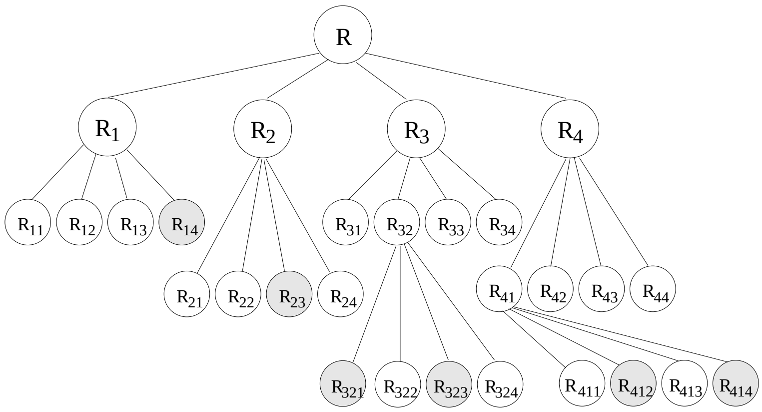
However, the regions R_{32} and R_{41} are not homogeneous and must be divided further. Here is a magnified image of the further division:



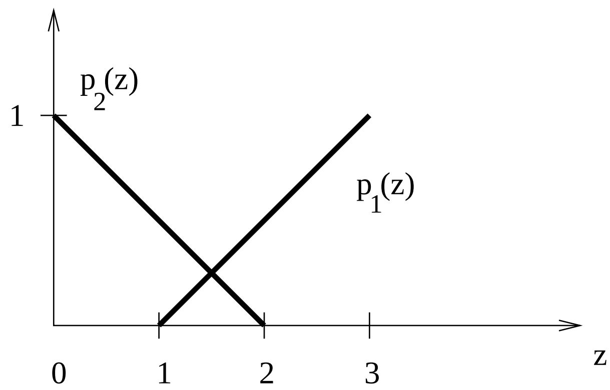
These regions are all homogeneous and can be merged into the two main regions from the previous step, and we are done:



Here is the requested quadtree:



4.



We obtain from the image the following distributions:

$$p_1(z) = \begin{cases} 0, & z < 1 \\ \frac{1}{2}z - \frac{1}{2}, & 1 \leq z \leq 3 \\ 0, & z > 3 \end{cases}$$

$$p_2(z) = \begin{cases} 0, & z < 0 \\ -\frac{1}{2}z + 1, & 0 \leq z \leq 2 \\ 0, & z > 2 \end{cases}$$

Because of the symmetry T=1.5. Theoretical proof follows:

$$\int_0^T p_1(z)dz$$

is the probability for object misclassification, and

$$\int_T^3 p_2(z)dz$$

is the probability for background misclassification.

So, the probability for misclassification is

$$E(T) = P_1 \int_0^T p_1(z)dz + P_2 \int_T^3 p_2(z)dz.$$

Next we minimize $E(T)$. We derive with respect to T and set the result to zero:

$$E'(T) = P_1 p_1(T) - P_2 p_2(T) = 0$$

Thus T must realize

$$P_1 p_1(T) = P_2 p_2(T)$$

Because $P_1 = P_2$,

$$\frac{1}{2}T - \frac{1}{2} = -\frac{1}{2}T + 1 \Leftrightarrow T = \frac{3}{2} = 1.5$$

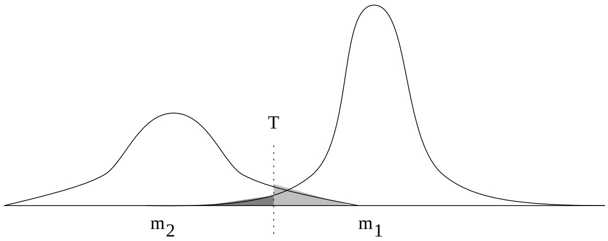
5.

