DATA ANALYSIS AND VISUALIZATION

INSTRUCTOR: UMME AMMARAH

IMAGE ENHANCEMENT

SPATIAL DOMAIN METHODS

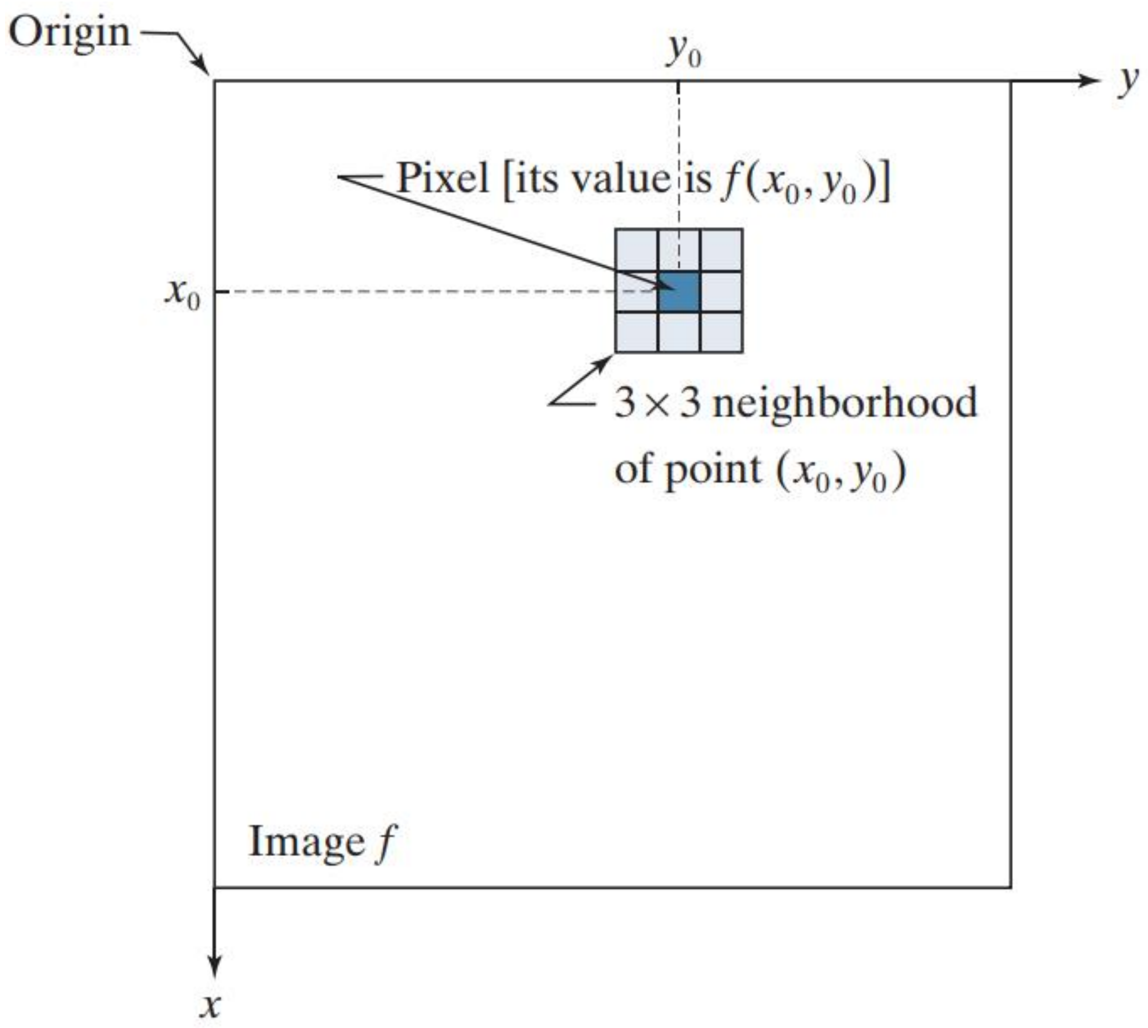
IMAGE ENHANCEMENT

- To process an image so that the result is more suitable than the original image for a specific application.

- Categories:
  - Spatial domain methods
  - Frequency domain methods

SPATIAL DOMAIN METHOD

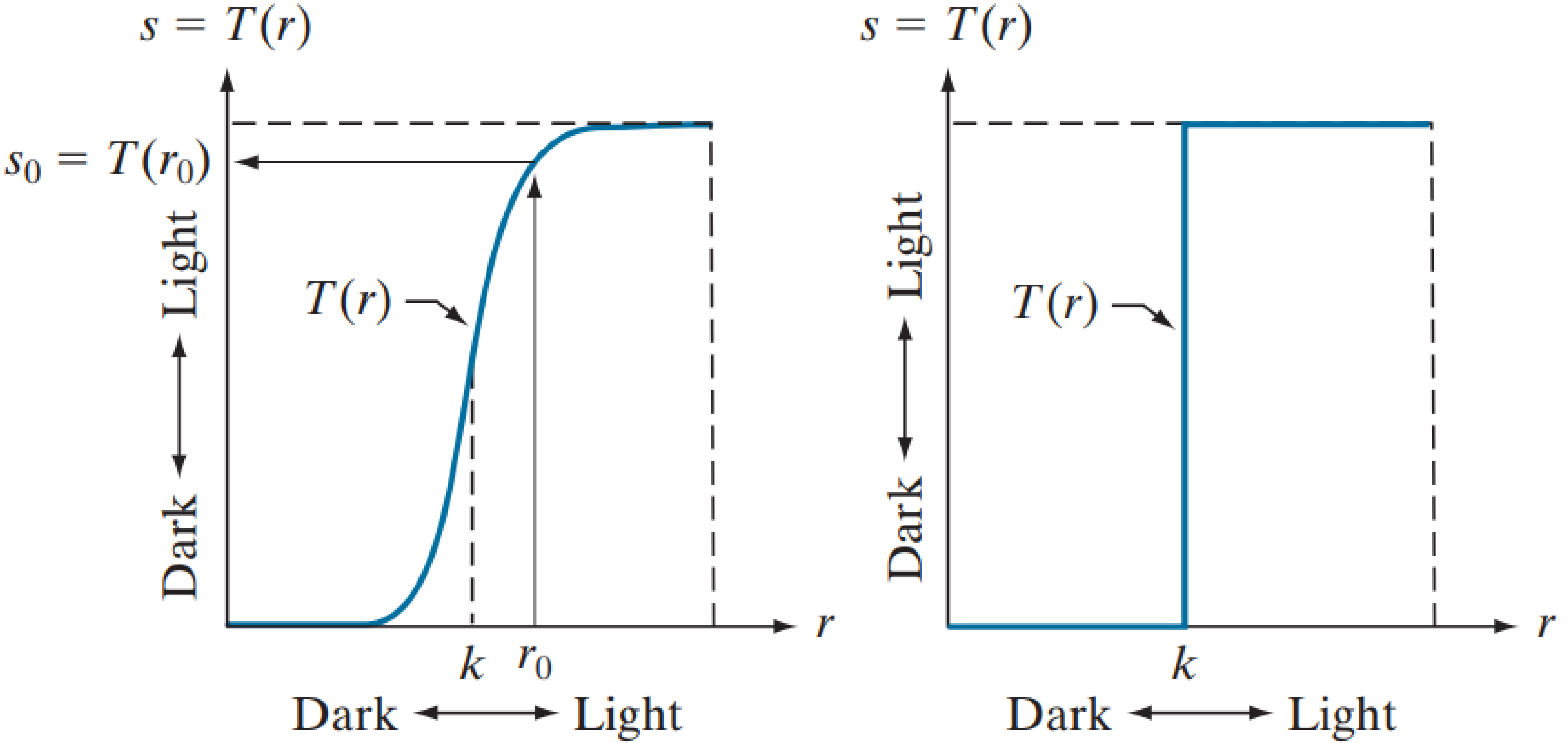
A 3×3 neighborhood about a point (x_0, y_0) in an image. The neighborhood is moved from pixel to pixel in the image to generate an output image.



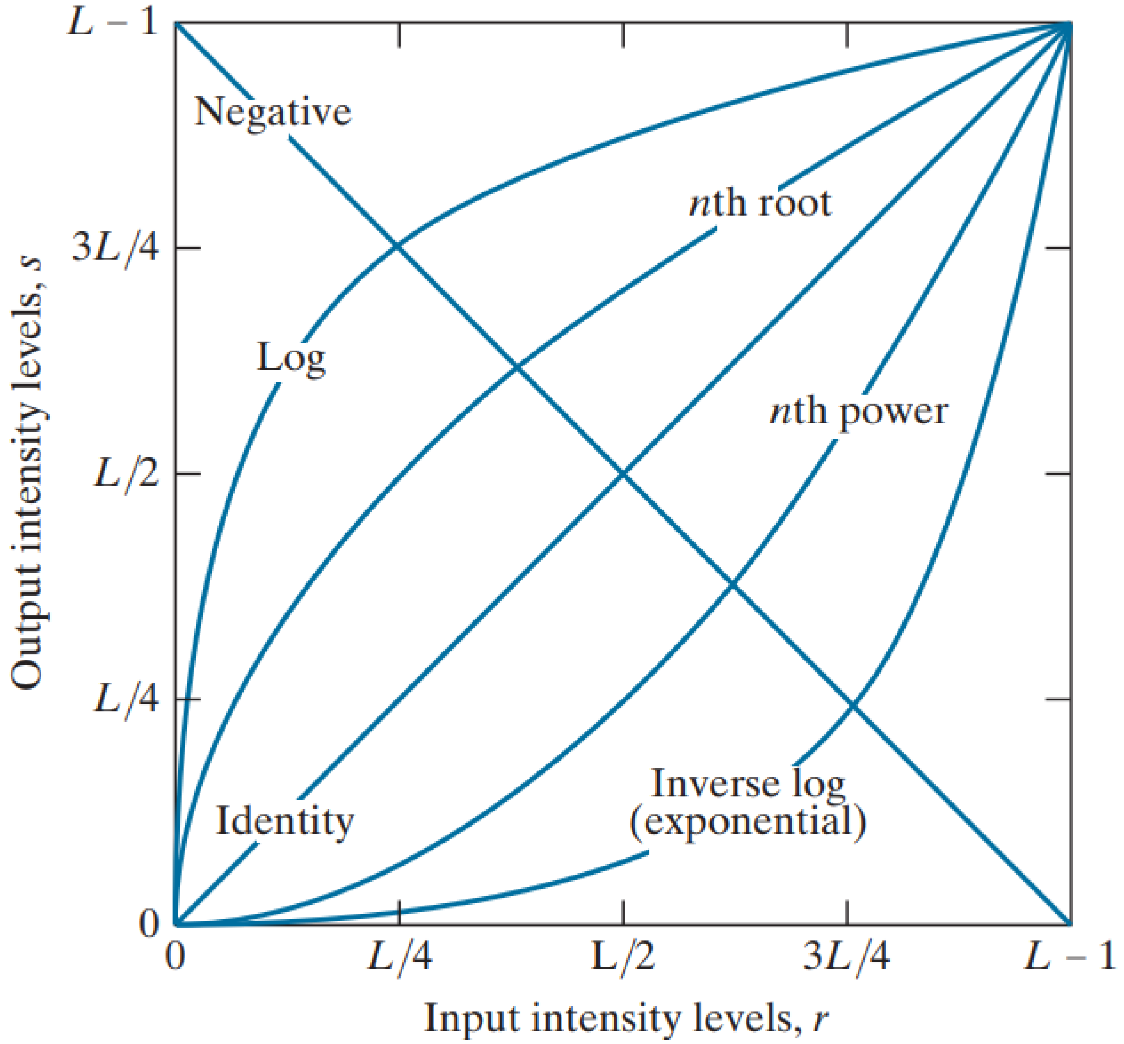
SPATIAL DOMAIN METHODS

- When the neighborhood is 1×1 then g depends only on the value of f at (x, y) and T becomes a gray-level transformation (or mapping) function.
- Examples: Point processing techniques (e.g. contrast stretching, thresholding)

