The Hit-or-Miss Transformation

The hit-or-miss transformation of an image *A* by *B* is denoted by $A \circledast B$. *B* is a pair of structuring elements $B=(B_1,B_2)$ rather than a single element.

B1: set of elements of *B* associated with an object

B2 : set of elements of *B* associated with the background

The hit-or-miss transform is defined as follows:

$$A \circledast B = (A \ominus B_1) \cap (A^c \ominus B_2)$$

This transform is useful in locating all pixel configurations that match the B_1 structure (i.e a hit) but do not match that of B_2 (i.e. a miss). Thus, the hit-or-miss transform is used for shape detection.

Example: Use the hit-or-miss transform to identify the locations of the following shape pixel configuration in the image below using the two structuring elements *B1* and *B2*.

$$\begin{matrix}0&1&0\\1&1&1\\0&1&0\end{matrix}$$

Shap

$$\begin{matrix}0&0&0&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0&0\\0&0&1&0&0&1&1&1&1&0&0\\0&1&1&1&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&1&1&0&0\\0&0&0&0&1&0&0&1&1&1&0\\0&0&0&1&1&1&0&0&1&0&0\\0&0&0&0&1&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0&0\end{matrix}$$

Image *A*

$$\begin{matrix}&1&\\1&\boxed{1}&1\\&1&\end{matrix}$$

B_1

$$\begin{matrix}1&&1\\&\boxed{}&\\1&&1\end{matrix}$$

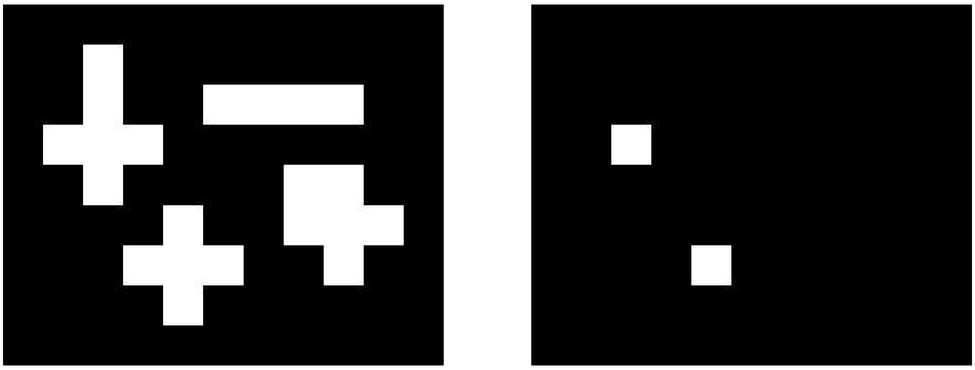
B_2

Solution:

$$A \ominus B_1 = \begin{matrix} 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \end{matrix} \qquad A^c = \begin{matrix} 1&1&1&1&1&1&1&1&1&1&1 \\ 1&1&0&1&1&1&1&1&1&1&1 \\ 1&1&0&1&1&0&0&0&0&1&1 \\ 1&0&0&0&1&1&1&1&1&1&1 \\ 1&1&0&1&1&1&1&0&0&1&1 \\ 1&1&1&1&0&1&1&0&0&0&1 \\ 1&1&1&0&0&0&1&1&0&1&1 \\ 1&1&1&1&0&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1&1&1&1 \end{matrix}$$

$$A^c \ominus B_2 = \begin{matrix} 1&0&1&0&1&1&1&1&1&1&1 \\ 1&0&1&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&1&1&1&1&1&1 \\ 1&0&1&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&1 \\ 1&1&1&0&1&0&0&0&0&0&0 \\ 1&1&0&0&0&0&0&0&1&0&1 \\ 1&1&1&0&1&0&1&1&1&1&1 \end{matrix} \qquad A \circledast B = \begin{matrix} 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&\mathbf{1}&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&\mathbf{1}&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0 \end{matrix}$$

The figure below shows an example of applying the hit-or-miss transform on the image in the previous example.



(a) (b)

Figure 12.1 (a) Binary image. (b) Result of applying hit-or-miss transform.

Basic Morphological Algorithms (Applications)

The principle application of morphology is extracting image components that are useful in the representation and description of shape. Morphological algorithms are used for boundaries extraction, skeletonization (i.e. extracting the skeleton of an object), and thinning.

