Image Preprocessing
Gonzalez
Spatial Preprocessing

Ahmad Asadi

March 1, 2016

Date Performed: February 21, 2016
Instructor: Professor Safabakhsh

1 Overview and Definitions

Spatial Domain Process Formally is defined as:

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

where T is an operator defined over a neighbourhood of (x, y) on f. Also it can be defined on a set of input images.

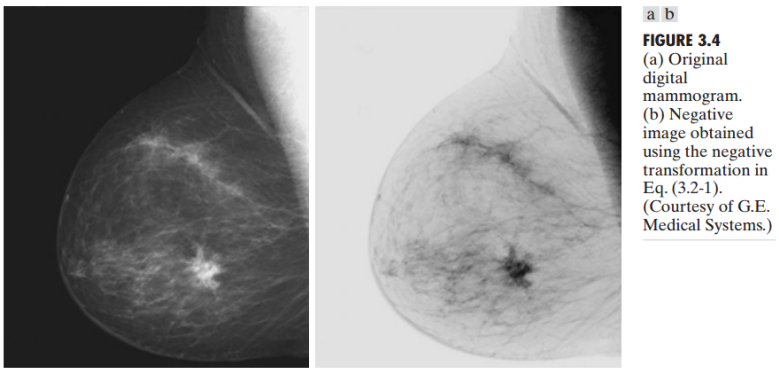
Definition of Neighbourhood Usually as a **square or rectangular subimage** centered at (x, y).

Gray-level transformation If neighbourhood radius is equal to 1 then T is a **Gray-level**, or **Intensity** or **Mapping**, transformation function. It is:

$$s = T(r)$$

where s and r are respectively the gray level of $g(x, y)$ and $f(x, y)$.
Techniques using gray-level transformation, as they operate on just a single point, are mostly referred to as *point processing*.

Mask Masks or filters are 2D arrays with a specific size in which inner weights define the application and effect of the mask.



a b

FIGURE 3.4
(a) Original digital mammogram. (b) Negative image obtained using the negative transformation in Eq. (3.2-1). (Courtesy of G.E. Medical Systems.)

Figure 1: This figure illustrates evident contribution of reversing the intensity (negating image) to illustrate existence of lesion in a part of body.

2 Some Basic Gray-level Transformations

Negative Image A negative image of an input image with gray levels in the range [0 , L-1] is defined as:

$$s = (L-1) - r$$

Following is an example of application:

Log Transformation The general form of this transformation is as follows:

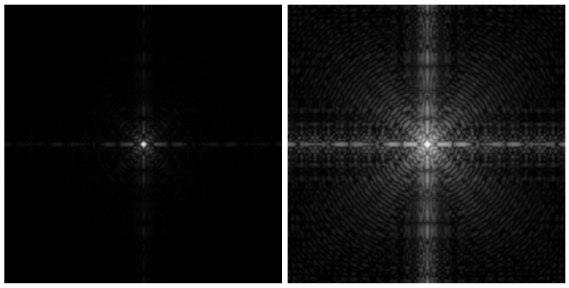
$$s = c \log^{(1+r)}$$

This transformation maps a narrow range of low gray-levels in input image to a wider range in output.
This transformation is appropriate for applications in which having more details of darker parts of image along with bright parts makes worthwhile contributions to accomplish the required task.

Power-Law Transformation This kind of transformation has a basic form as follows.

$$s = cr^{\gamma}$$

This transformation is sort of generalization for log-transformation. As it is displayed in fig3, in which plots of s versus r for various values of γ are shown, with lower values of γ than 1 this transformation operates as



a b

FIGURE 3.5
(a) Fourier spectrum.
(b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.