CONTRAST STRETCHING AND THRESHOLDING



INTENSITY TRANSFORMATION

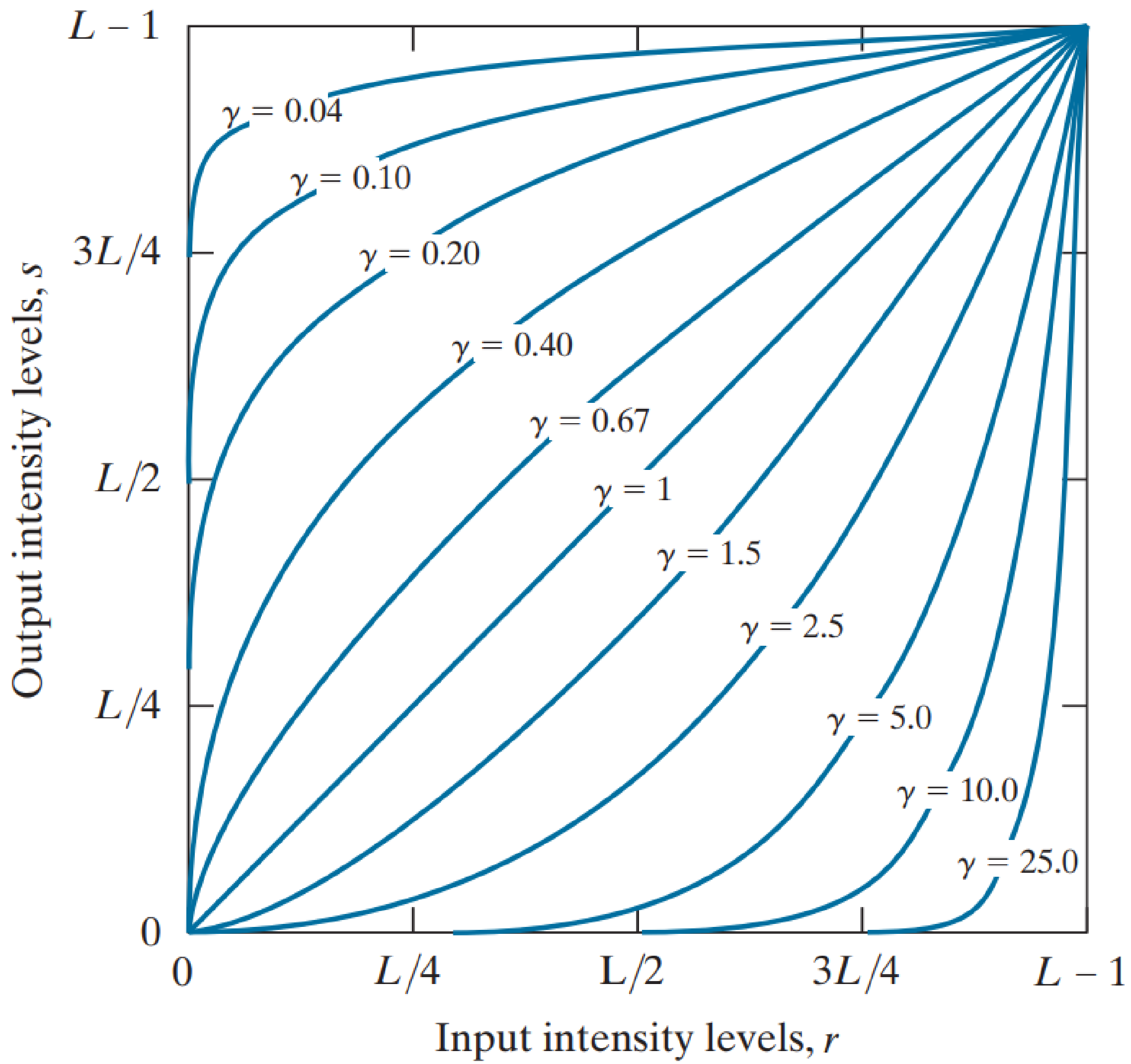


LOG TRANSFORMATIONS

- Log function: $s = c\log(1+r)$
  - Stretch low intensity levels Compress high intensity levels
- Inverse log function: $s = c\log^{-1}(r)$
  - Stretch high intensity levels Compress low intensity levels

POWER LAW (GAMMA) TRANSFORMATION

- $s = cr^{\gamma}$
- Gamma correction/gamma encoding

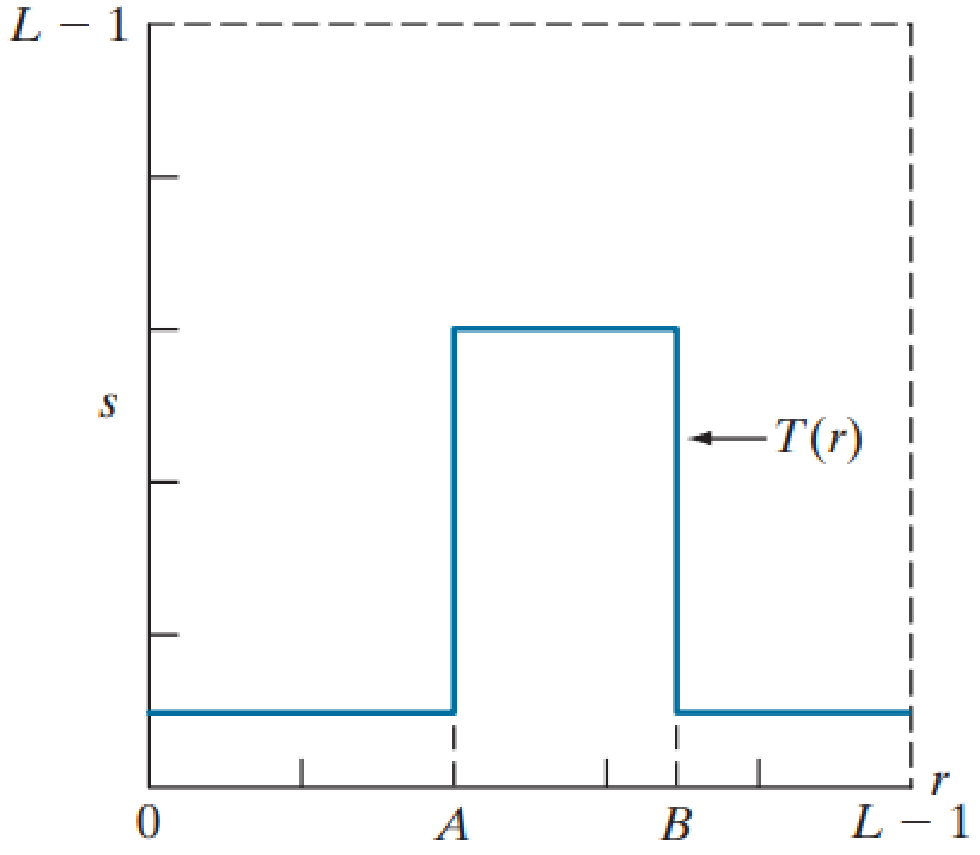


Results of applying the transformation
with g = 3 0. , 4.0, and 5.0, respectively.
(c = 1 in all cases.)

Washed-out appearance was reduced by a large gamma value

INTENSITY LEVEL SLICING

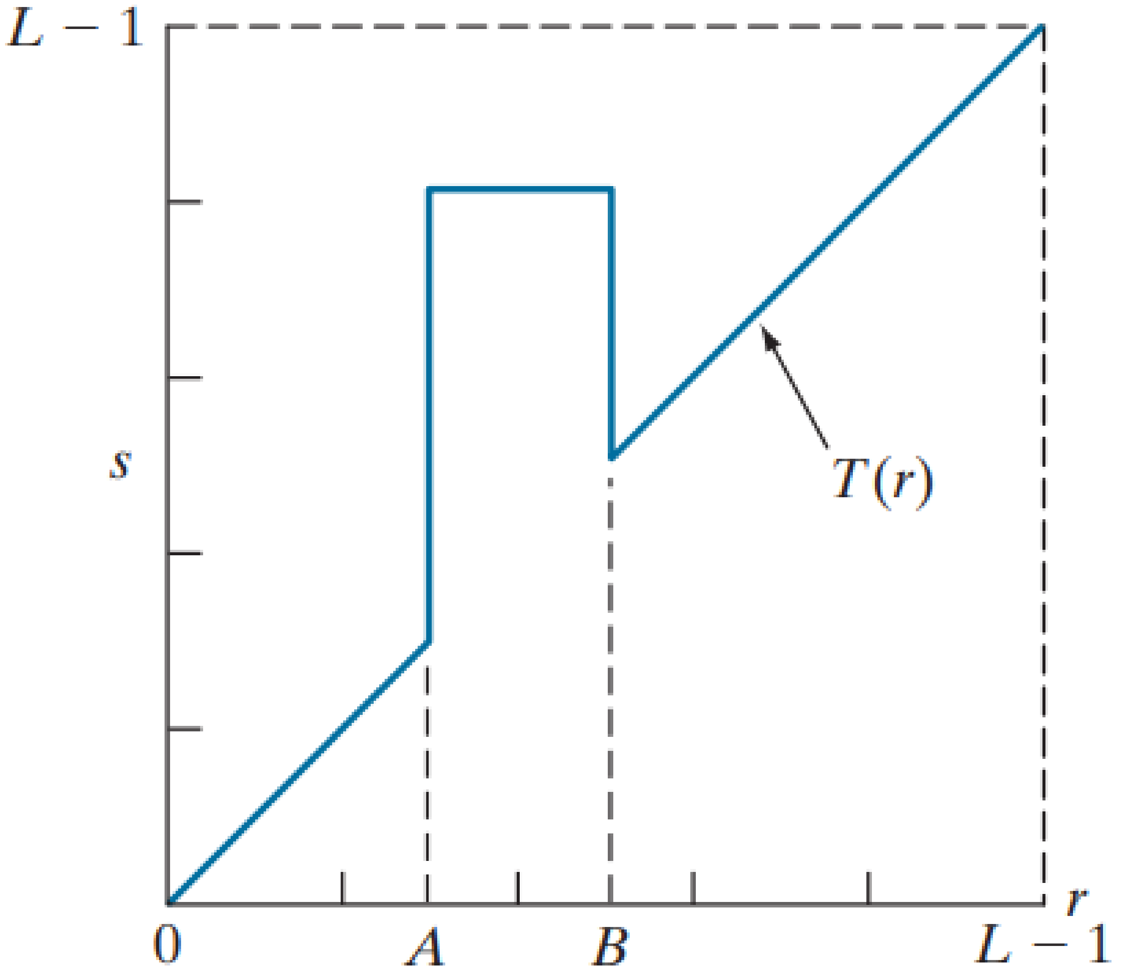
- To highlight a specific range of gray levels in an image (e.g. to enhance certain features).



- One way is to display a high value for all gray levels in the range of interest and a low value for all other gray levels (binary image).