Boundary Extraction

The boundary of a set A, denoted by β(A), can be obtained by:

$$\beta(A) = A - (A \ominus B)$$

where B is the structuring element.

The figure below shows an example of extracting the boundary of an object in a binary image.

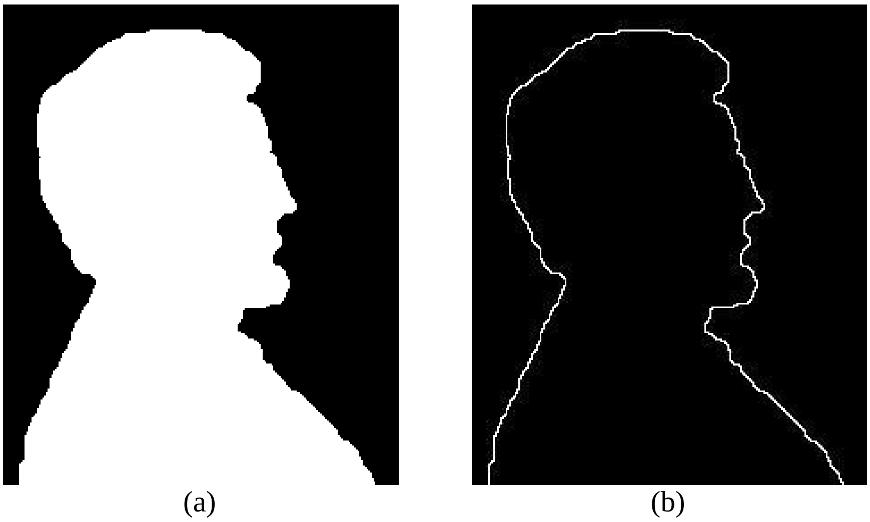


Figure 12.2 (a) Binary image. (b) Object boundary extracted using the previous equation and 3×3 square structuring element.

Note that, because the size of structuring element is 3×3 pixels, the resulted boundary is one pixel thick. Thus, using 5×5 structuring element will produce a boundary between 2 and 3 pixels thick as shown in the next figure.

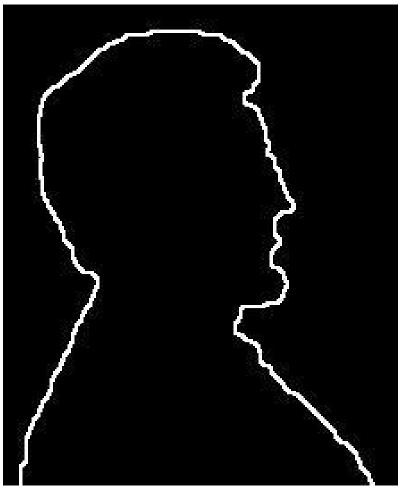


Figure 12.3 Object boundary extracted using 5×5 square structuring element

Thinning

Thinning means reducing binary objects or shapes in an image to strokes that are a single pixel wide. The thinning of a set *A* by a structuring element *B*, is defined as:

$$\begin{aligned} A \otimes B &= A - (A \circledast B) \\ &= A \cap (A \circledast B)^c \end{aligned}$$

Since we only match the pattern (shape) with the structuring elements, no background operation is required in the hit-or-miss transform.

Here, *B* is a sequence of structuring elements:

$$\{B\} = \{B^1, B^2, B^3, \dots, B^n\}$$

where B^i is the rotation of B^{i-1}. Thus, the thinning equation can be written as:

$$A \otimes \{B\} = ((\dots\left((A \otimes B^1) \otimes B^2\right)\dots) \otimes B^n)$$

The entire process is repeated until no further changes occur. The next figure shows an example of thinning the fingerprint ridges so that each is one pixel thick.