The probability distribution for gray levels in class i is

$$p_i(z) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left[-\frac{(z-\mu_i)^2}{2\sigma_i^2}\right]$$

The probability $p(z)$ for the gray level z in the image is the joint probability of the gray levels of the bubbles and the background, $p(z) = P_1 p_1(z) + P_2 p_2(z)$. P_i is the *a priori* probability for the class i, i.e., the probability that a pixel belongs to class i without knowing its gray level value. So,

$$p(z) = P_1 \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left[-\frac{(z-\mu_1)^2}{2\sigma_1^2}\right] + P_2 \frac{1}{\sqrt{2\pi}\sigma_2} \exp\left[-\frac{(z-\mu_2)^2}{2\sigma_2^2}\right]$$



If the threshold is T, the probability for the error is (see Figure)

$$E(T) = E_1(T) + E_2(T) = P_1 \int_{-\infty}^{T} p_1(z)dz + P_2 \int_{T}^{\infty} p_2(z)dz$$

The error function is minimized by setting its derivative to zero, $E'(T) = P_1 p_1(T) - P_2 p_2(T) = 0$, and we obtain

$$P_1 p_1(T) = P_2 p_2(T).$$

Because the exponentials may be tricky, we take the logarithm on both sides:

$$\ln P_1 p_1(T) - \ln P_2 p_2(T) = \ln \frac{P_1}{\sqrt{2\pi}\sigma_1} - \frac{(T-\mu_1)^2}{2\sigma_1^2} - \ln \frac{P_2}{\sqrt{2\pi}\sigma_2} + \frac{(T-\mu_2)^2}{2\sigma_2^2} = 0,$$

$$\Leftrightarrow -\sigma_2^2(T^2 - 2T\mu_1 + \mu_1^2) + \sigma_1^2(T^2 - 2T\mu_2 + \mu_2^2) + 2\sigma_1^2\sigma_2^2 \ln \frac{\sigma_2 P_1}{\sigma_1 P_2} = 0$$

$$\Leftrightarrow (\sigma_1^2 - \sigma_2^2)T^2 + 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)T + \sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln \frac{\sigma_2 P_1}{\sigma_1 P_2} = 0.$$

When we now place the given values in the equation and observe also that $P_1/P_2 = 20\%/80\% = 0.25$, we obtain $-225T^2 - 167500T - 1.4506 \cdot 10^7 = 0$, and the final result is $\underline{T_1 \approx 100}$ and $\underline{T_2 \approx 644}$. The larger solution is due to the fact that the distribution of the bubbles is sharper (the variance is smaller) and on the right it will go (again!) under the distribution of the background.

T-61.5100 Digitaalinen kuvankäsittely, Harjoitus 10/07

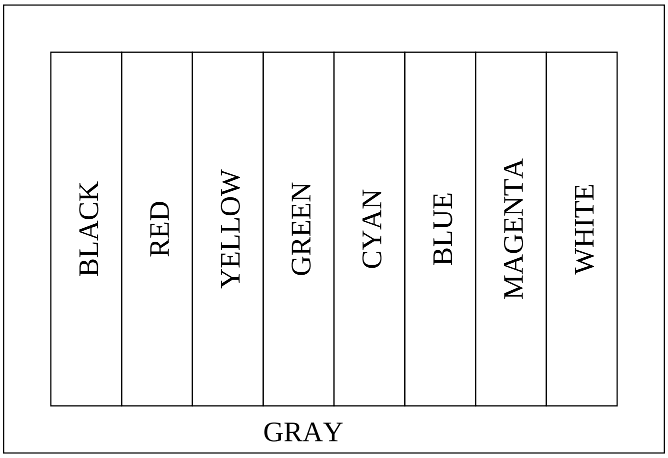
Värikuvien käsittely

1. (a) Oletetaan annetuksi kaksi väriä, c_1 ja c_2, joiden koordinaatit kromaattisuusdiagrammissa (kuva 6.5 kirjassa) ovat (x_1, y_1) ja (x_2, y_2). Johda yleinen yhtälö(t) c_1:n ja c_2:n suhteellisille prosenttiosuuksille värissä c, kun tämä sijaitsee näitä värejä yhdistävällä suoralla.