Figure 2: Log transformation effect on input picture

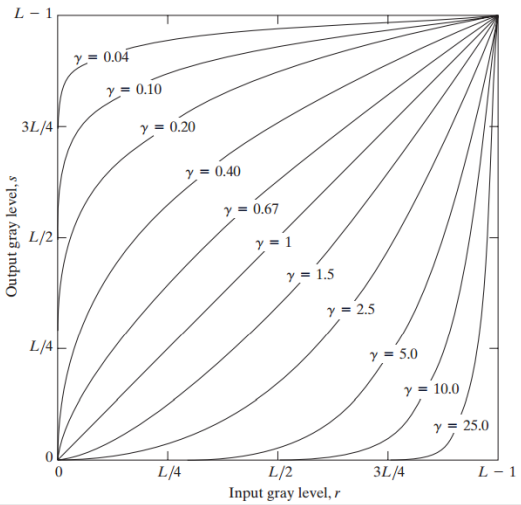


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

Figure 3: Plots of s versus r for various values of γ

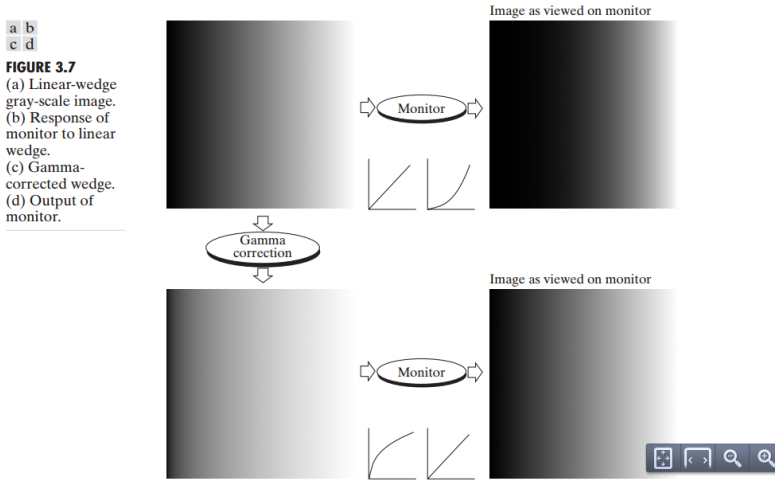


Figure 4: Power-Law Transformation result and effect on an input image

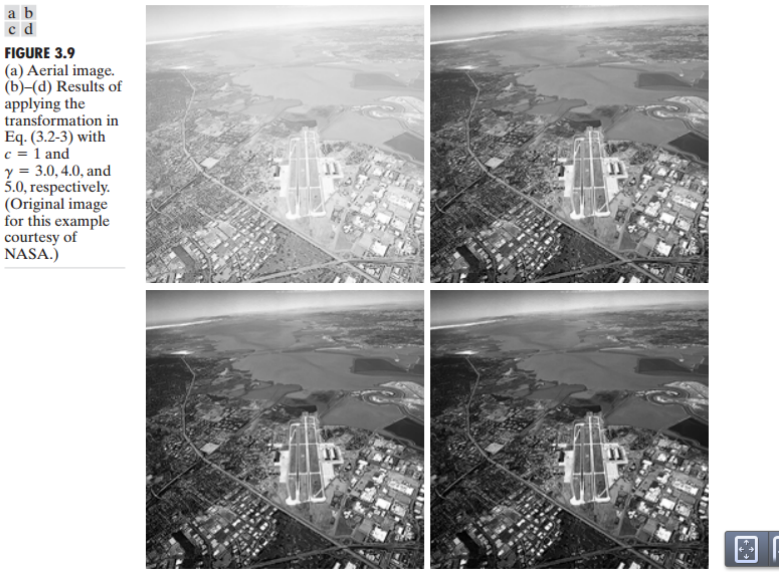
a log-transformation and in cases with γ values larger than 1, it operates totally inversely.

