Ψηφιακή Επεξεργασία Εικόνας (ΨΕΕ) – ΜΥΕ037

Εαρινό εξάμηνο 2023-2024

Intensity Transformations (Histogram Processing)

Άγγελος Γιώτης

a.giotis@uoi.gr

Contents

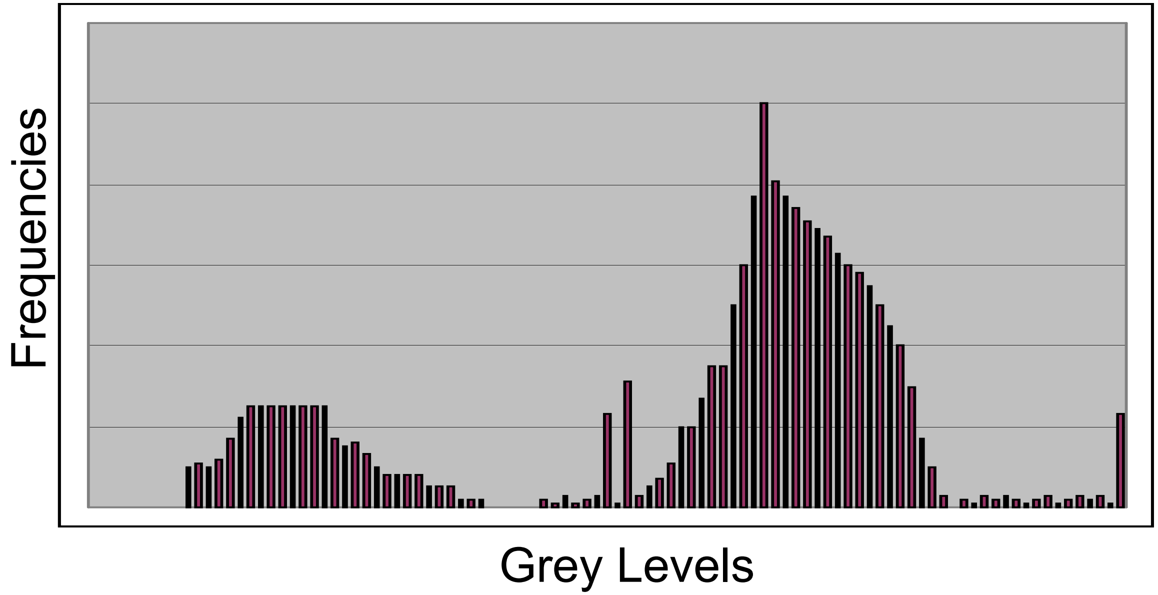
Over the next few lectures we will look at image enhancement techniques working in the spatial domain:

- Histogram processing
- Spatial filtering
- Neighbourhood operations

Image Histograms

The histogram of an image shows us the distribution of grey levels in the image

Massively useful in image processing, especially in segmentation



Histogram processing

- Let r_k, for $k = 0,1,2,\ldots,L-1$, denote the intensities of an *L*-level digital image, $f(x,y)$.

- The *unnormalized histogram* of f is defined as:

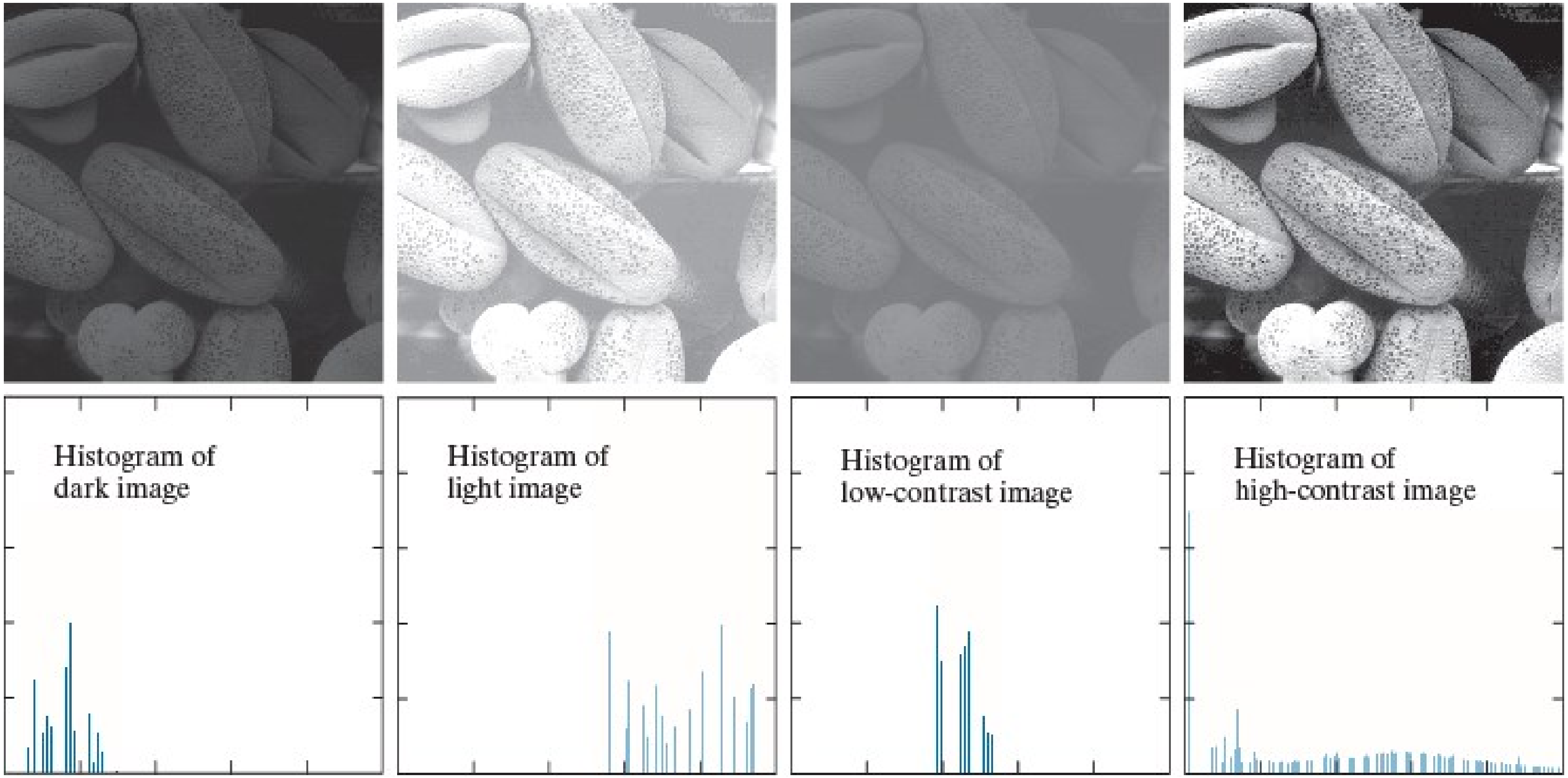
$$h(r_k) = n_k \quad \text{for } k = 0,1,2,\ldots,L-1$$