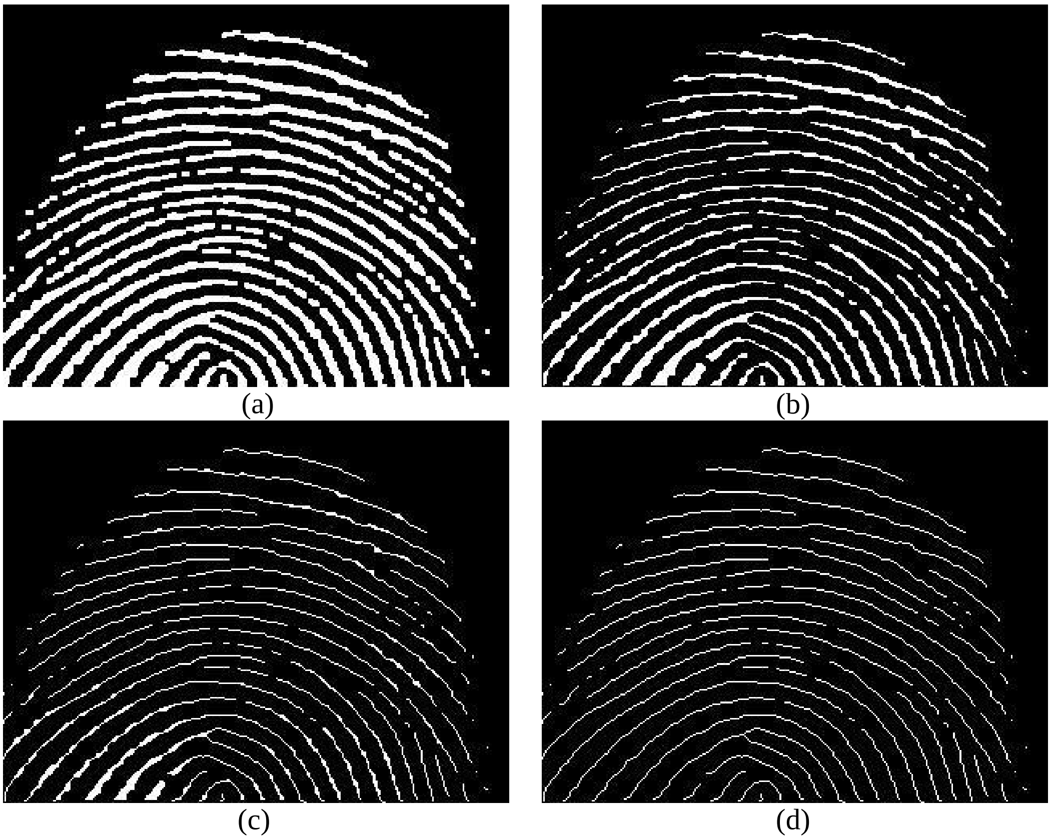


Figure 12.4 (a) Original fingerprint image. (b) Image thinned once. (c) Image thinned twice. (d) Image thinned until stability (no changes occur).

Skeletonization (Skeleton Extraction)

is another way to reduce binary objects to thin strokes that retain important structural information about the shapes of the original objects. The skeleton of *A* can be expressed in terms of erosions and openings as follows:

$$S(A) = \bigcup_{k=0}^{K} S_k(A)$$

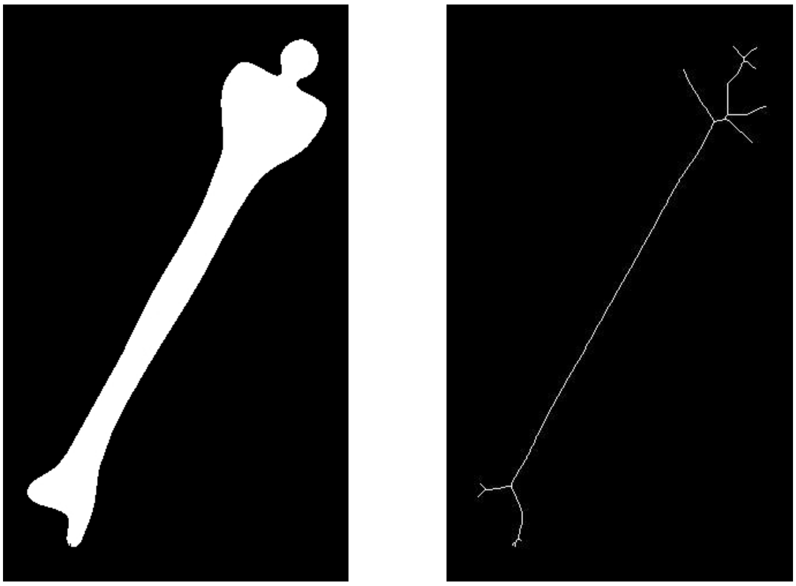
with

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

where *B* is a structuring element, and $(A \ominus kB)$ indicates *k* successive erosions of *A*:

$$(A \ominus kB) = (\dots(A \ominus B) \ominus B) \ominus \dots) \ominus B$$

The figure below illustrates an example of extracting a skeleton of an object in a binary image.