   (b) Lisätään tarkasteluun kolmas väri c_3, jonka koordinaatit ovat (x_3, y_3). Johda yleinen yhtälö(t) c_1:n, c_2:n, ja c_3:n suhteellisille prosenttiosuuksille värissä c, kun tämä sijaitsee näiden kolmen värin määräämän kolmion sisällä.

2. (a) Hahmottele, miltä oheisen kuvan RGB-komponentit näyttäisivät mustavalkomonitorilla. (Alueiden värit on nimetty kuvaan, värikuva löytyy kirjan sivulta 345). Kaikilla väreillä on maksimi intensiteetti ja saturaatio. Keskiharmaa reunus kuuluu myös kuvaan.

   (b) Hahmottele myös, miltä oheisen kuvan HSI-komponentit näyttäisivät mustavalkomonitorilla.



3. Johda CMY intensiteettimuunnosfunktio

$$s_i = kr_i + (1 - k), \quad i = 1, 2, 3 \quad (\text{for } C, M, Y),$$

vastaavasta RGB yhtälöstä

$$s_i = kr_i, \quad i = 1, 2, 3 \quad (\text{for } R, B, G).$$

4. CIELab-värikoordinaatistossa (1976) määritellään L^*, a^* ja b^* seuraavasti:

$$L^* = 25(100Y/Y_0)^{1/3} - 16, \quad 1 \leq 100Y \leq 100$$
$$a^* = 500[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}]$$
$$b^* = 200[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}]$$

$$(\Delta s)^2 = (\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2$$

Kuinka Δs muuttuu, kun

(a) X, Y ja Z muuttuvat 5%?

(b) X, Y ja Z muuttuvat 10%?

(c) X_0, Y_0 ja Z_0 muuttuvat 10%? X, Y ja Z pysyvät vakioina.

Mitä tästä voidaan päätellä?

T-61.5100 Digital image processing, Exercise 10/07

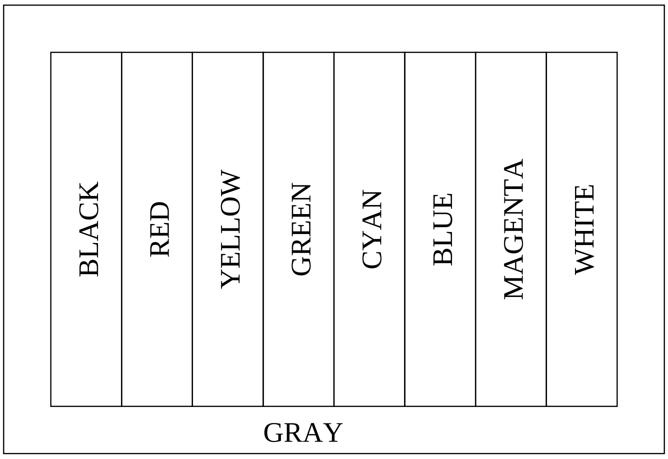
Color image processing

1. (a) Consider any two valid colors c_1 and c_2 with coordinates (x_1, y_1) and (x_2, y_2) in the chromaticity diagram of Fig. 6.5 in the textbook. Derive the necessary general expression(s) for computing the relative percentages of colors c_1 and c_2 composing a given color c that is known to lie on the straight line joining these two colors.

   (b) A third color c_3 with coordinates (x_3, y_3) is then added. Derive the necessary general expression(s) for computing the relative percentages of c_1, c_2, and c_3 composing a given color c that is known to lie within the triangle whose vertices are at the coordinates of these three colors.

2. (a) Sketch the RGB components of the following image as they would appear on a monochrome monitor. (The color names are printed on each area in the image, see the true color image in p. 345 in the textbook). All colors are at maximum intensity and saturation. Consider the middle gray border as part of the image.

   (b) Sketch also the HSI components as they would appear on a monochrome monitor.