Where n_k is the number of pixels in f with intensity r_k, and the subdivisions of the intensity scale are called *histogram bins.*

- *Normalized histogram of f:*

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

Histogram Examples



Contrast Stretching

- We can fix images that have poor contrast by applying a pretty simple contrast specification
- The interesting part is how do we decide on this transformation function?

Histogram Equalisation

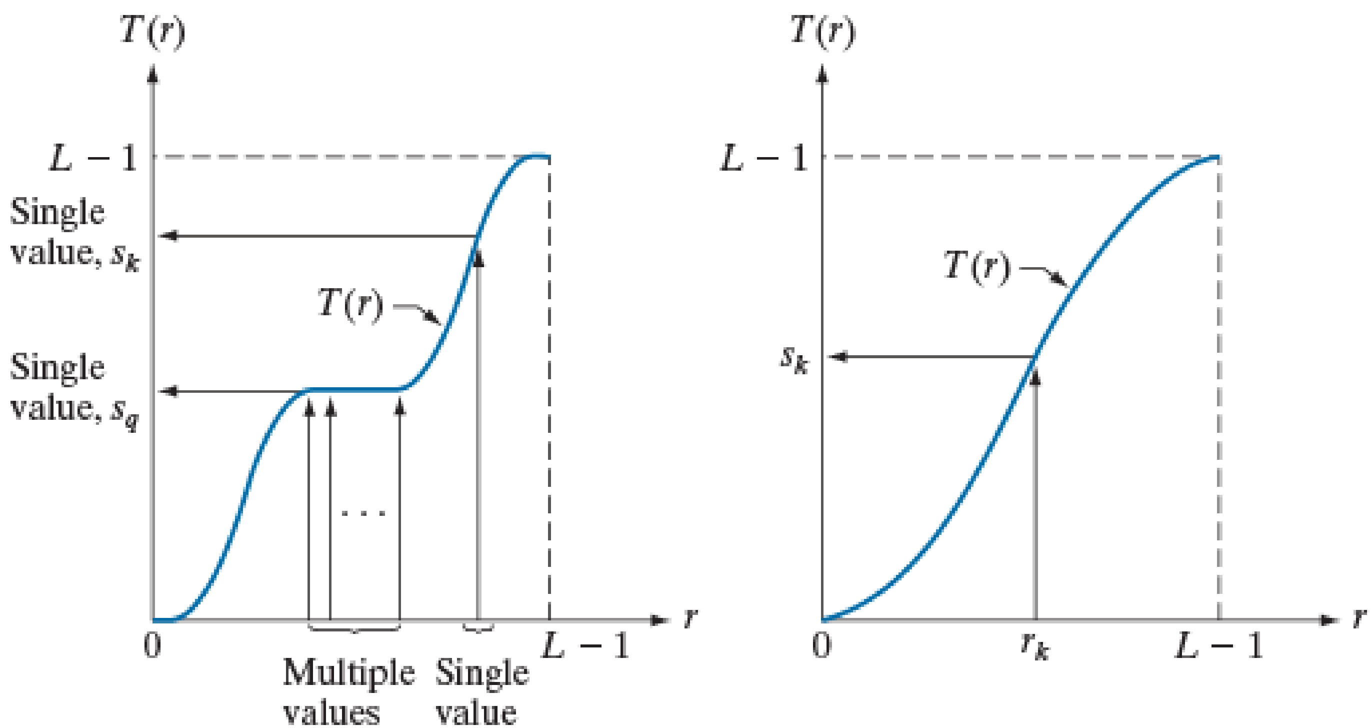
- Spreading out the frequencies in an image (or equalising the image) is a simple way to improve dark or washed out images.
- At first, the continuous case will be studied:
  - r is the intensity of the image in $[0, L\text{-}1]$.
  - we focus on transformations $s=T(r)$:
    - $T(r)$ is monotonically increasing.
    - $T(r)$ must satisfy:

$$0 \le T(r) \le L-1, \; for \;\; 0 \le r \le L-1$$

Histogram Equalisation (cont...)

- The condition for $T(r)$ to be monotonically increasing guarantees that ordering of the output intensity values will follow the ordering of the input intensity values (avoids reversal of intensities).
- If $T(r)$ is **strictly** monotonically increasing then the mapping from s back to r will be 1-1.
- The second condition ($T(r)$ in $[0,1]$) guarantees that the range of the output will be the same as the range of the input.

Histogram Equalisation (cont...)



a) We cannot perform inverse mapping (from s to r).

b) Inverse mapping is possible.

Histogram Equalisation (cont...)

- We can view intensities r and s as random variables and their histograms as probability density functions (pdf) $p_r(r)$ and $p_s(s)$.
- Fundamental result from probability theory:
  - If $p_r(r)$ and $T(r)$ are known and $s=T(r)$ is continuous and differentiable, then

$$p_s(s) = p_r(r)\frac{1}{\left|\frac{ds}{dr}\right|} = p_r(r)\left|\frac{dr}{ds}\right|$$

Histogram Equalisation (cont...)