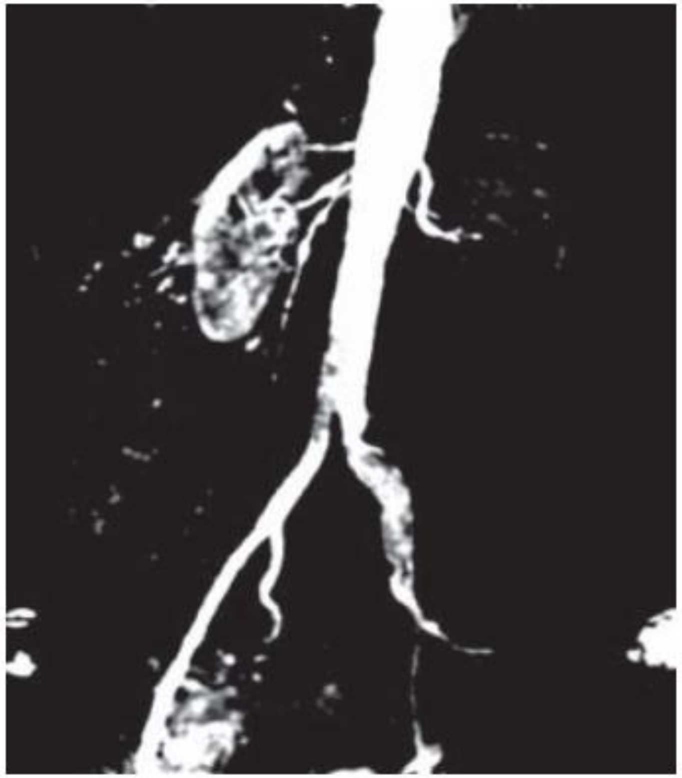
INTENSITY LEVEL SLICING

- The second approach is to brighten the desired range of gray levels but preserve the background and gray-level tonalities in the image.

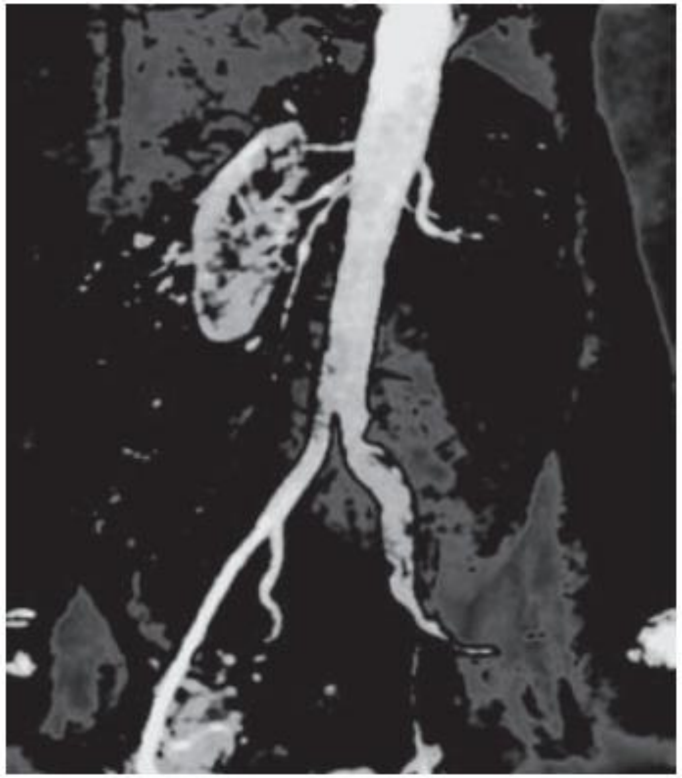


EXAMPLE

Original Image



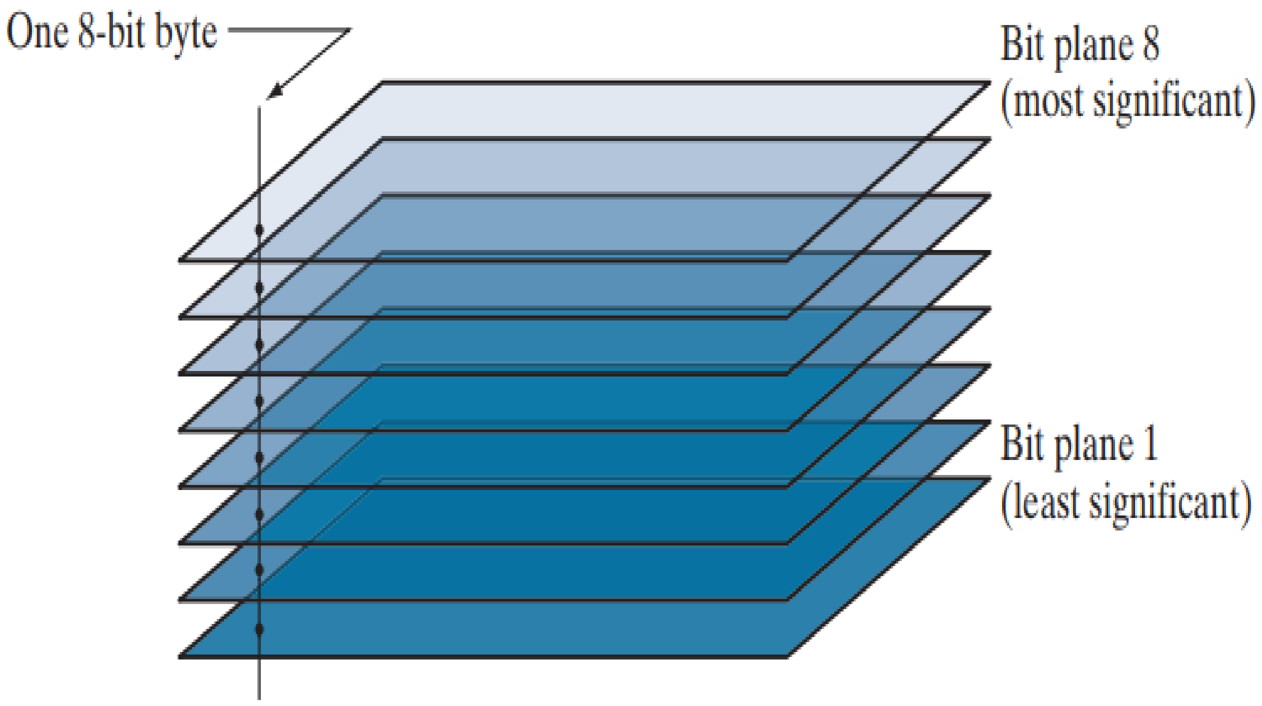
Result of using slicing method 1



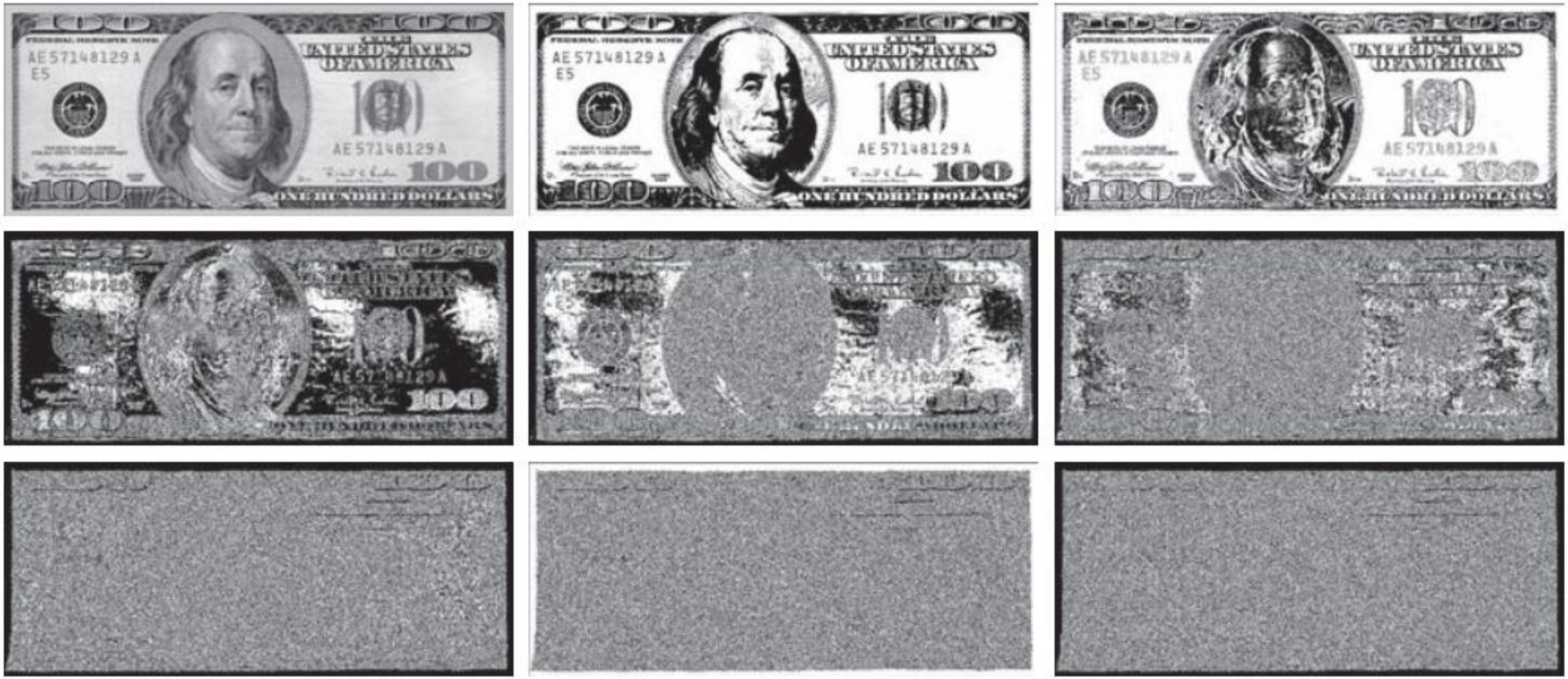
Result of using slicing method 2

BIT PLANE SLICING

- To highlight the contribution made to the total image appearance by specific bits.
  - Plane 1 contains the least significant bit and plane 8 contains the most significant bit.
  - Only the higher order bits (top four) contain visually significant data. The other bit planes contribute the more subtle details.
  - Plane 7 corresponds exactly with an image thresholded at gray level 128.



Each pixel is represented by 8 bits, the image is composed of 8 1-bit planes



a b c
d e f
g h i

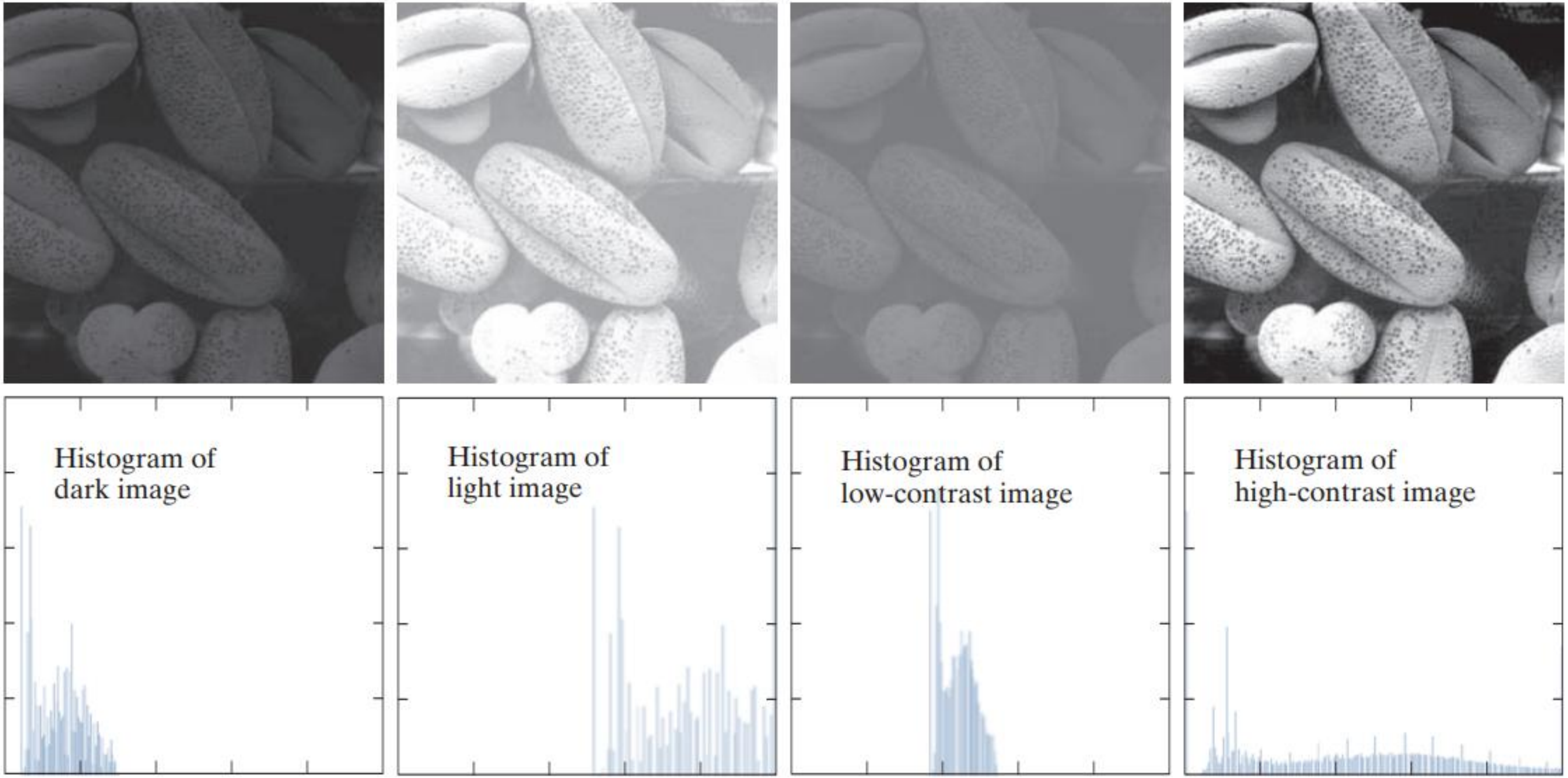
(a) An 8-bit gray-scale image of size 550 x1192 pixels. (b) through (i) Bit planes 8 through 1, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image..