3. Derive the CMY intensity transformation function

$$s_i = kr_i + (1 - k), \quad i = 1, 2, 3 \quad (\text{for } C, M, Y),$$

from its RGB counterpart

$$s_i = kr_i, \quad i = 1, 2, 3 \quad (\text{for } R, B, G).$$

4. L^*, a^*, and b^* are defined in the CIELab colour coordinates (1976) in the following way:

$$L^* = 25(100Y/Y_0)^{1/3} - 16, \quad 1 \leq 100Y \leq 100$$
$$a^* = 500[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}]$$
$$b^* = 200[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}]$$

$$(\Delta s)^2 = (\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2$$

Determine Δs when

(a) X, Y, and Z are changed by 5%?

(b) X, Y, and Z are changed by 10%?

(c) X_0, Y_0, and Z_0 are changed by 10%? X, Y, and Z are assumed to be constant.

What can be said about Δs?

T-61.5100 Digital image processing, Exercise 10/07

1.

a) Denote by c the given color and let its coordinates be denoted by (x_0, y_0). The distance between c and c_1 is

$$d(c, c_1) = [(x_0 - x_1)^2 + (y_0 - y_1)^2]^{1/2}.$$

Similarly the distance between c_1 and c_2 is

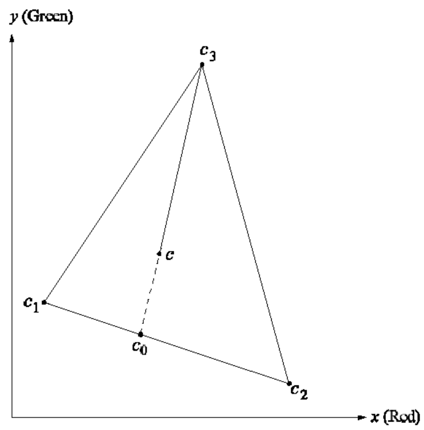
$$d(c_1, c_2) = [(x_1 - x_2)^2 + (y_1 - y_2)^2]^{1/2}.$$

The percentage p_1 of c_1 in c is

$$p_1 = \frac{d(c_1, c_2) - d(c, c_1)}{d(c_1, c_2)} \times 100.$$

The percentage p_2 of c_2 is simply $p_2 = 100 - p_1$. In the preceding equation we see, for example, that when $c = c_1$, then $d(c, c_1) = 0$ and it follows that $p_1 = 100\%$ and $p_2 = 0\%$. Similarly, when $d(c, c_1) = d(c_1, c_2)$, it follows that $p_1 = 0\%$ and $p_2 = 100\%$. Values in between are easily seen to follow from these simple relations.

b) Consider the following figure, in which c_1, c_2, and c_3 are the given vertices of the color triangle and c is an arbitrary color point contained within the triangle or on its boundary. The key to solving this problem is to realize that any color on the border of the triangle is made up of proportions from the two vertices defining the line segment that contains the point. The contribution to a point on the line by the color vertex opposite this line is 0%.



The line segment connecting points c_3 and c is shown extended (dashed segment) until it intersects the line segment connecting c_1 and c_2. The point of intersection is denoted c_0 . Because we have the values of c_1 and c_2, if we knew c_0, we could compute the percentages of c_1 and c_2 contained in c_0 by using the method in a). Denote the ratio of the content of c_1 and c_2 in c_0 by R_{12}. If we now add color c_3 to c_0, we know from a) that the point will start to move toward c_3 along the line shown. For any position of a point along this line we could determine the percentage of c_3 and c_0, again, by using the method described in a). The ratio R_{12} will remain the same for any point along the segment connecting c_3 and c_0.