- The pdf of the output is determined by the pdf of the input and the transformation.
- This means that we can determine the histogram of the output image.
- A transformation of particular importance in image processing is the cumulative distribution function (CDF) of a random variable:

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$

Histogram Equalisation (cont...)

- It satisfies the first condition as the area under the curve increases as r increases.
- It satisfies the second condition as for $r=L-1$ we have $s=L-1$.
- To find $p_s(s)$ we have to compute

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = (L-1)\frac{d}{dr}\int_0^r p_r(w)\,dw = (L-1)p_r(r)$$

Histogram Equalisation (cont...)

Substituting this result:

$$\frac{ds}{dr} = (L-1)\,p_r(r)$$

to

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

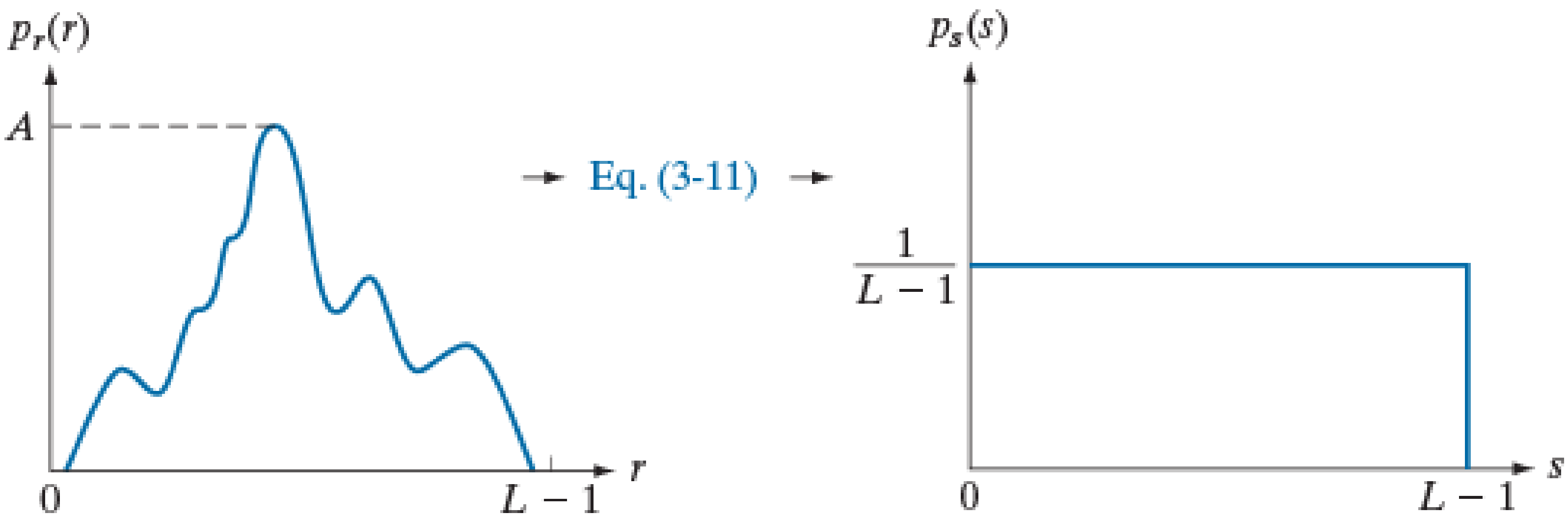
Uniform pdf

yields

$$p_s(s) = p_r(r)\left|\frac{1}{(L-1)\,p_r(r)}\right| = \frac{1}{L-1}, \quad 0 \le s \le L-1$$

Histogram Equalisation (cont...)

- A continuous histogram will always result in a uniform histogram



Histogram Equalisation (cont...)

The formula for histogram equalisation in the discrete case is given

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j$$

where

- r_k: input intensity
- s_k: processed intensity
- n_j: the frequency of intensity j
- MN: the number of image pixels.

Histogram Equalisation (cont...) Example

A 3-bit 64x64 image has the following intensities:

TABLE 3.1 Intensity distribution and histogram values for a 3-bit, 64 × 64 digital image.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

Applying histogram equalization:

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7p_r(r_0) = 1.33$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08$$

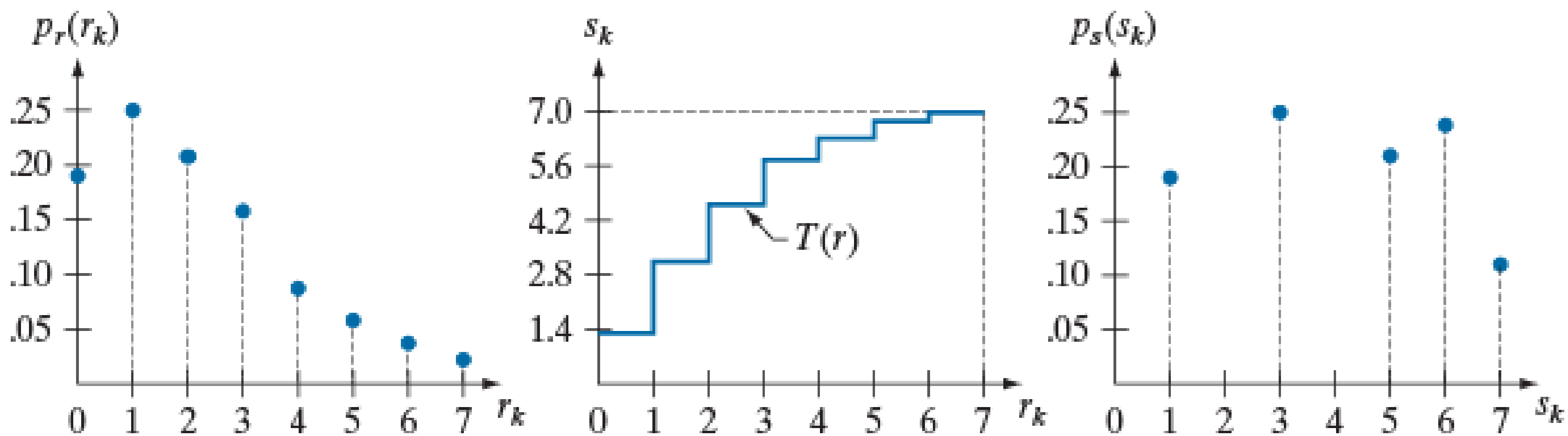
s_2 = ??