IMAGE RECONSTRUCTED FROM BIT PLANES

(a) 8 and 7 (b) 8, 7, and 6 (c) 8, 7, 6, and 5

We conclude that, in this example, storing the four highest-order bit planes would allow us to reconstruct the original image in acceptable detail. Storing these four planes instead of the original image requires 50% less storage.

HISTOGRAM PROCESSING


Histogram of
dark image
Histogram of
light image
Histogram of
low-contrast image
Histogram of
high-contrast image

METHODS OF HISTOGRAM PROCESSING

- Two well-known methods of image enhancement:
  - Histogram equalization
  - Histogram matching

IMAGE HISTOGRAM

- Mathematically represented as:

$$h(r_k) = n_k$$

  - r_k represents the intensities of an L-level digital image. k = 0, 1, 2, … L-1 and n_k is the number of pixels in f with intensity r_k

$$p(r_k) = \frac{h(r_k)}{MN} = \frac{n_k}{MN}$$

- Normalized Histogram:
  - M and N are the number of image rows and columns, sum of p(r_k) for all values of k is always 1.
  - It estimates of the probabilities of intensity levels occurring in an image

IMAGE HISTOGRAM

- An image histogram is a plot of the gray level frequencies.

0	0	1	0	2	0
1	0	7	7	7	0
0	7	0	0	7	0
1	0	0	7	2	0
0	0	7	1	0	1
1	0	7	7	7	0

frequencies

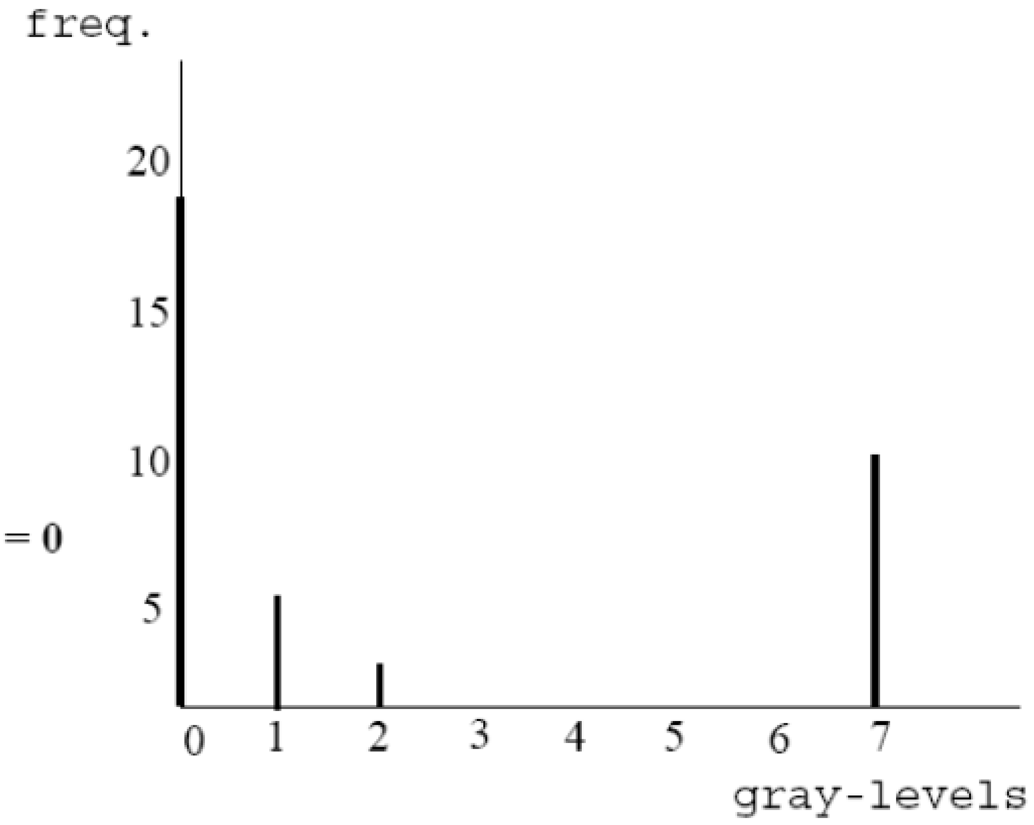
f(0)= 18

f(1)= 6

f(2)= 2

f(3)=f(4)=f(5)=f(6) = 0

f(7) = 10



NORMALIZED HISTOGRAM

- Divide frequencies by total number of pixels to represent as probabilities.

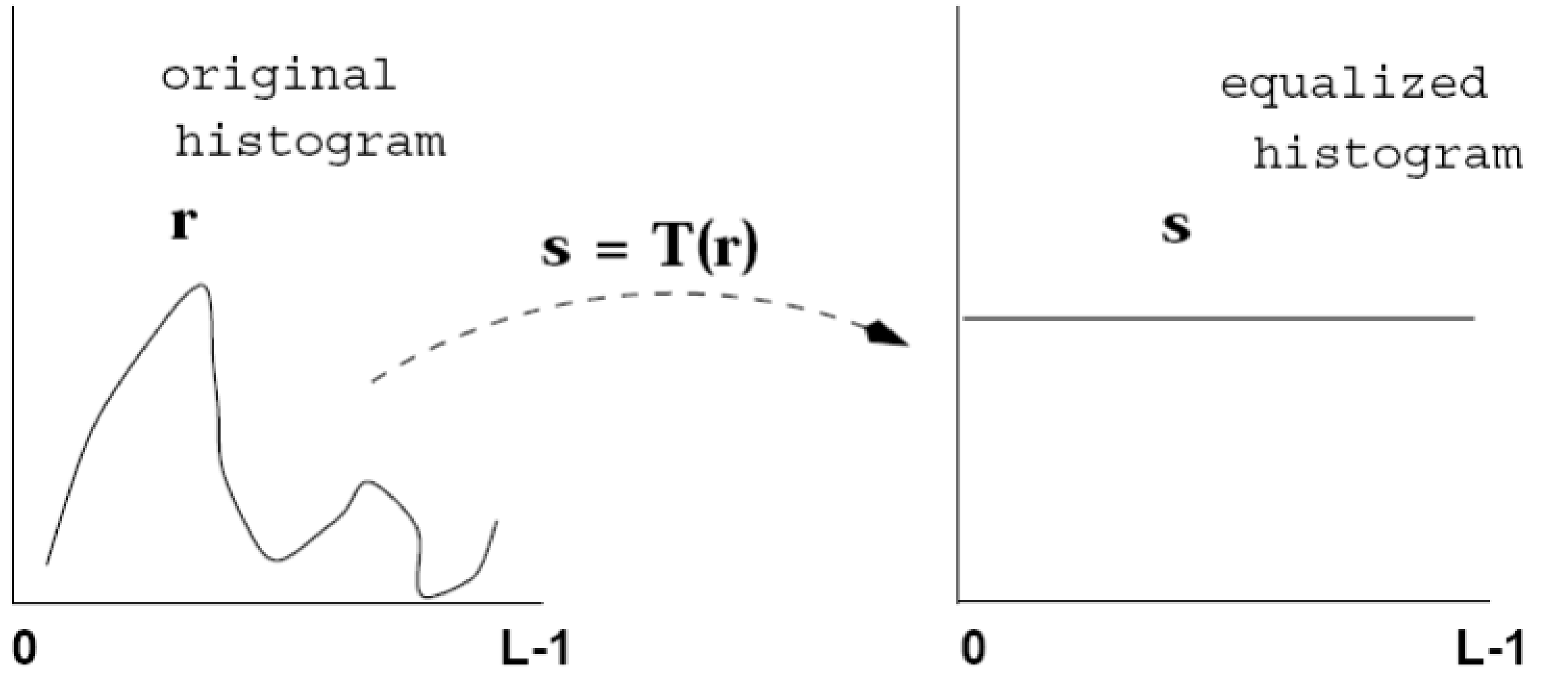
$$P(0) = \frac{f(0)}{36} = \frac{1}{2} \qquad P(1) = \frac{f(1)}{36} = \frac{1}{6}$$

$$P(2) = \frac{f(2)}{36} = \frac{1}{18} \qquad P(3) = P(4) = P(5) = P(6) = 0$$

$$P(7) = \frac{f(7)}{36} = \frac{5}{18}$$

HISTOGRAM EQUALIZATION

- The main idea is to redistribute the gray-level values uniformly.



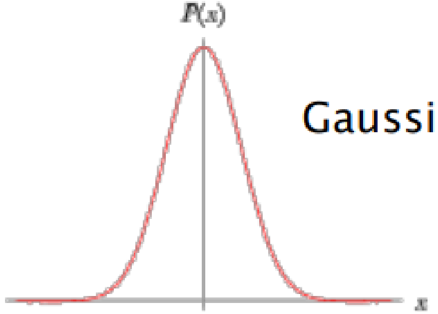
PROBABILITY DENSITY/ PROBABILITY DISTRIBUTION

- The probability density function (pdf) is a real-valued function f_x(x) describing the density of probability at each point in the sample space.