One of the most important and useful applications of this transformation is **gamma correction** in which value ranges in an input image would be mapped to a more appropriate range for certain image displayers to generate fitter images. The value of γ in different applications corresponds to the using device and its properties. In fig4 in gamma correction process is $\gamma = 0.4$.
Note that, varying the value of gamma correction changes not only the brightness, but also the ratios of red to green to blue.

Note that, power-law transformation also affects image's contrast. Fig5 illustrate the effect of this transformation on image's contrast. In this picture, the value of γ increases in each case.

Piecewise-Linear Transformation Functions This kind of transformation, fragments a transformation range to several subtransformations each specified in a disjoint range of overall transformation's definition range. Thus in each subtransformation any of above-mentioned transformations are applicable. Event though, this method provides us a great opportunity to make more complex and more advantageous transformations, it needs more user input.
There are some examples of this method's applications.

Contrast Stretching Creating a transformation function which is partially defined for different ranges for (r, s) pairs. Fig6 demonstrates such an example function.



a b
c d

FIGURE 3.9
(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0$, and 5.0, respectively. (Original image for this example courtesy of NASA.)

Figure 5: An illustration of power-law transformation's effect on image contrast.

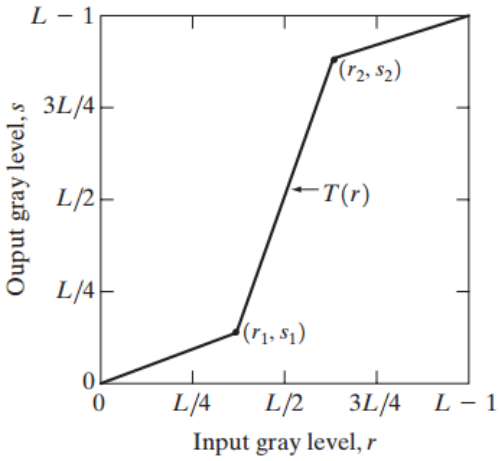


Figure 6: An example of piecewise-transformation function for contrast stretching.

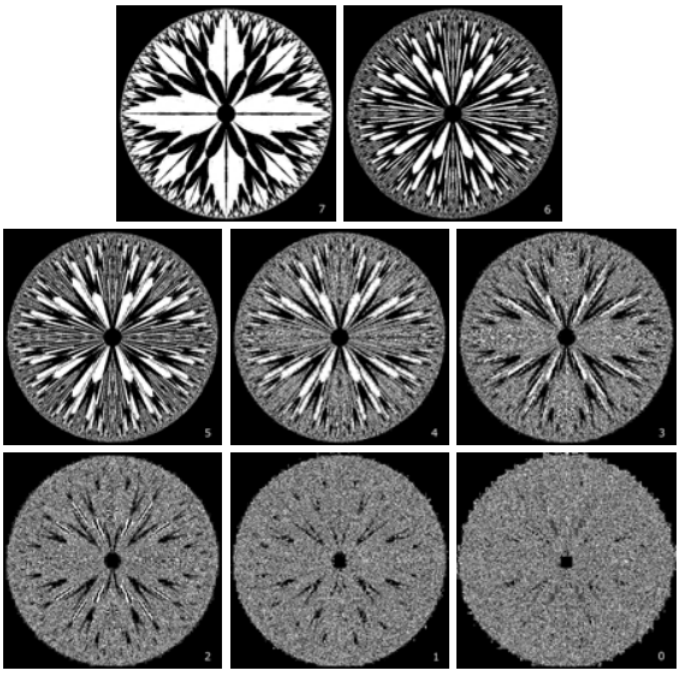


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

Figure 7: Applying a bit-plane slicing transformation on bit planes of a fractal image in order to separate bit-planes.

Gray-Level Slicing This kind of transformation is widely used in making a specified gray-level range of input image more salience.

Bit-Plane Slicing Separating bit planes of an image can make worthwhile contribution in achieving better results in applications. Fig7 displays application of such transformation on an input image of a fractal.