(a) (b)

Figure 12.5 (a) Bone image. (b) Skeleton extracted from (a).

Gray-scale Morphology

The basic morphological operations of dilation, erosion, opening and closing can also be applied to gray images.

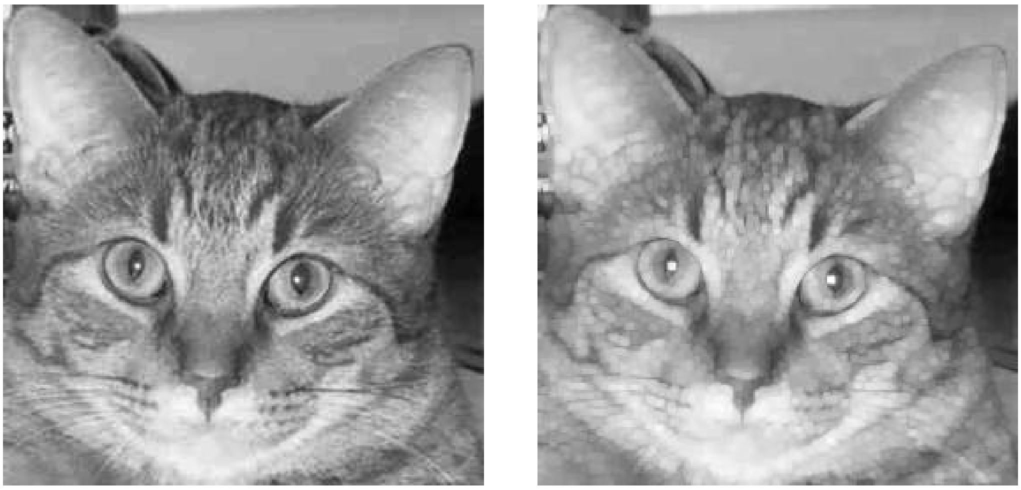
Gray-scale Dilation

The *gray-scale dilation* of a gray-scale image *f* by a structure element *b* is defined as:

$$(f \oplus b)(x,y) = \max\{f(x-x', y-y') + b(x',y') \mid (x',y') \in D_b\}$$

where D_b is the domain of the structuring element *b*. This process operates in the same way as the spatial convolution.

The figure below shows the result of dilating a gray image using a 3×3 square structuring element.



(a) (b)

Figure 12.6 (a) Original gray image. (b) Dilated image.

We can see that gray-scale dilation produces the following:

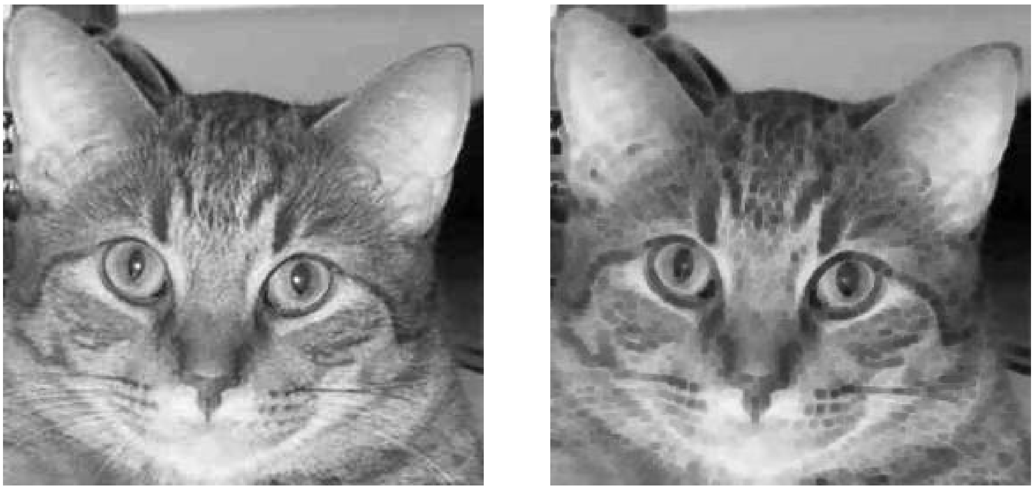
1. Bright and slightly blurred image.
2. Small, dark details have been reduced.