Histogram Equalisation (cont...) Example

Rounding to the nearest integer:

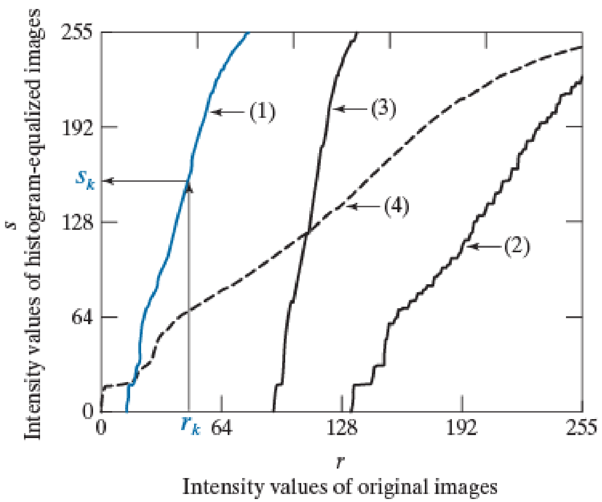
$$s_0 = 1.33 \rightarrow 1 \quad s_1 = 3.08 \rightarrow 3 \quad s_2 = 4.55 \rightarrow 5 \quad s_3 = 5.67 \rightarrow 6$$

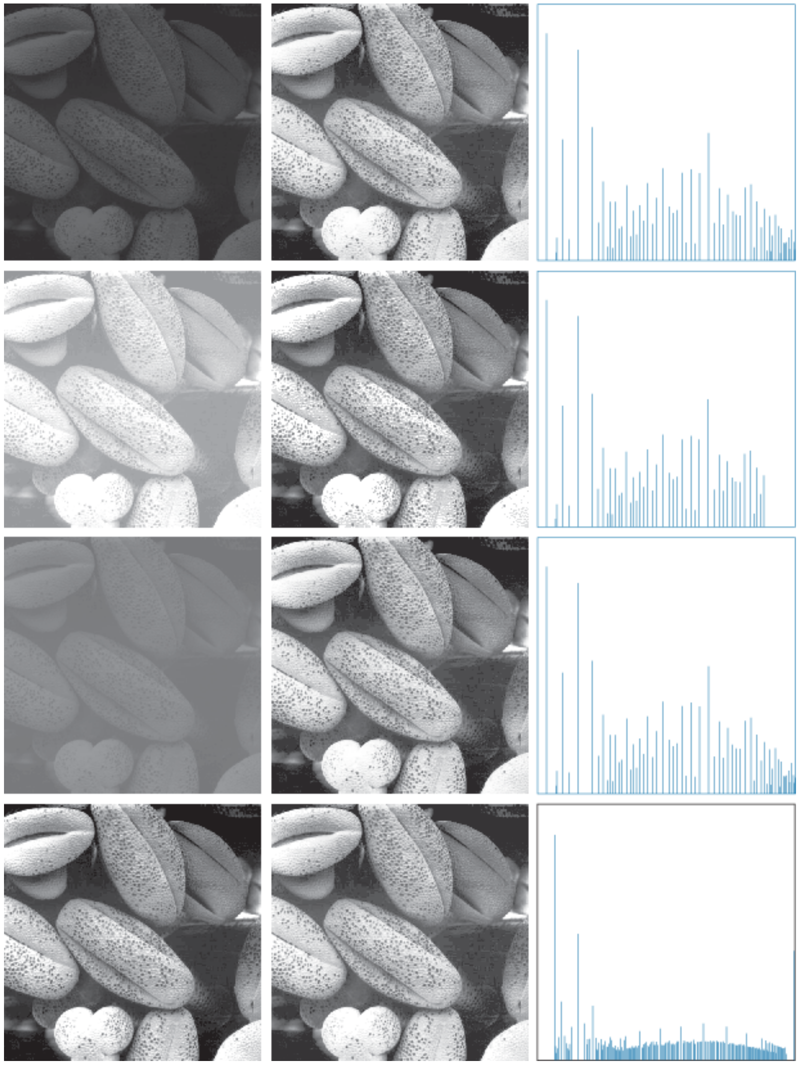
$$s_4 = 6.23 \rightarrow 6 \quad s_5 = 6.65 \rightarrow 7 \quad s_6 = 6.86 \rightarrow 7 \quad s_7 = 7.00 \rightarrow 7$$



Due to discretization, the resulting histogram, though extended, will rarely be perfectly flat.