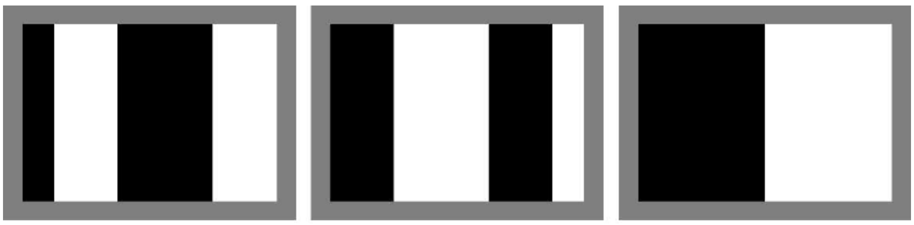
So, if we can obtain c_0, we can then determine the ratio R_{12}, and the percentage of c_3, in color c. The point c_0 is not difficult to obtain. Let $y = a_{12}x + b_{12}$ be the straight line containing points c_1 and c_2, and $y = a_{3c}x + b_{3c}$ the line containing c_3 and c. The intersection of these two lines gives the coordinates of c_0. The lines can be determined uniquely because we know the coordinates of the two point pairs needed to determine the line coefficients. Solving for the intersection in terms of these coordinates is straightforward, but tedious. Our interest here is

in the fundamental method, not the mechanics of manipulating simple equations so we do not give the details.
At this juncture we have the percentage of c_3 and the ratio between c_1 and c_2. Let the percentages of these three colors composing c be denoted by p_1, p_2, and p_3 respectively. Since we know that $p_1 + p_2 = 100 - p_3$, and that $p_1/p_2 = R_{12}$, we can solve for p_1 and p_2. Finally, note that this problem could have been solved the same way by intersecting one of the other two sides of the triangle. Going to another side would be necessary, for example, if the line we used in the preceding discussion had an infinite slope. Also a simple test to determine if the color of c is equal to any of the vertices should be the first step in the procedure, in this case no additional calculations would be required.

2.

a) For the given image, the maximum intensity and saturation requirement means that the RGB component values are 0 or 1. We can create the following table with 0 and 255 representing black and white, respectively.

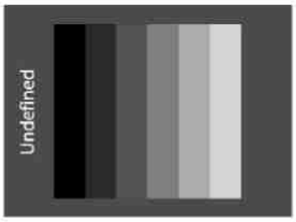
Color	R	G	B	Mono R	Mono G	Mono B
Black	0	0	0	0	0	0
Red	1	0	0	255	0	0
Yellow	1	1	0	255	255	0
Green	0	1	0	0	255	0
Cyan	0	1	1	0	255	255
Blue	0	0	1	0	0	255
Magenta	1	0	1	255	0	255
White	1	1	1	255	255	255
Gray	0.5	0.5	0.5	128	128	128



Red component Green component Blue component

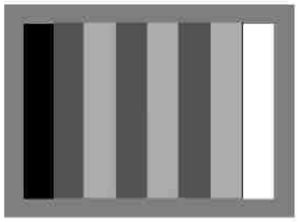
b) Using Eqs. (6.2-2) through (6.2-4), we get the results shown in the following:

Color	R	G	B	H	S	I	Mono H	Mono S	Mono I
Black	0	0	0	-	-	0	-	-	0
Red	1	0	0	0	1	0.33	0	255	85
Yellow	1	1	0	0.17	1	0.67	43	255	170
Green	0	1	0	0.33	1	0.33	85	255	85
Cyan	0	1	1	0.5	1	0.67	128	255	170
Blue	0	0	1	0.67	1	0.33	170	255	85
Magenta	1	0	1	0.83	1	0.67	213	255	170
White	1	1	1	-	0	1	-	0	255
Gray	0.5	0.5	0.5	-	0	0.5	-	0	128



Hue component

Saturation component



Intensity component

Note that, in accordance with Eq. (6.2-2), hue is undefined when $R = G = B$ since $\theta = \cos^{-1}(\frac{0}{0})$. In addition, saturation is undefined when $R = G = B = 0$ since Eq. (6.2-3) yields $S = 1 - \frac{3\min(0)}{3 \cdot 0} = 1 - \frac{0}{0}$.

3.

