T-61.5070 COMPUTER VISION, Exercise 4/08

1.

Edge detecting spatial filters are described in Sec. 4.3.2, pp. 77–83. The convolution masks for *Roberts operator* are

$$h_1 : \begin{array}{|r|r|}\hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \text{ and } h_2 : \begin{array}{|r|r|}\hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}.$$

Sobel operator defines two convolution masks suitable for the detection of horizontal and vertical edges:

$$h_1 : \begin{array}{|r|r|r|}\hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array} \text{ and } h_2 : \begin{array}{|r|r|r|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}.$$

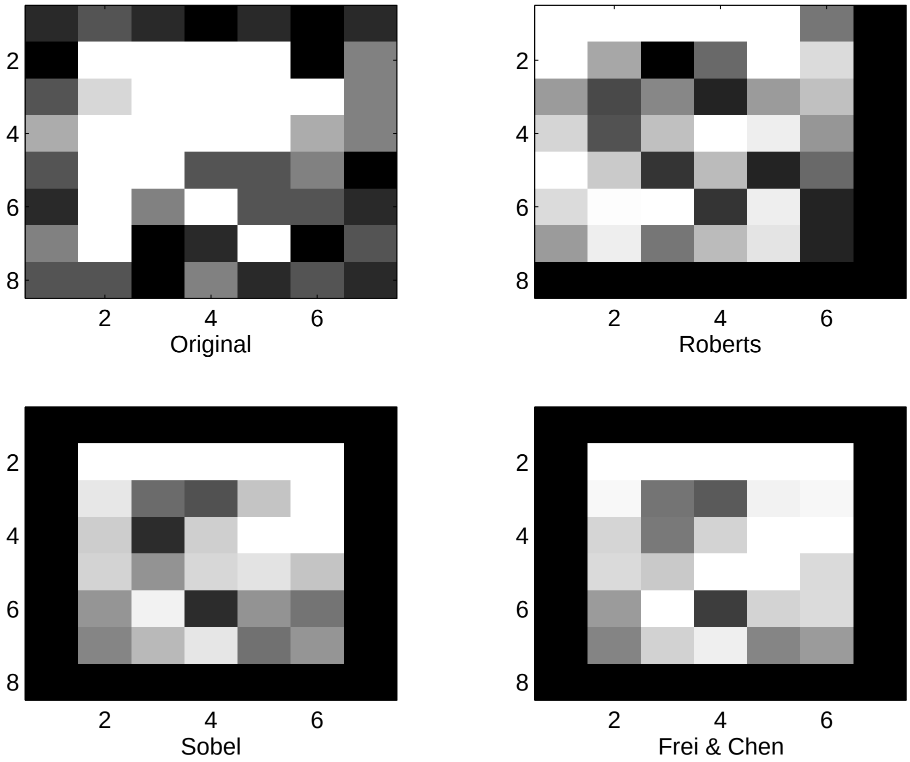
Frei and Chen used a set of nine orthogonal masks to detect edges and lines, or neighborhoods without edges and lines. Four of the nine masks are suitable for edge detection:

$$h_1 : \begin{array}{|r|r|r|}\hline 1 & \sqrt{2} & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -\sqrt{2} & -1 \\ \hline \end{array}, h_2 : \begin{array}{|r|r|r|}\hline 1 & 0 & -1 \\ \hline \sqrt{2} & 0 & -\sqrt{2} \\ \hline 1 & 0 & -1 \\ \hline \end{array}, h_3 : \begin{array}{|r|r|r|}\hline 0 & -1 & \sqrt{2} \\ \hline 1 & 0 & -1 \\ \hline -\sqrt{2} & 1 & 0 \\ \hline \end{array}, \text{ and } h_4 : \begin{array}{|r|r|r|}\hline \sqrt{2} & -1 & 0 \\ \hline -1 & 0 & 1 \\ \hline 0 & 1 & -\sqrt{2} \\ \hline \end{array}.$$

The output of mask h_k is given by

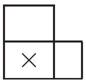
$$f_k(i,j) = \sum_{(m,n)} \sum_{\in R} h_k(i-m, j-n) g(m,n),$$

where $g(m,n)$ is the image pixel and R defines the local neighborhood of the pixel $f_k(i,j)$. The L_2 norm of the mask outputs, $\sqrt{\sum_k f_k^2}$, was used to describe edginess in the response images shown below. Sometimes, the simpler L_1 norm ($\sum_k |f_k|$) is used as an approximation altough the systematic error increases with dimension.



2.

Median smoothing (pp. 74–76) ranks local pixel intensities and replaces the gray level of each pixel by the median of the gray levels in a neighborhood of that pixel. Define the support for a nonsymmetric median smoothing mask as



As shown by the following example, a nonsymmetric median filter may shift edge locations.

$$\begin{array}{ccc} \text{Original image} & & \text{Filtered image} \\ \begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{matrix} & \rightarrow & \begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{matrix} \end{array}$$

A symmetric median smoothing mask, e.g., $\boxed{}\boxed{\times}\boxed{}$ preserves edge locations.

3.