Histogram equalization example



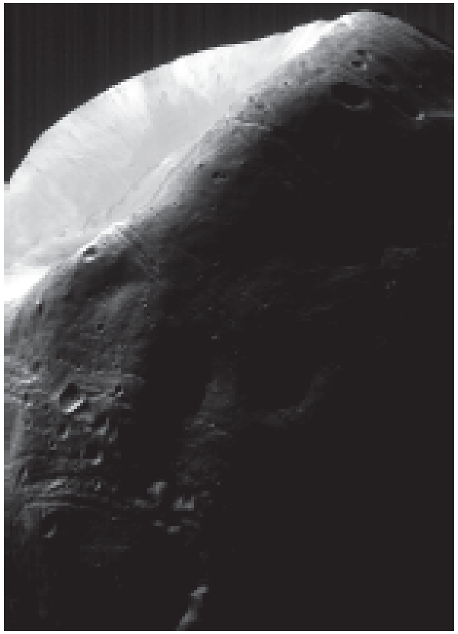


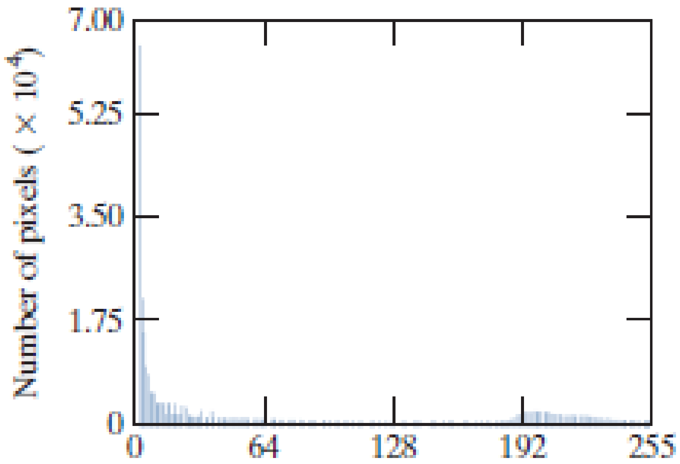
Histogram Matching (Specification)

- Histogram equalization - transformation aims to create an output image with a uniform histogram.
- Beneficial for automatic enhancement - predictable and straightforward implementation.
- However, there are scenarios where histogram equalization may not be suitable.
- Sometimes, it's necessary to **specify** the shape of the histogram for the processed image.
- The technique used to generate images with a specified histogram is known as **histogram matching** or **histogram specification**.

Histogram Specification

- Histogram equalization does not always provide the desirable results.





- Image of Phobos (Mars moon) and its histogram.
- Many values near zero in the initial histogram

Histogram specification (cont.)

- In these cases, it is more useful to specify the final histogram.
- Problem statement:
  - Given $p_r(r)$ from the image and the target histogram $p_z(z)$, estimate the transformation $z=T(r)$.
- The solution exploits histogram equalization.

Histogram specification (cont…)

- Equalize the initial histogram of the image:

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$

- Equalize the target histogram:

$$s = G(z) = (L-1)\int_0^r p_z(w)\,dw$$

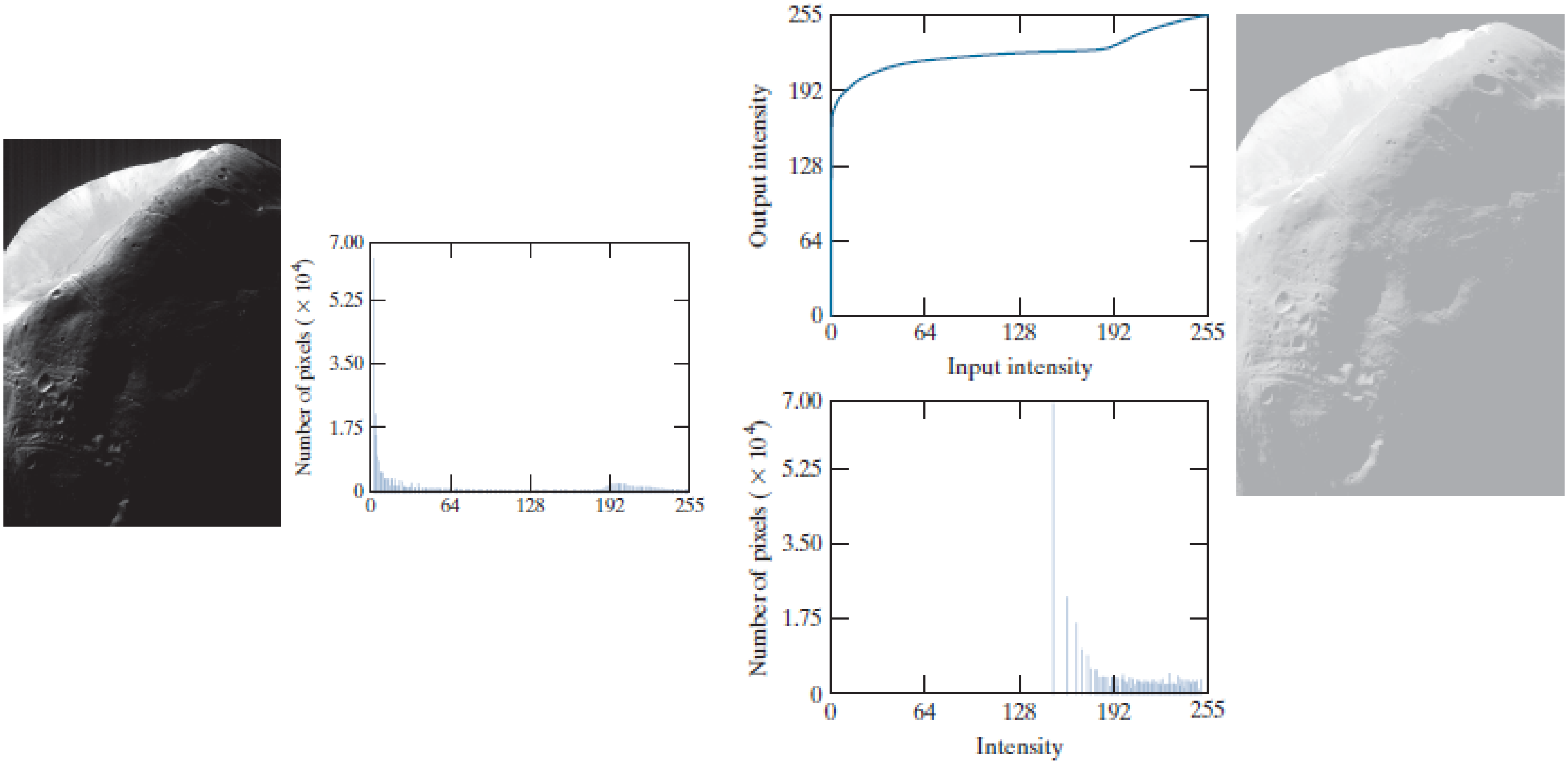
$$G(z) = T(r)$$

- Obtain the inverse transform: $z = G^{-1}(s) = G^{-1}(T(r))$

In practice, for every value of r in the image:

- get its equalized transformation $s=T(r)$.
- perform the inverse mapping $z=G^{-1}(s)$, where $s=G(z)$ is the equalized target histogram.