- The integral of f_x(x) defines the cumulative probability distribution function(cdf) F_x(x)

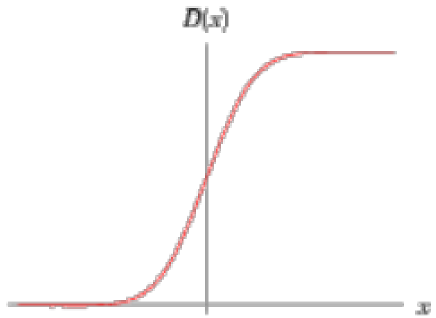
$$F_X(x) = P(X \le x) = \int_{-\infty}^{x} f_X(a)da$$

$f_X(x)$



Gaussian

$F_X(x)$



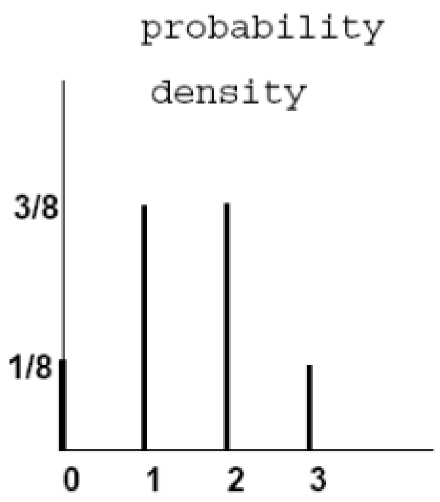
non-decreasing

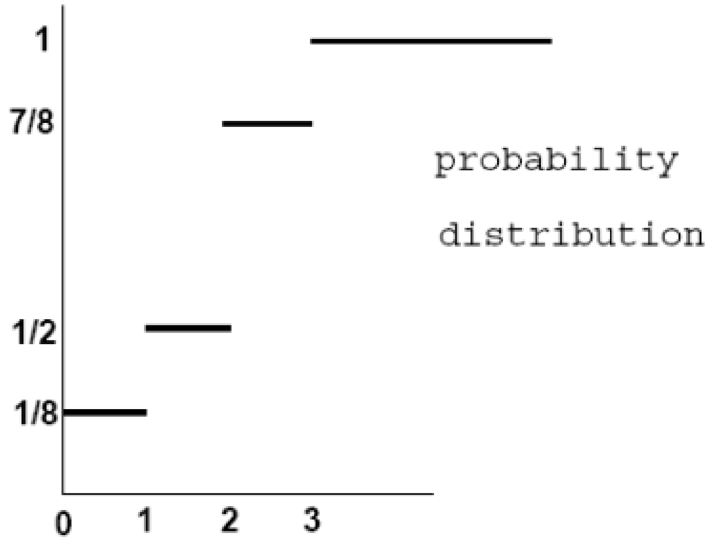
EXAMPLE

In the discrete case:

$$F_X(x) = P(X \le x) = \sum_{k=0}^{x} P(X = k)$$

$$F_X(0) = P(X \le 0) = P(X = 0) = 1/8$$
$$F_X(1) = P(X \le 1) = P(X = 0) + P(X = 1) = 1/2$$
$$F_X(2) = P(X \le 2) = P(X = 0) + P(X = 1) + P(X = 2) = 7/8$$
$$F_X(3) = P(X \le 3) = P(X = 0) + P(X = 1) + P(X = 2) + P(X = 3) = 1$$





non-decreasing

EXAMPLE

1	2	1	1	1
2	5	3	5	2
2	5	5	5	2
2	5	3	5	2
1	1	1	2	1

Max Value = 5 (requires 3 bits)
Number of grey levels (L) = 8
N = 25

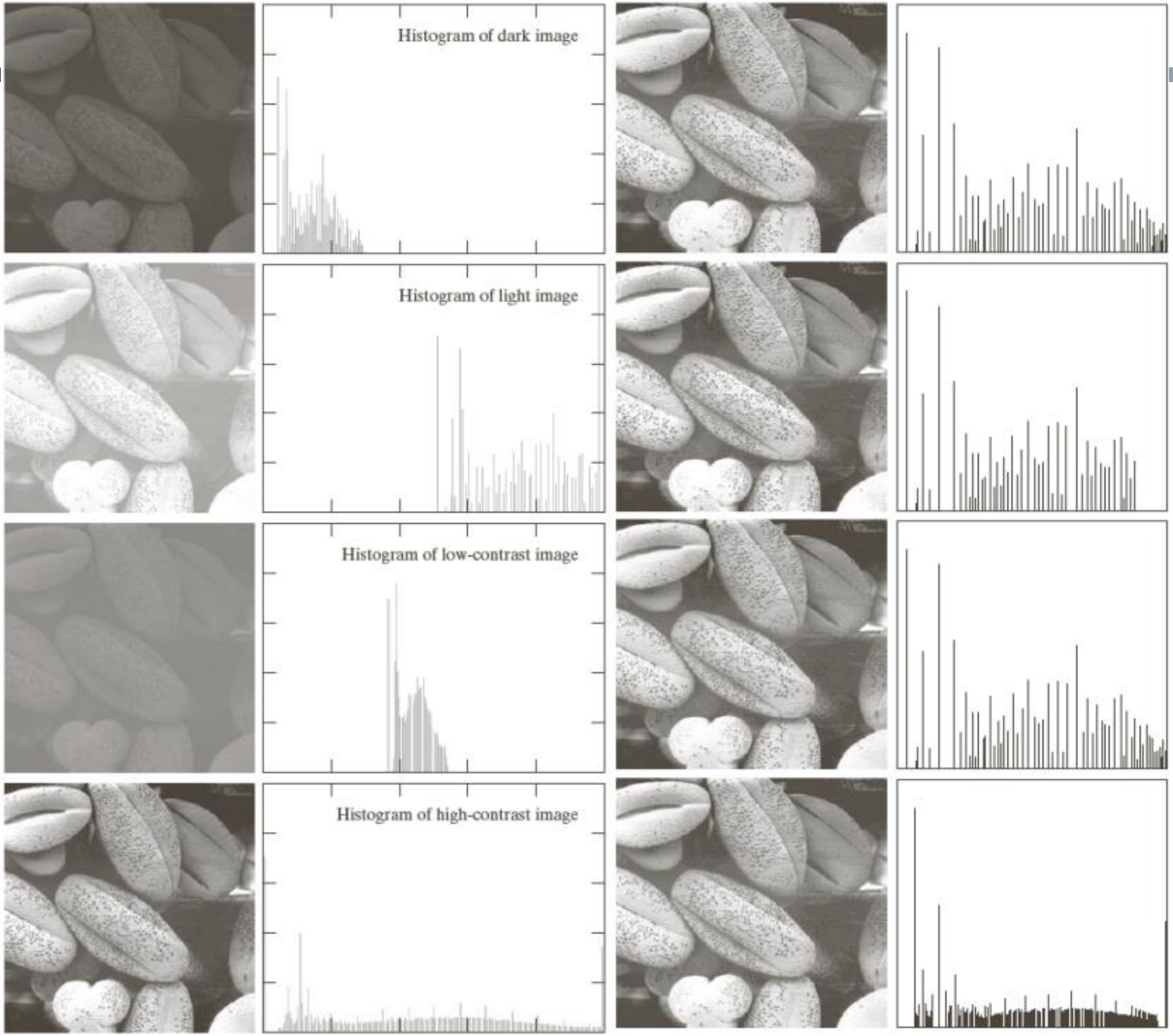
Gray level r_k	n_k	P(r_k)= n_k/N (pdf)	s_k (CDF)	Sk x(L-1)	Hist. equalized level
0	0	0	0	0	0
1	8	0.32	0.32	2.24	2
2	8	0.32	0.64	4.48	4
3	2	0.08	0.72	5.04	5
4	0 4	0	0.72	5.04	5
5	7	0.28	1	7	7
6	0	0	1	7	7
7	0	0	1	7	7

EXAMPLE CONT.

Grey Level	0	2	4	5	7
Frequency	0	8	8	6 2	7

2	4	2	2	2
4	7	5	7	4
4	7	7	7	4
4	7	5	7	4
2	2	2	4	2

Modified Image according to equalized histogram


Histogram of dark image
Histogram of light image
Histogram of low-contrast image
Histogram of high-contrast image

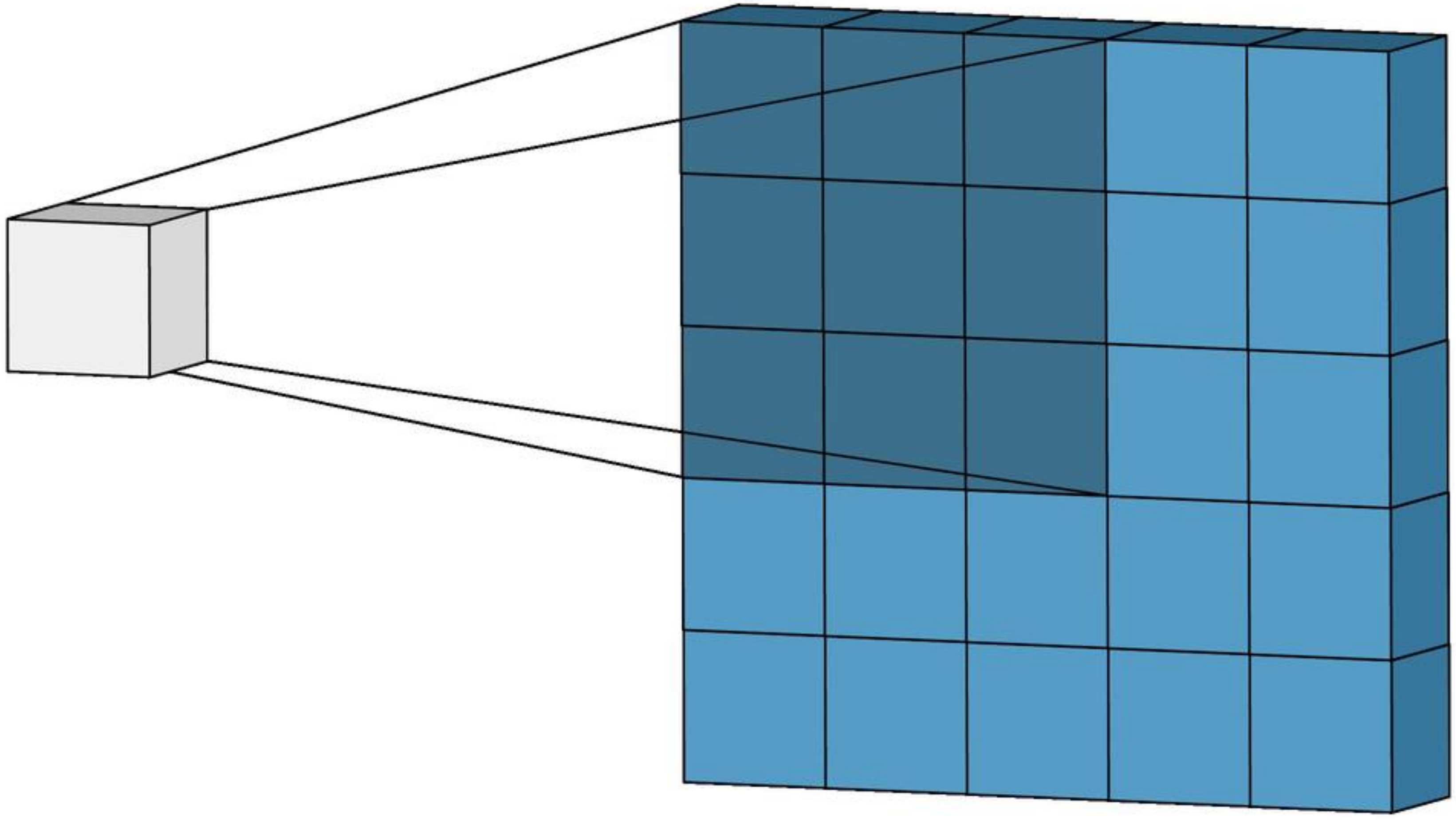
SPATIAL FILTERING

FILTER KERNEL

- The kernel is an array whose size defines the neighborhood of operation
- Whose coefficients determine the nature of the filter.
- Other terms used to refer to a spatial filter kernel are mask, template, and window.

LINEAR SPATIAL FILTERING

- The process consists simply of moving the filter mask from point to point in an image.
- At each point (x,y) the response of the filter at that point is calculated using a predefined relationship.
- A linear spatial filter performs a sum-of-products operation between an image f and a filter kernel, w.