Gray-scale Erosion

The *gray-scale erosion* of a gray-scale image *f* by a structure element *b* is defined as:

$$(f \ominus b)(x,y) = \min\{f(x+x', y+y') - b(x',y') \mid (x',y') \in D_b\}$$

The next figure shows the result of eroding a gray image using a 3×3 square structuring element.



(a) (b)

Figure 12.7 (a) Original gray image. (b) Eroded image.

We can see that gray-scale erosion produces the following:

1. Dark image.
2. Small, bright details were reduced.

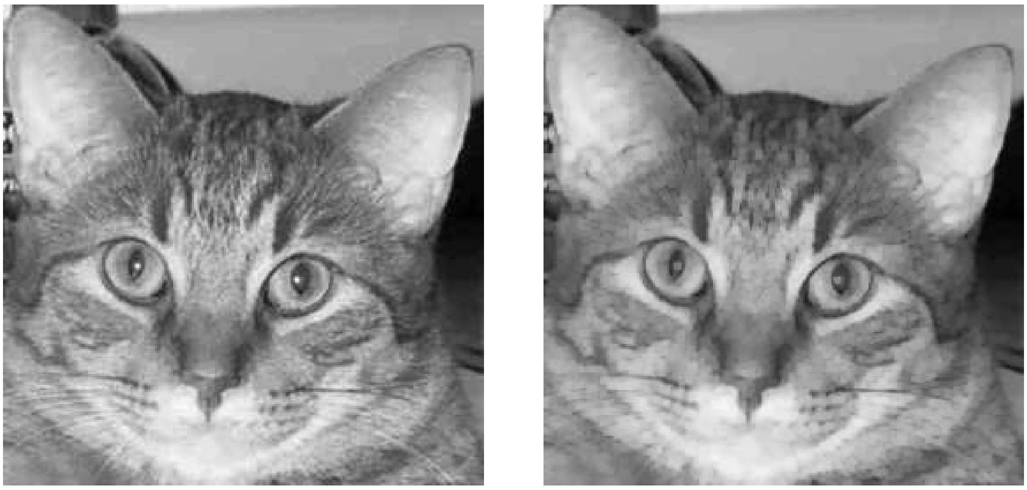
Gray-scale Opening and Closing

The opening and closing of gray-scale images have the same form as in the binary images:

$$\text{Opening :} \quad f \circ b = (f \ominus b) \oplus b$$

$$\text{Closing:} \quad f \bullet b = (f \oplus b) \ominus b$$

The figure below shows the result of opening a gray image.

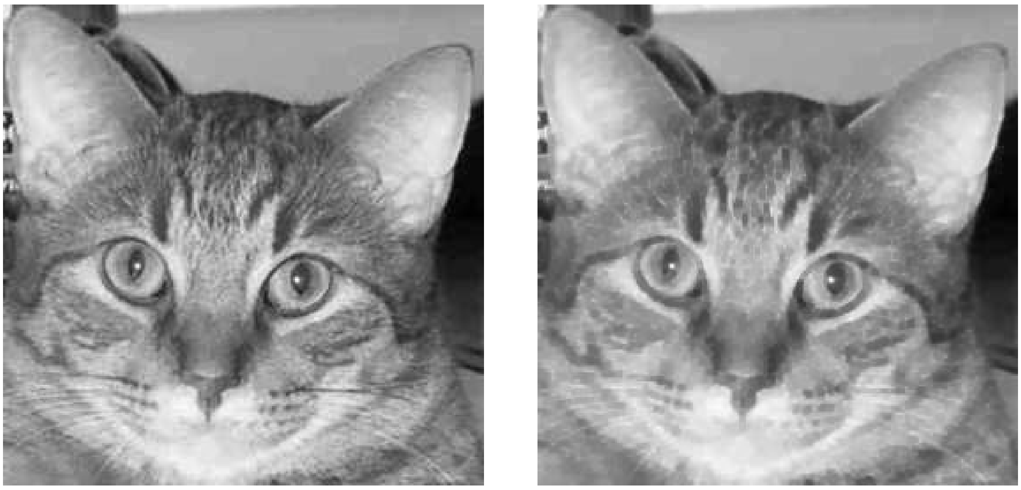


(a) (b)

Figure 12.8 (a) Original gray image. (b) Opened image.