Histogram Specification (cont...)



Histogram equalization

Histogram specification (cont…)

The discrete case:

•Equalize the initial histogram of the image:

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j$$

$$G(z) = T(r)$$

•Equalize the target histogram:

$$s_k = G(z_q) = (L-1)\sum_{i=0}^{q} p_z(r_i)$$

•Obtain the inverse transform: $z_q = G^{-1}(s_k) = G^{-1}(T(r_k))$

Histogram specification (algo…)

1. Compute the histogram, pr(r), of the input image, and use it to map the intensities in the input to those in the histogram-equalized image. Round the resulting s_k, to the integer range $[0, L-1]$.
2. Compute all values of $G(z_q)$ for $q = 0, .., L-1$, where $p_z(z_i)$ are the values of the specified histogram. Round the values of G to integers in $[0, L-1]$and store them in a lookup table
3. For every s_k, use the stored values of G from Step 2 to find the corresponding value of z_q so that G(z_q) is closest to s_k. Store these mappings from s to z. When more than one value of z_q gives the same match (i.e., the mapping is not unique), choose the smallest value by convention.
4. Form the histogram-specified image by mapping every equalized pixel with value s_k to the corresponding pixel with value z_q in the histogram-specified image, using the mappings found in Step 3.

Histogram Specification (cont...) Example

Consider again the 3-bit 64x64 image:

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

It is desired to transform this histogram to:

$$p_z(z_0) = 0.00 \quad p_z(z_1) = 0.00 \quad p_z(z_2) = 0.00 \quad p_z(z_3) = 0.15$$

$$p_z(z_4) = 0.20 \quad p_z(z_5) = 0.30 \quad p_z(z_6) = 0.20 \quad p_z(z_7) = 0.15$$

with $z_0 = 0, z_1 = 1, z_2 = 2, z_3 = 3, z_4 = 4, z_5 = 5, z_6 = 6, z_7 = 7.$

Histogram Specification (cont...) Example

The first step is to equalize the input (as before):

$$s_0 = 1,\ s_1 = 3,\ s_2 = 5,\ s_3 = 6,\ s_4 = 6,\ s_5 = 7,\ s_6 = 7,\ s_7 = 7$$

The next step is to equalize the output:

$$G(z_0) = 0 \quad G(z_1) = 0 \quad G(z_2) = 0 \quad G(z_3) = 1$$
$$G(z_4) = 2 \quad G(z_5) = 5 \quad G(z_6) = 6 \quad G(z_7) = 7$$

Notice that $G(z)$ is not strictly monotonic. We must resolve this ambiguity by choosing, e.g. the smallest value for the inverse mapping.

Histogram Specification (cont…) Example

Perform inverse mapping: find the smallest value of z_q that provides the closest $G(z_q)$ to s_k:

$s_k = T(r_i)$	$G(z_q)$
$s_0 = 1$	$G(z_0) = 0$
$s_1 = 3$	$G(z_1) = 0$
$s_2 = 5$	$G(z_2) = 0$
$s_3 = 6$	$G(z_3) = 1$
$s_4 = 6$	$G(z_4) = 2$
$s_5 = 7$	$G(z_5) = 5$
$s_6 = 7$	$G(z_6) = 6$
$s_7 = 7$	$G(z_7) = 7$

$$s_k \rightarrow z_q$$

$$1 \rightarrow 3$$

$$3 \rightarrow 4$$

$$5 \rightarrow 5$$

$$6 \rightarrow 6$$

$$7 \rightarrow 7$$

e.g. every pixel with value $s_0=1$ in the histogram-equalized image would have a value of 3 (z_3) in the histogram-specified image.

Histogram Specification (cont...) Example

Notice that due to discretization, the resulting histogram will rarely be exactly the same as the desired histogram.