Some methods for finding edges in multispectral images are described in Sec. 4.3.7., p. 94. The edge detector of Cervenka and Charvat, Eq. (4.68) in the textbook, is not appropriate for 3-D images because it does not take the gradient in z-direction into account. In 3-D data object boundaries are surfaces; edge elements in two dimensions become surface elements in three dimensions. The two-dimensional image gradient, when generalized to three dimensions, is the local surface normal. For a given volume image $f(x, y, z)$, the gradient of f is the vector defined by

$$\nabla f = \frac{\partial f}{\partial x}\boldsymbol{i} + \frac{\partial f}{\partial y}\boldsymbol{j} + \frac{\partial f}{\partial z}\boldsymbol{k},$$

where $\boldsymbol{i} = [1\ 0\ 0]^T$, $\boldsymbol{j} = [0\ 1\ 0]^T$, and $\boldsymbol{k} = [0\ 0\ 1]^T$ are the unit vectors. A simple measure of 3-D edginess is the magnitude of the surface gradient computed from differences approximating the first derivatives in the directions of x, y, and z:

$$M = \sqrt{(\Delta_x f)^2 + (\Delta_y f)^2 + (\Delta_z f)^2},$$

where

$$\begin{aligned} \Delta_x f &= f(x+1, y, z) - f(x, y, z), \\ \Delta_y f &= f(x, y+1, z) - f(x, y, z), \\ \Delta_z f &= f(x, y, z+1) - f(x, y, z). \end{aligned}$$

This method is applied to multispectral images by forming the z-component using the n spectral bands $f_0(x, y)$, $f_1(x, y)$, $\cdots$, $f_{n-1}(x, y)$:

$$f(x, y, z) = f_z(x, y)$$

The brightness difference of the same pixel in two adjacent spectral bands, $\Delta_z f$, is very informative. (An example: The AVHRR instrument of NOAA weather satellites has five spectral channels. Some object characteristics are emphasized by differences between adjacent spectral bands, for example: The difference $f_1(x, y) - f_0(x, y)$ serves the detection of vegetation and snow,and discrimination between land and sea. The difference $f_3(x, y) - f_2(x, y)$ serves the detection of cirrus clouds and the discrimination between watercloud and surface, and between snow and cloud. The difference $f_4(x, y) - f_3(x, y)$ serves the detection of precipitation clouds.)

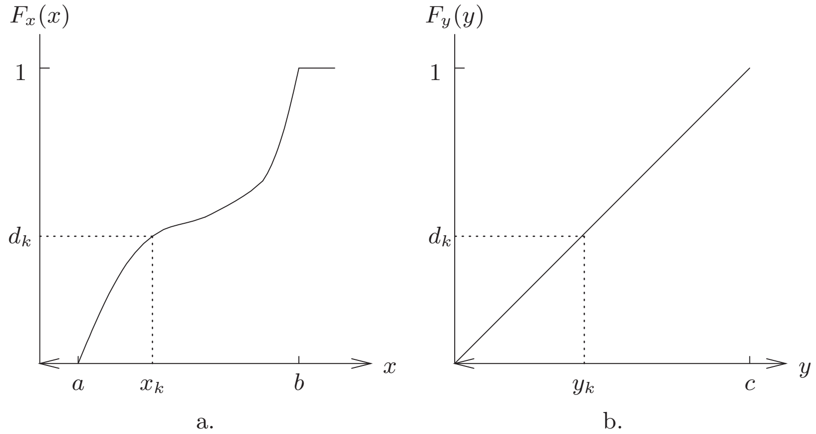


Figure 1: Distribution function of some $f_x(x)$ (a.) and distribution function of constant density $f_y(y)$ (b.).

4.

Histogram equalization is described in Sec. 4.1.2., pp. 59–62 in the textbook. A function $y = g(x)$ is determined for transforming the non-uniform density function of x, $f_x(x)$, into the constant density function of y, $f_y(y) =$ k. A cumulative distribution function of some $f_x(x)$, $F_x(x)$, is depicted in Fig. 1.a. A distribution function of constant density $f_y(y)$, $F_y(y)$, is depicted in Fig. 1.b. The intensity transform $y = g(x)$ should be found which maps a value of x to a value of y so that $F_y(y) = F_x(x)$. Solve for y:

$$\begin{aligned}
F_y(y) = \int_0^y f_y(y)dy &= \int_a^x f_x(x)dx = F_x(x). \\
\int_0^y kdy &= \int_a^x f_x(x)dx \\
ky &= \int_a^x f_x(x)dx \\
y &= \frac{1}{k}\int_a^x f_x(x)dx.
\end{aligned}$$

The normalization coefficient k can be determined from

$$F_y(c) = \int_0^c f_y(y)dy = 1 \Leftrightarrow kc = 1 \Leftrightarrow k = \frac{1}{c}.$$

Therefore,

$$y = g(x) = c\int_a^x f_x(x)dx = cF_x(x).$$

The histogram equalization algorithm is as follows:

1. Compute the histogram of intensity levels $f_x(x)$ in the input image.
2. Sum $f_x(x)$ to obtain the distribution function $F_x(x)$.
3. Use $F_x(x)$ as the intensity transformation function $g(x)$; that is,

   $$y = cF_x(x),$$

   where c is the largest intensity level in the transformed histogram.