Note the decreased sizes of the small, bright details, with no appreciable effect on the darker gray levels.

The figure below shows the result of closing a gray image.



(a) (b)

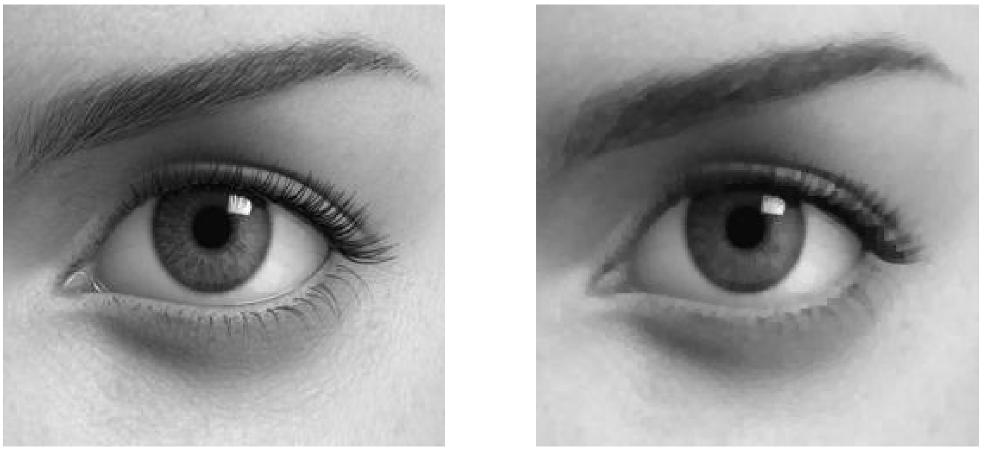
Figure 12.9 (a) Original gray image. (b) Closed image.

Note the decreased sizes of the small, dark details, with relatively little effect on the bright features.

Gray-Scale Morphology Applications

Morphological smoothing

Smoothing is obtained by performing a morphological opening followed by a closing as shown in the figure below.



(a) (b)

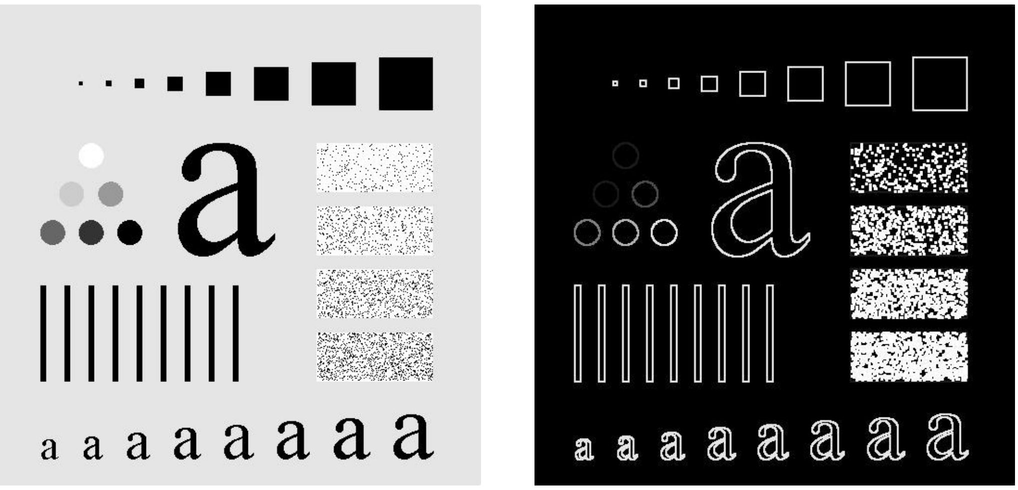
Figure 12.10 (a) Original gray image. (b) Morphological smoothed image.

Morphological gradient

is produced from subtracting an eroded image from its dilated version. It is defined as:

$$g = (f \oplus b) - (f \ominus b)$$

The resulted image has edge-enhancement characteristics, thus morphological gradient can be used for edge detection as shown in the figure below.



(a) (b)

Figure 12.11 (a) Original gray image. (b) Morphological gradient.