3 Histogram Processing

Definition A histogram of a gray level image with gray level range $[0, L-1]$ is a function $h(r_k) = n_k$ where r_k is kth gray level and n_k is the number of pixels with gray level r_k. Note that normalization gives an estimate of the probability of a pixel having gray level r_k, as follows:

$$P(r_k) = \frac{n_k}{n}.$$

Histogram Equalization We define a transformation in following format:

$$s = T(r)$$

Where r is the gray level of a pixel, s is a new gray level for pixel and $T(.)$ is a transformation function satisfying following conditions:

a Is *monotonically increasing* in range $[0, 1]$ and is *single_vaued*

b $0 \leq T(r) \leq 1$ for $0 \leq r \leq 1$

Theorem Taking two random variables r and s, given their probability density functions p_r and p_s, where $s = T(r)$, with known transformation function T satisfying conditions (a) and (b), where T^{-1} satisfies condition (a), p_s can be obtained simply from formula:

$$p_s = p_r \cdot \left| \frac{dr}{ds} \right|$$

Histogram Smoothing Taking *cumulative density function (CDF)* as transformation function, gives a transformation T smoothing the histogram of an image:

$$s = T(r) = CDF_r = \int_0^r p_r(\omega) d\omega$$
$$\frac{ds}{dr} = \frac{dT}{dr} = \frac{d \int_0^r p_r(\omega) d\omega}{dr} = \frac{p_r}{dr}$$
$$p_s = p_r \cdot |\frac{dr}{ds}| \Rightarrow p_s = p_r \cdot |\frac{1}{\frac{ds}{dr}}| = p_r \cdot |\frac{1}{p_r}| = 1$$
$$p_s = 1 \Rightarrow s \text{ is drawn from a } uniform \text{ distribution}$$

Histogram Matching (Specification) The method used to generate image having a *specific histogram shape* is called histogram matching.
Assume the given image's gray level is denoted by r and the specific gray level that we would achieve on processed image is denoted by z.
Given:

$$p_r \text{ and } p_z \text{ for input and desired output image.}$$

Define:

$$G(z) = \int_0^s p_s(\omega) d\omega = s$$

From before, we have:

$$s = T(r) = \int_0^r p_r(\theta) d\theta$$

Thus:

$$G(z) = \int_0^s p_s(\omega) d\omega = \int_0^r p_r(\theta) d\theta = s = T(r)$$

Therefore:

$$z = G^{-1}(s) = G^{-1}(T(r))$$

Note that: G is computable given p_z and then if $G^{-1}(z)$ exists, it can be computed by above-mentioned equation.

Implementation in discrete mode Denote each set of gray levels by $\{r_j\}$, $\{s_j\}$ and $\{z_j\}$, where $j = 0, 1, \cdots, L-1$.
Assume that $T(r)$ is a transformation function demonstrated by a table specifying the value of s_i for r_i directly. By applying this transformation, histogram-equalized image can be obtained. Also, given p_z we can compute $G(z)$ as a transformation function by computing *CDF* of p_z. Correspondingly, given each $v_i = G(z_i)$ we can compute $z_i = G^{-1}(v_i)$ having transformation function $G(z)$. According to definitions, $v = s$, so we can compute z_i using this process for each r_i of input image's gray levels. This final transformation, also, could be implemented using a table. Having this final transformation table, we can obtain each final value z_i for each input r_i directly and generate corresponding *histogram-specified* output image.

4 Local Enhancement

Although this global approach is suitable for overall enhancement, there are cases in which it is necessary to enhance details over small areas in an image. The histogram processing techniques previously described are easily adaptable to local enhancement. The procedure is to define a square or rectangular neighborhood and move the center of this area from pixel to pixel. At each location, the histogram of the points in the neighborhood is computed and either a histogram equalization or histogram specification transformation function is obtained. This function is finally used to map the gray level of the pixel centered in the neighborhood.

5 Using Histogram Statistics