Each component of the CMY image is a function of a single component of the corresponding RGB image,

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}.$$

In RGB color space, we know that

$$s_i = kr_i, \quad i = 1, 2, 3 \quad (\text{for } R, B, G).$$

And from the first equation, we know that the CMY components corresponding to the r_i and s_i are (for clarity, we will use a prime to denote the CMY components)

$$r_i^{'} = 1 - r_i \quad \text{and} \quad s_i^{'} = 1 - s_i.$$

Thus,

$$r_i = 1 - r_i^{'}$$

and

$$s_i^{'} = 1 - s_i = 1 - kr_i = 1 - k(1 - r_i^{'}) = kr_i^{'} + (1 - k).$$

4.

CIELab color coordinates for device-independent color representation:

$$\begin{aligned} \text{brightness:} \quad & L^* = 25(100Y/Y_0)^{1/3} - 16, \ \ 1 \le 100Y \le 100 \\ \text{red-green content:} \quad & a^* = 500[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}], \\ \text{green-blue content:} \quad & b^* = 200[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}]. \end{aligned}$$

The values X, Y, and Z refer to intensities of red, green, and blue perceived by a subject. The set X_0, Y_0, and Z_0 is the tristimulus value of the reference white (e.g., zinc sulfate). (Note that the above definition is a bit different from the textbook's definition (p. 322). The condition under which the above equations are valid ($1 \le 100Y \le 100$) is the only difference.)

a) The new values of X, Y, and Z are $X_n = (1 + p)X = PX$, $Y_n = PY$, and $Z_n = PZ$. The change in L^* is given by

$$\Delta L^* = 25 \cdot 10^{2/3}[(PY/Y_0)^{1/3} - (Y/Y_0)^{1/3}] = 25 \cdot 10^{2/3}(P^{1/3} - 1)(Y/Y_0)^{1/3}.$$

The change in a^* is given by

$$\begin{aligned}\Delta a^* &= 500[(PX/X_0)^{1/3} - (PY/Y_0)^{1/3} - (X/X_0)^{1/3} + (Y/Y_0)^{1/3}] \\ &= 500(P^{1/3} - 1)[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}].\end{aligned}$$

The change in b^* is given by

$$\Delta b^* = 200(P^{1/3} - 1)[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}].$$

The change in s is determined by

$$(\Delta s)^2 = (\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2 = (P^{1/3} - 1)^2 A(X, Y, Z)$$
$$\Leftrightarrow |\Delta s| = |P^{1/3} - 1|\sqrt{A(X, Y, Z)}.$$

If values of X, Y, and Z are increased by 5 %, i.e., $P = 1.05$, the resulting change in s is

$$|\Delta s| = |1.05^{1/3} - 1|\sqrt{A(X, Y, Z)} = 0.0164\sqrt{A(X, Y, Z)}.$$

The relationship between the change in intensities of X, Y, and Z and in s is not linear.

b) For a 10 % change in X, Y, and Z:

$$|\Delta s| = |1.1^{1/3} - 1|\sqrt{A(X, Y, Z)} = 0.0323\sqrt{A(X, Y, Z)}.$$

c) The values of X_0, Y_0, and Z_0 are changed by 10 %. The changes in L^*, a^*, and b^* are

$$\begin{aligned}\Delta L^* &= 25 \cdot 10^{2/3}[(Y/PY_0)^{1/3} - (Y/Y_0)^{1/3}] \\ &= 25 \cdot 10^{2/3}(P^{-1/3} - 1)(Y/Y_0)^{1/3}, \\ \Delta a^* &= 500(P^{-1/3} - 1)[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}], \\ \Delta b^* &= 200(P^{-1/3} - 1)[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}].\end{aligned}$$

The absolute change in s is

$$|\Delta s| = |P^{-1/3} - 1|\sqrt{A(X_0, Y_0, Z_0)}.$$

For a 10 % increase in X_0, Y_0, and Z_0

$$|\Delta s| = |1.1^{-1/3} - 1|\sqrt{A(X_0, Y_0, Z_0)} = 0.0313\sqrt{A(X_0, Y_0, Z_0)}.$$