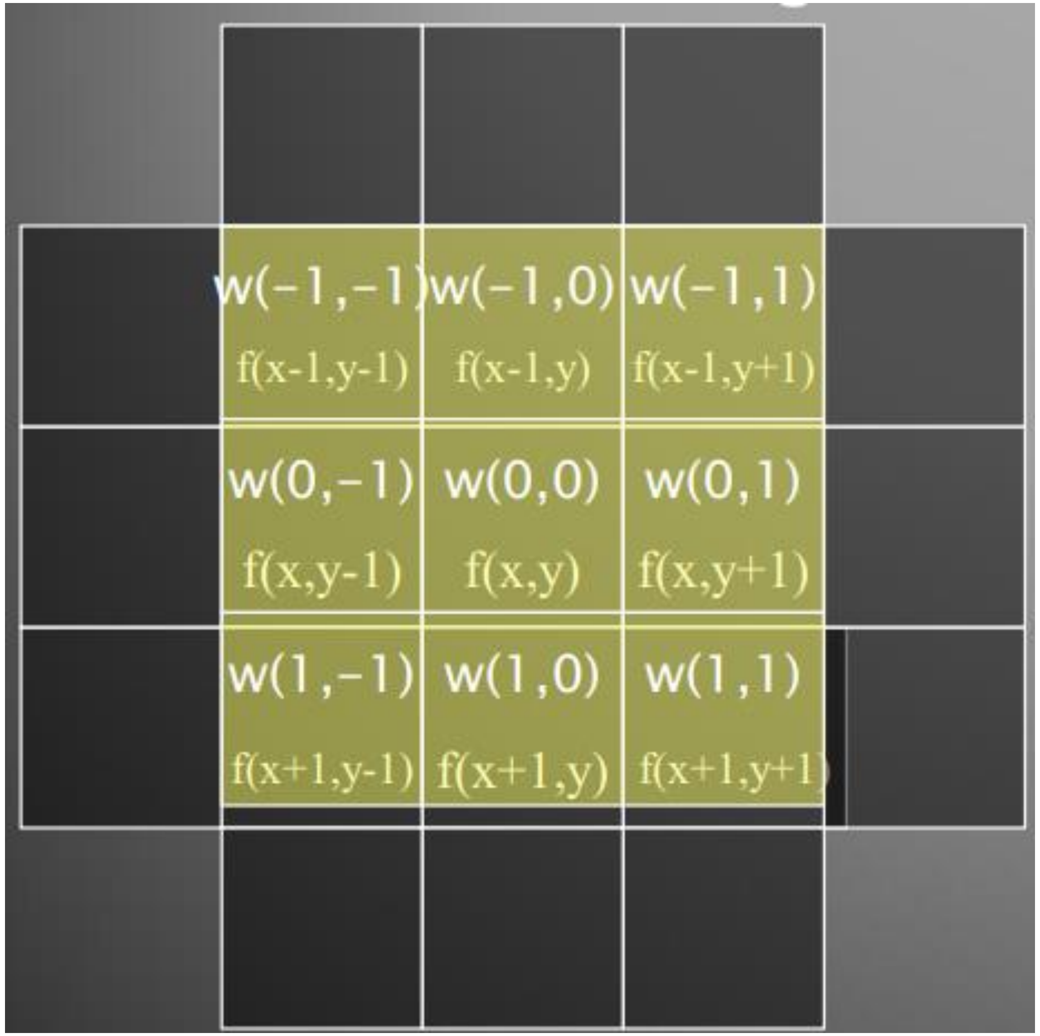


TRANSFORMED VALUE OF A PIXEL

- The result is the sum of products of the mask coefficients with the corresponding pixels directly under the mask.

$$g(x,y) = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \dots \\ + w(0,0)f(x,y) + \dots + w(1,1)f(x+1,y+1)$$

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$



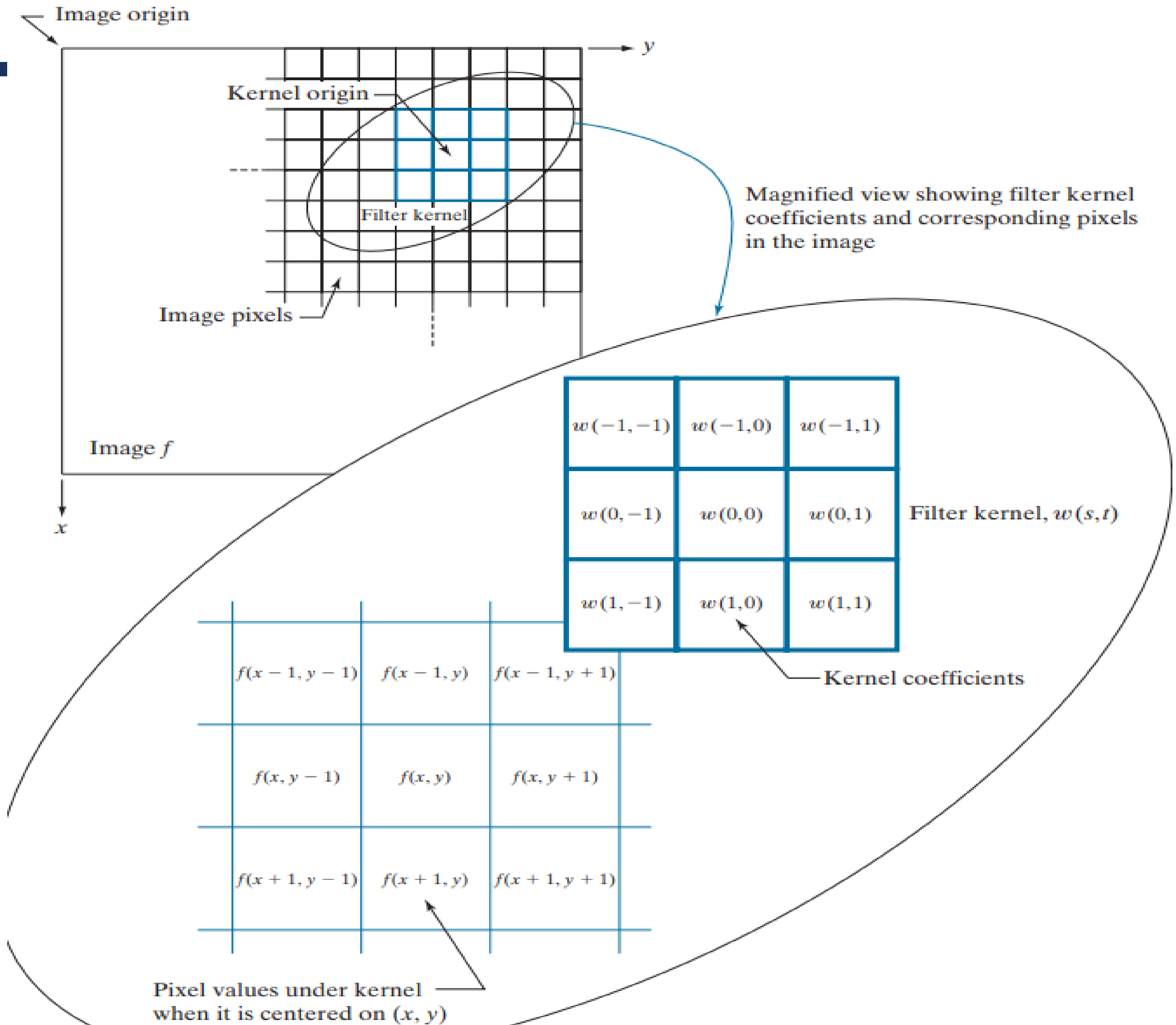


Image origin
y
Kernel origin
Filter kernel
Magnified view showing filter kernel coefficients and corresponding pixels in the image
Image pixels
Image f
x
$w(-1,-1)$
$w(-1,0)$
$w(-1,1)$
$w(0,-1)$
$w(0,0)$
$w(0,1)$
$w(1,-1)$
$w(1,0)$
$w(1,1)$
Filter kernel, $w(s,t)$
Kernel coefficients
$f(x-1, y-1)$
$f(x-1, y)$
$f(x-1, y+1)$
$f(x, y-1)$
$f(x, y)$
$f(x, y+1)$
$f(x+1, y-1)$
$f(x+1, y)$
$f(x+1, y+1)$
Pixel values under kernel when it is centered on (x, y)

Spatial Filtering (Neighborhood Processing)

Linear Spatial Filtering

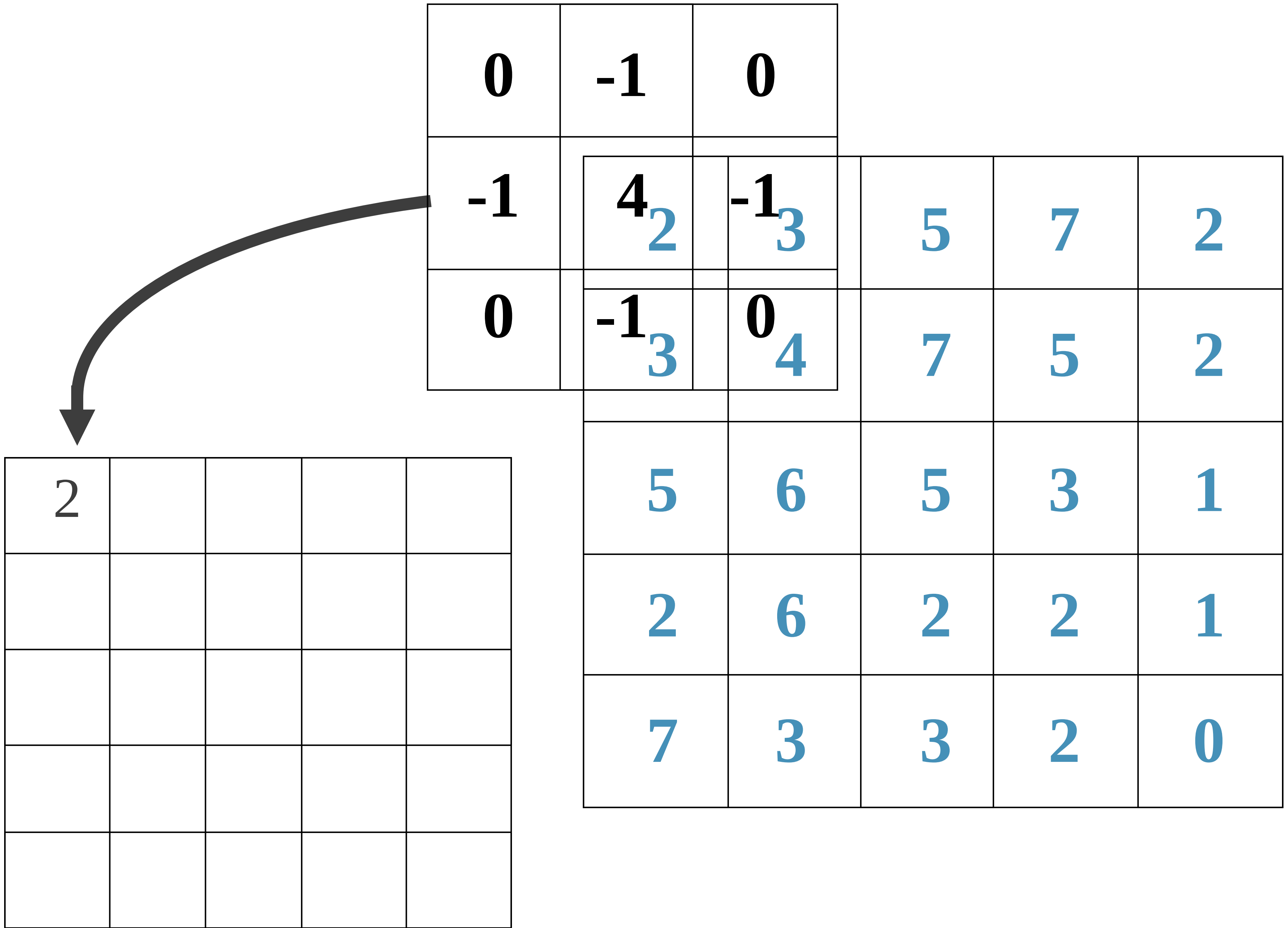
0	**-1**	**0**
-1	**4**	**-1**
0	**-1**	**0**