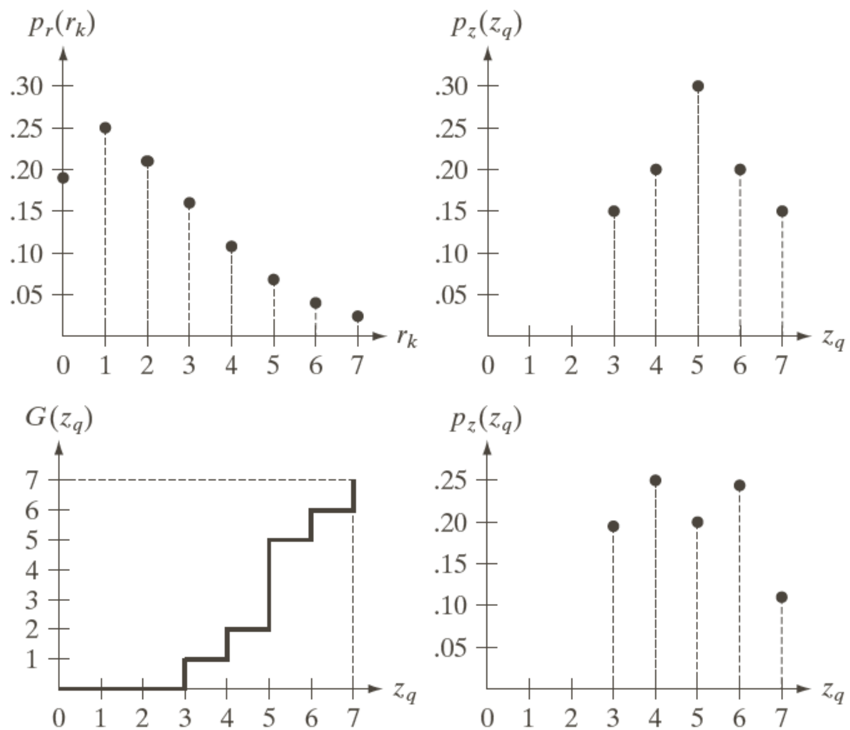
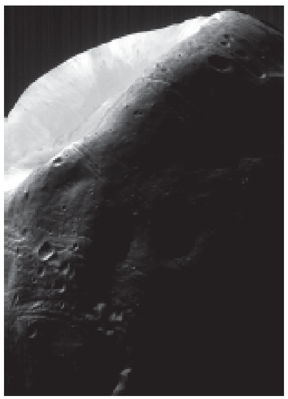
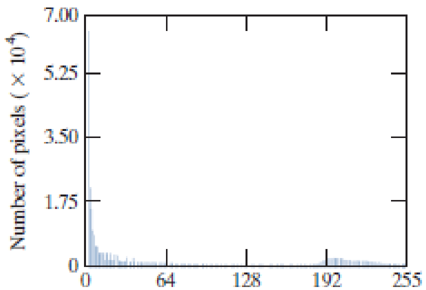
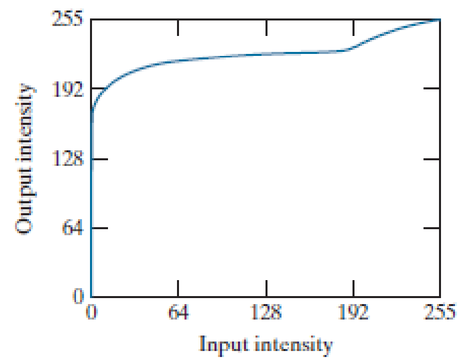


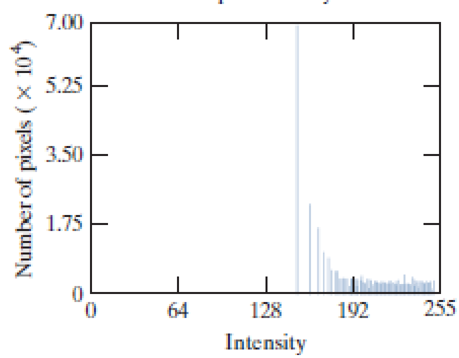
FIGURE 3.22 (a) Histogram of a 3-bit image. (b) Specified histogram. (c) Transformation function obtained from the specified histogram. (d) Result of performing histogram specification. Compare (b) and (d).

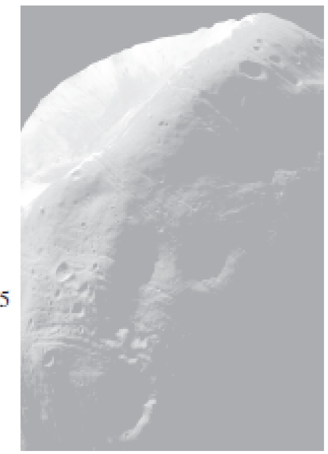
Histogram Specification (cont...)









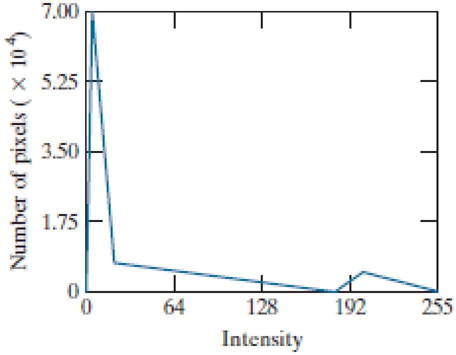


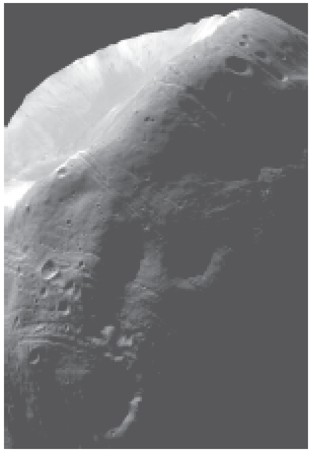
Original image

Histogram equalization

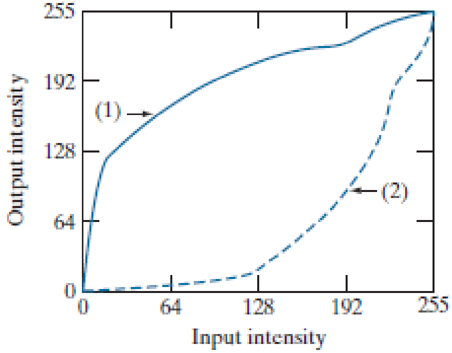
Histogram Specification (cont...)