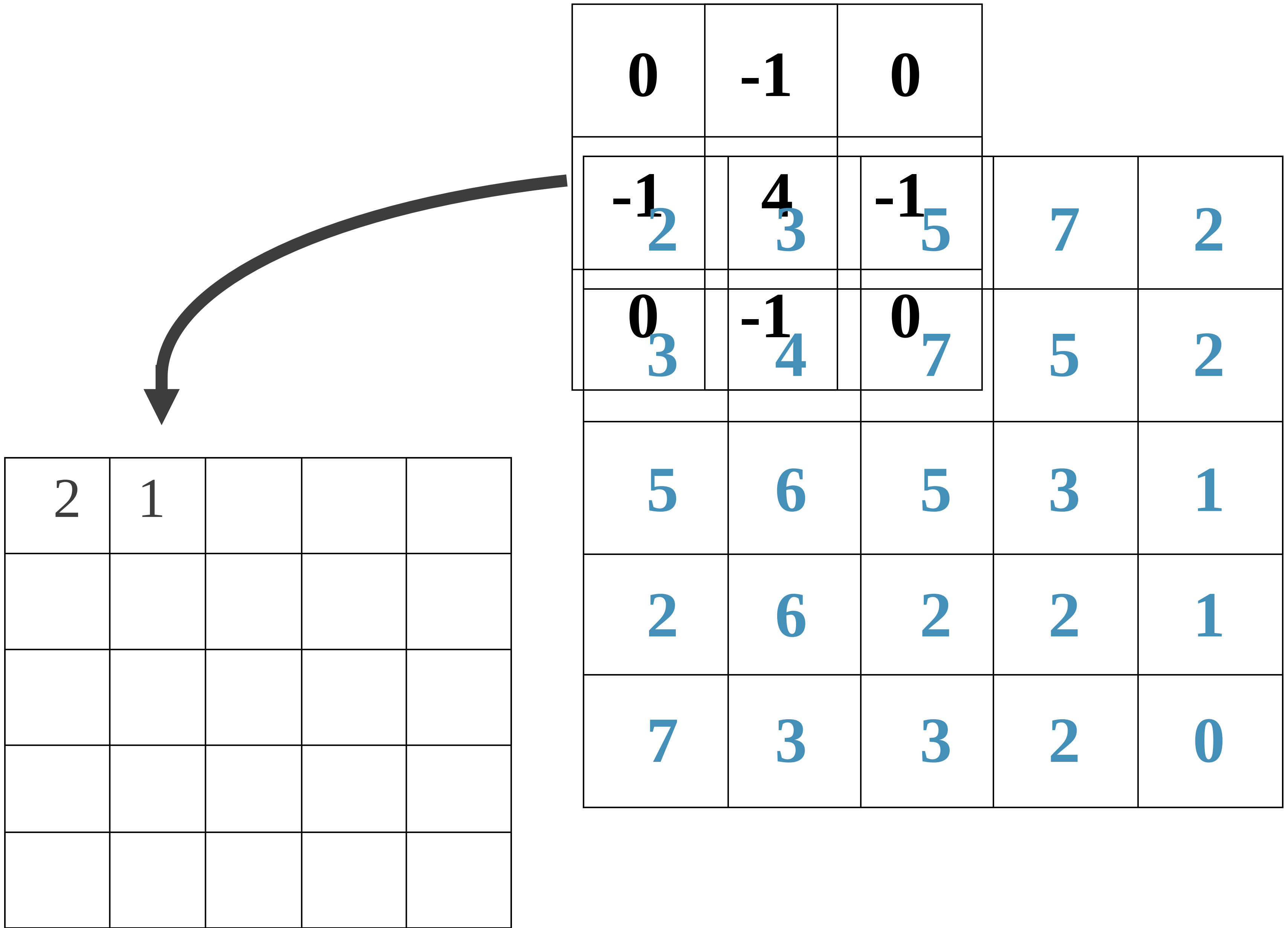
2	**3**	**5**	**7**	**2**
3	**4**	**7**	**5**	**2**
5	**6**	**5**	**3**	**1**
2	**6**	**2**	**2**	**1**
7	**3**	**3**	**2**	**0**

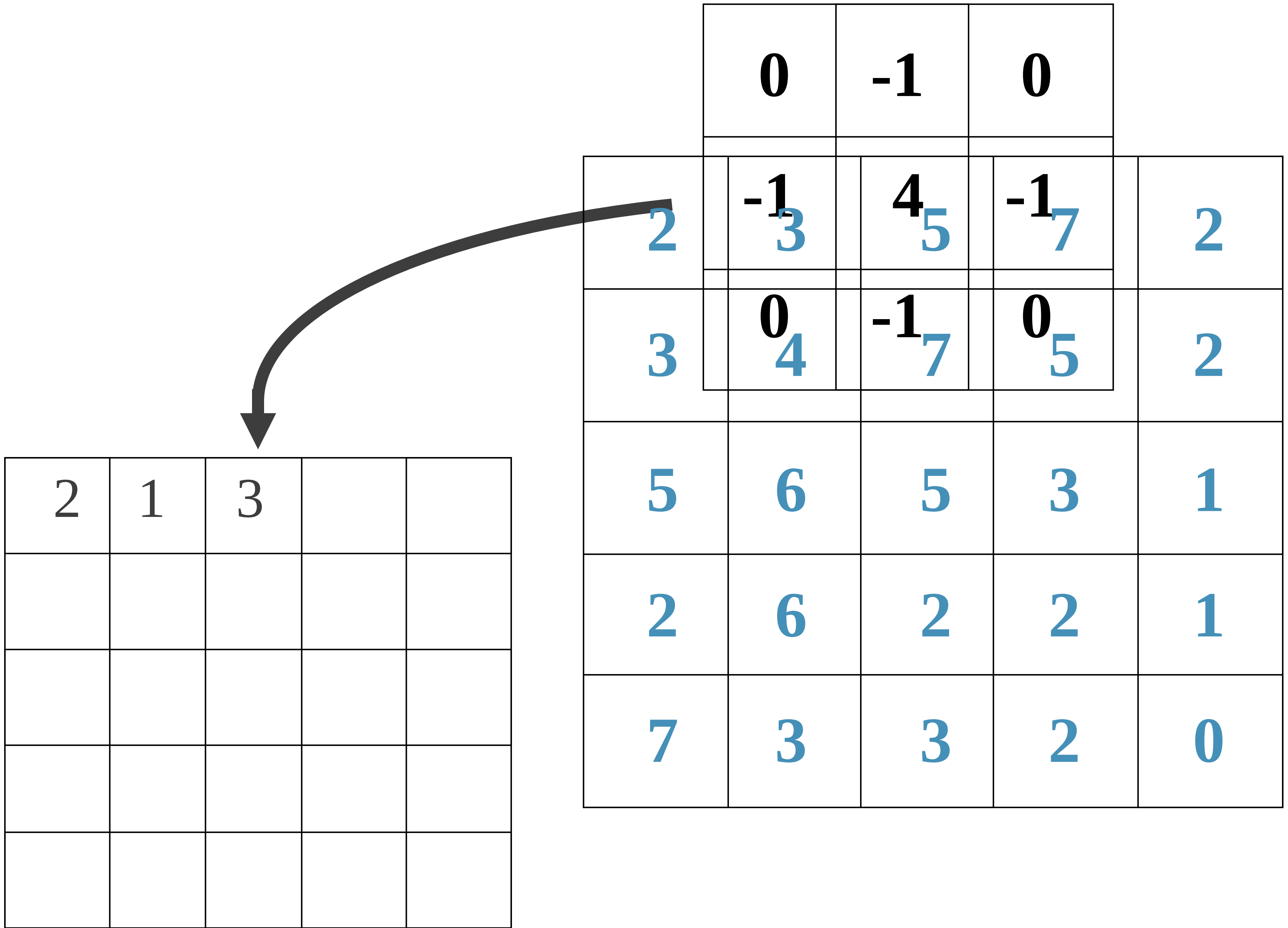
w: positioned so that its center coefficient is coincident with the origin of f

0	-1	0
-1	4	-1
0	-1	0

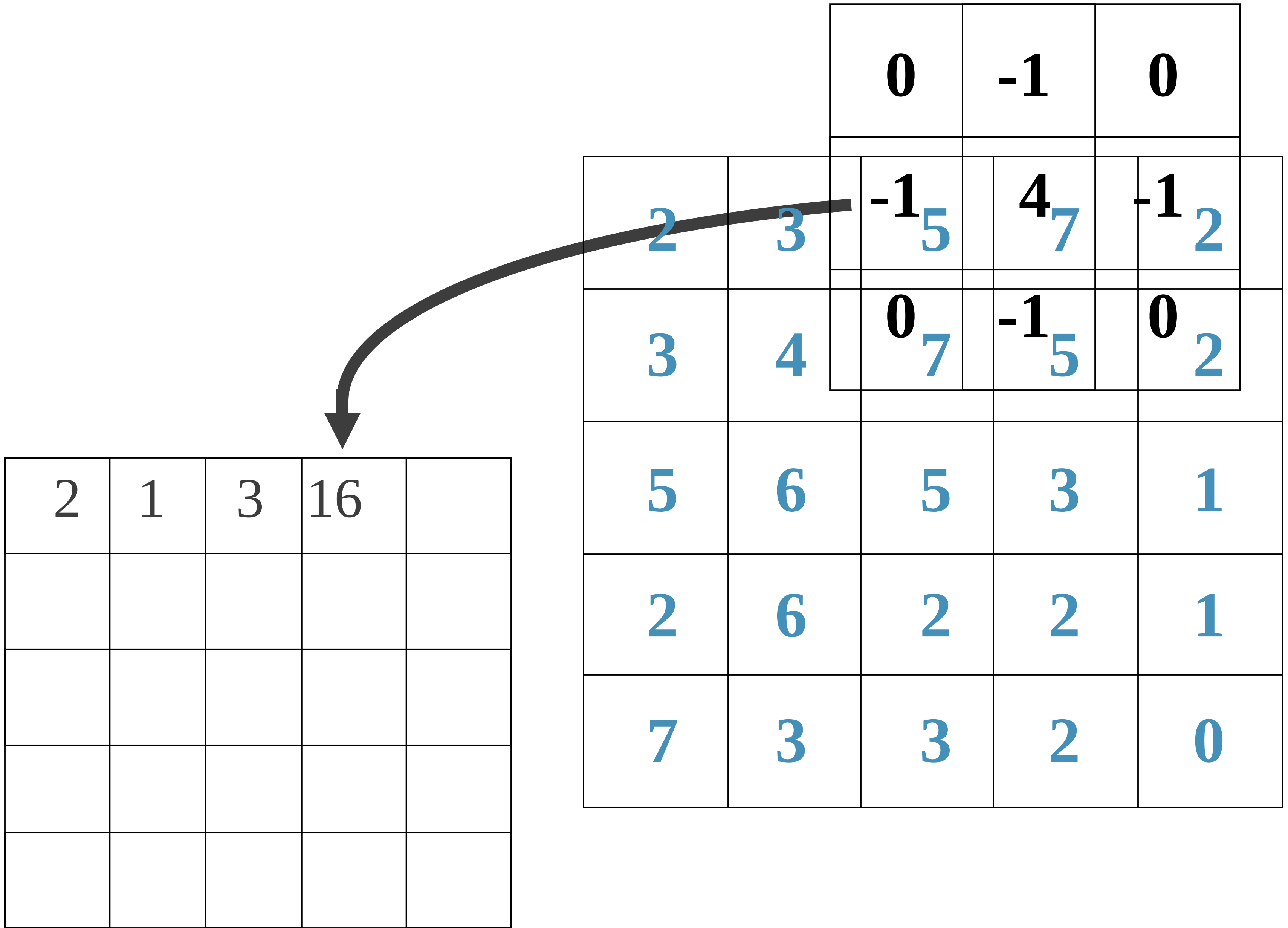
2	3	5	7	2
3	4	7	5	2
5	6	5	3	1
2	6	2	2	1
7	3	3	2	0

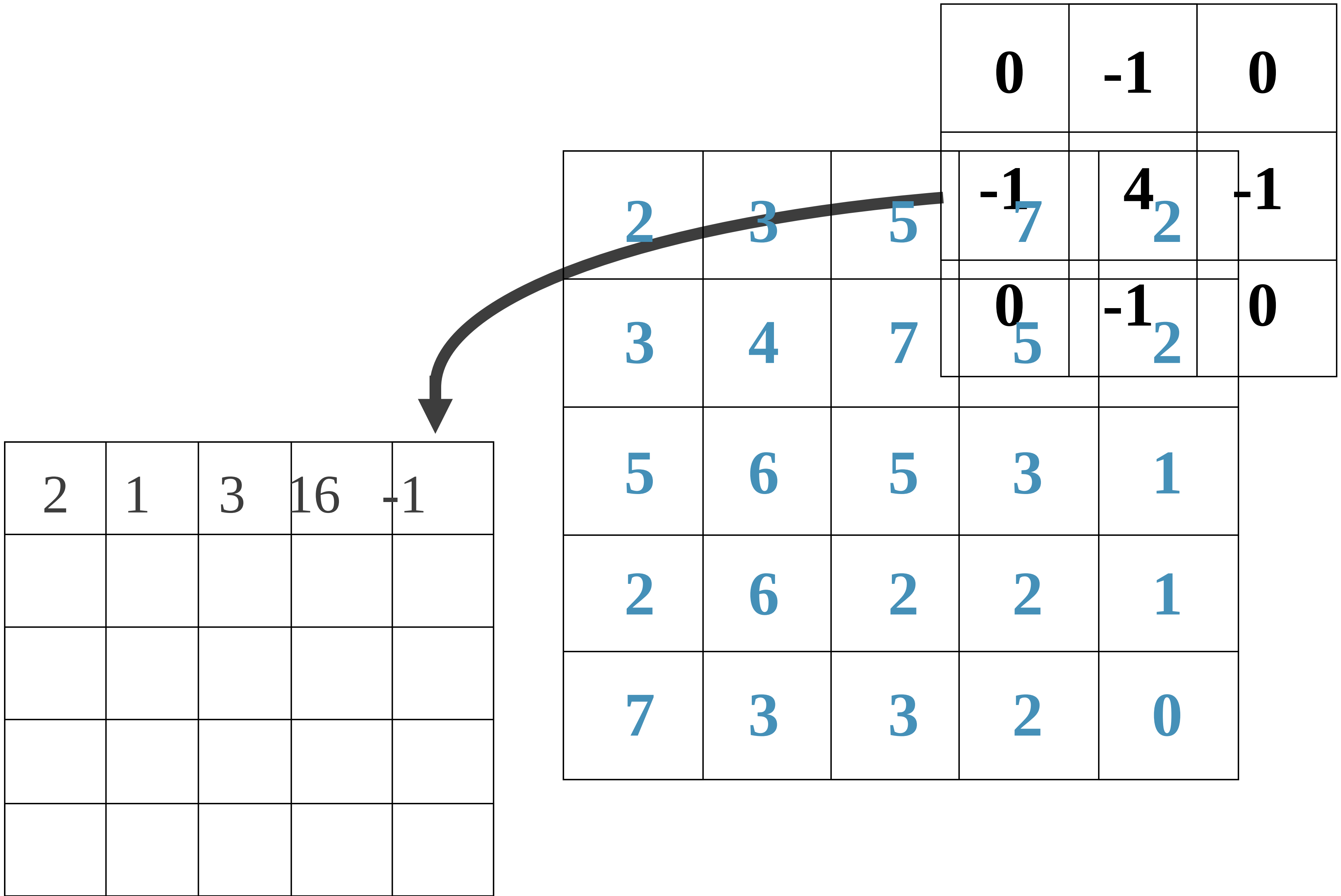

0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2

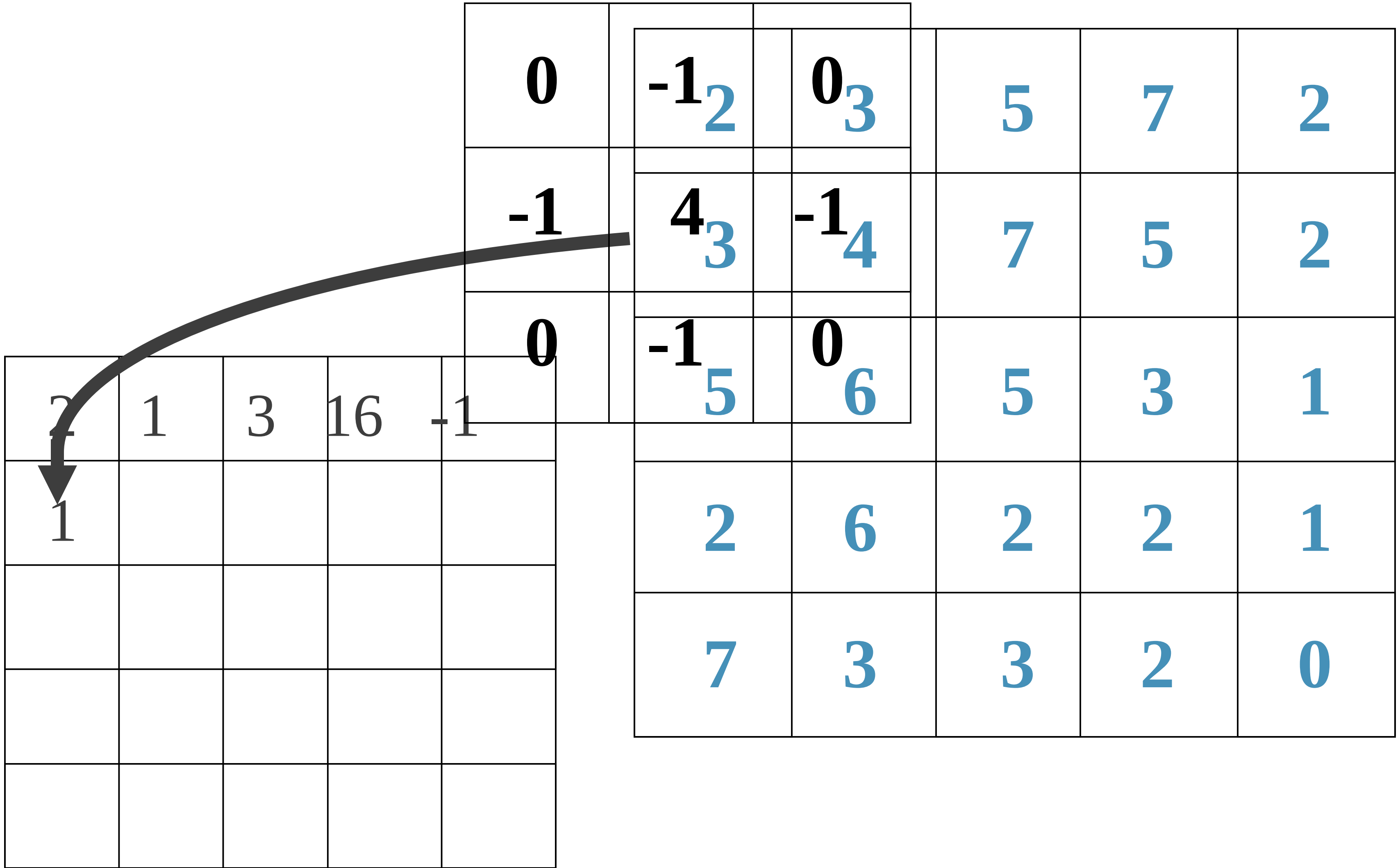

0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1

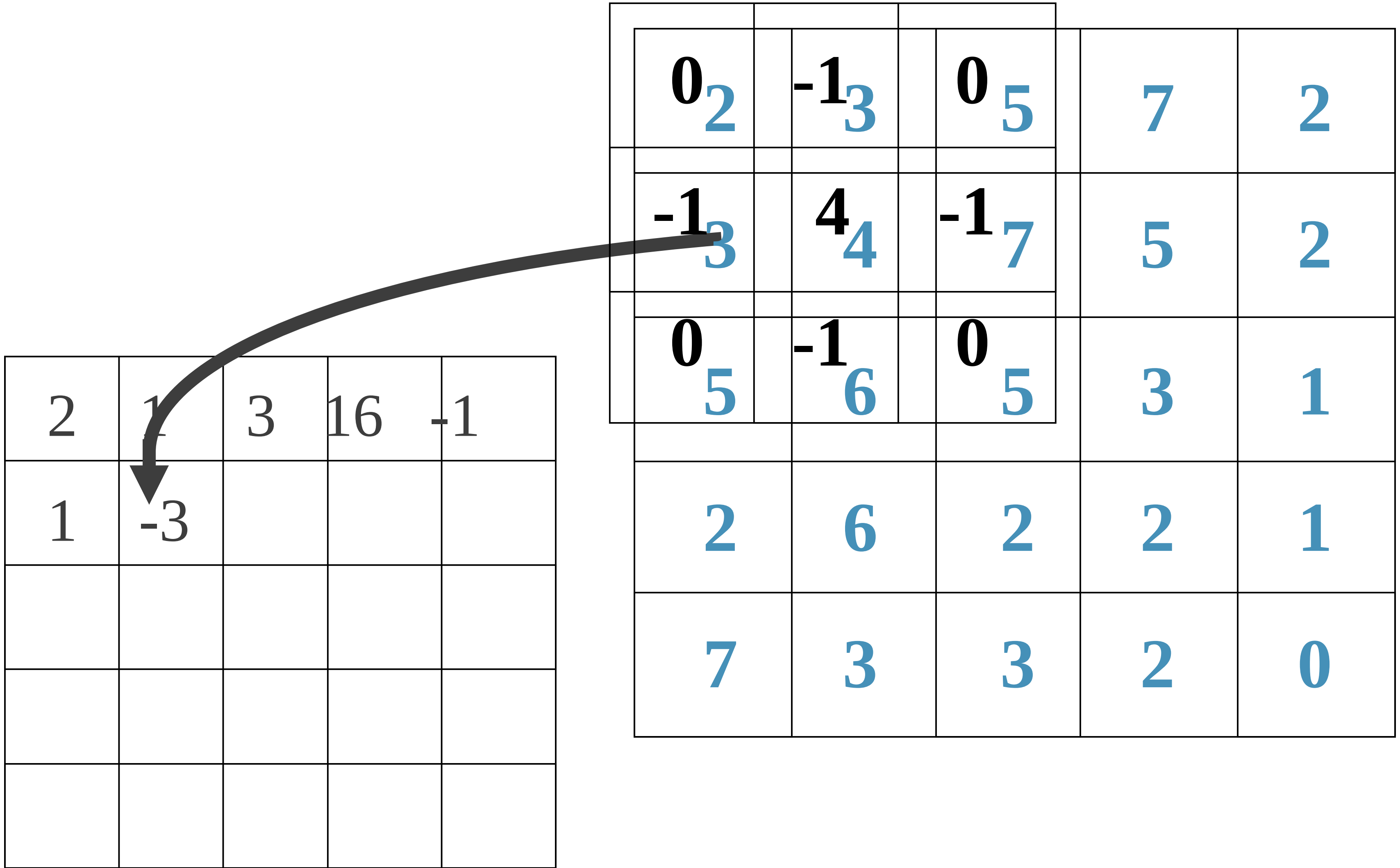


0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1 3


0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1 3 16


0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1 3 16 -1


0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1 3 16 -1
1


0 -1 0
-1 4 -1
0 -1 0
2 3 5 7 2
3 4 7 5 2
5 6 5 3 1
2 6 2 2 1
7 3 3 2 0
2 1 3 16 -1
1 -3