Specified histogram

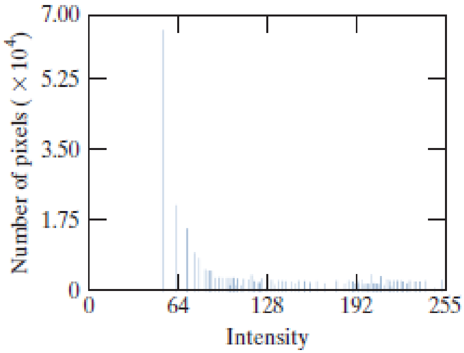




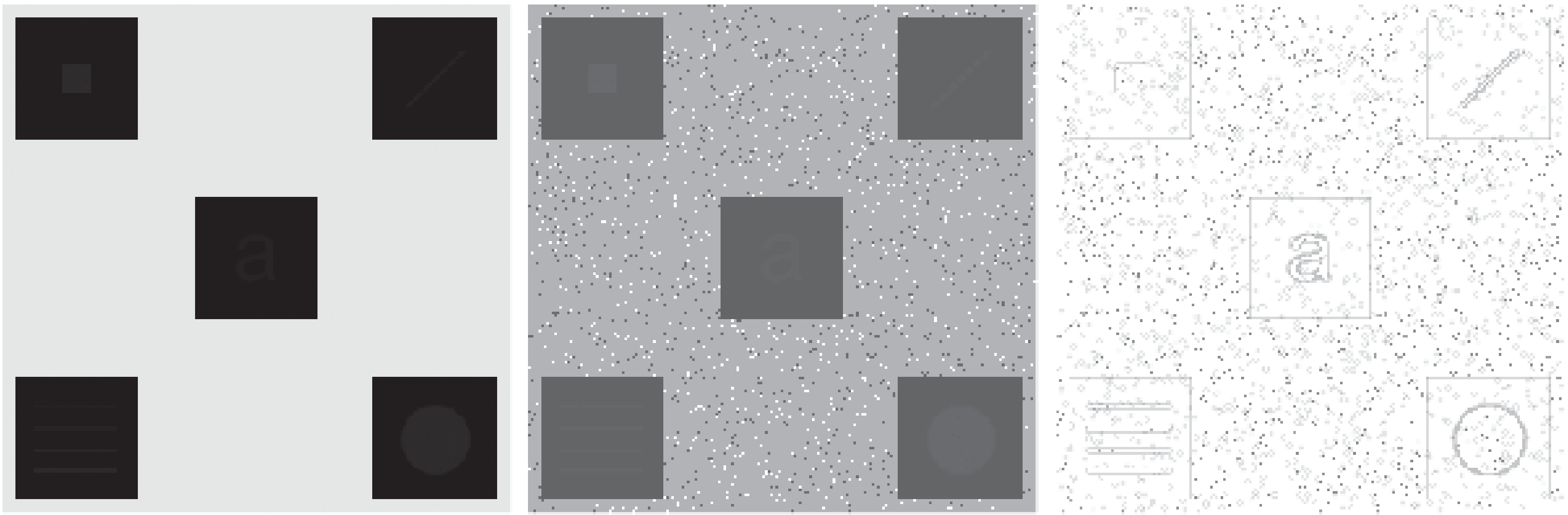
Transformation function and its inverse



Resulting histogram

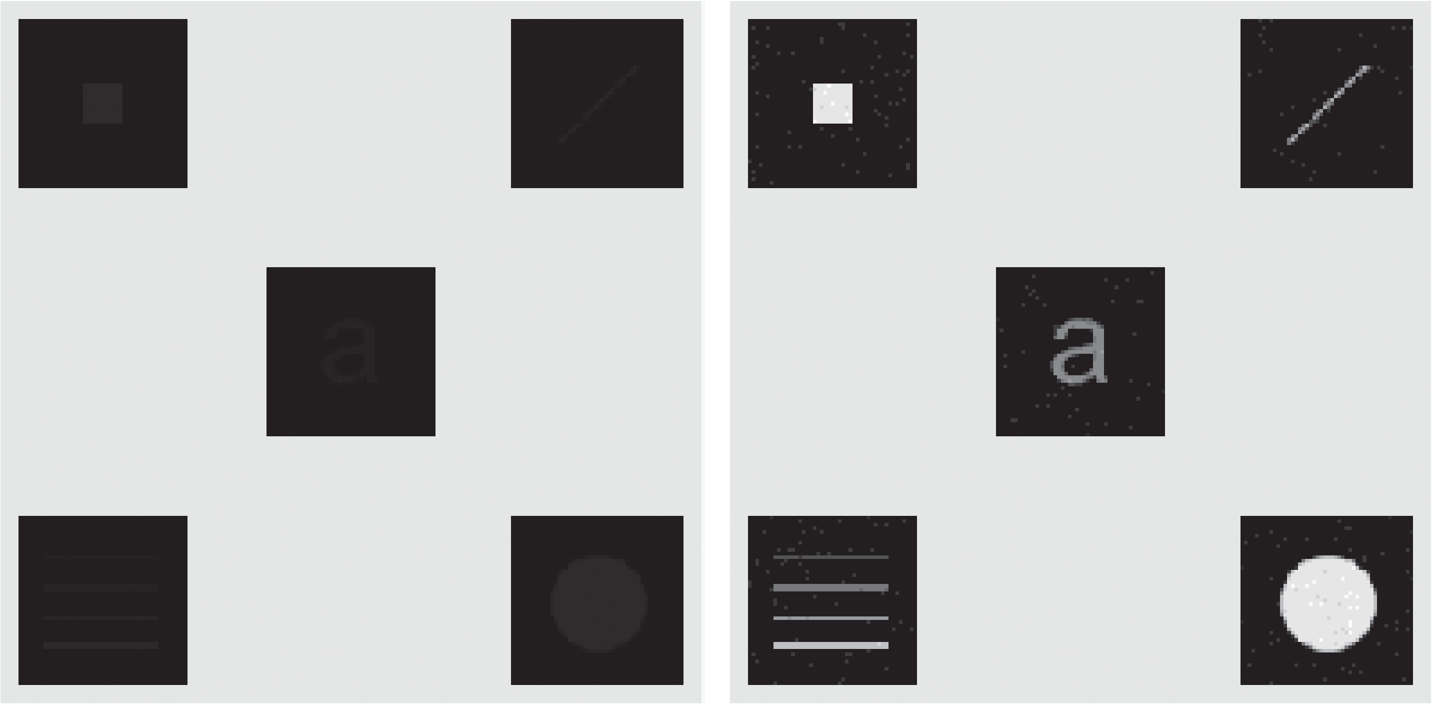


Local Histogram Processing



- Image in (a) is slightly noisy but the noise is imperceptible.
- HE enhances the noise in smooth regions (b).
- Local HE reveals structures having values close to the values of the squares and small sizes to influence HE (c).

Local Histogram Processing



Summary

We have looked at:

- Different kinds of image enhancement
- Histograms
- Histogram equalisation
- Histogram specification

Next we will start to look at spatial filtering and neighbourhood operations