HISTOGRAM PROCESSING

Let r_k, for $k = 0,1,2, \dots, L-1$, denote the intensities of an L-level digital image, $f(x, y)$. The unnormalized histogram of f is defined as

$$h(r_k) = n_k \quad for\ k = 0,1,2, \dots, L-1$$

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. Similarly, the normalized histogram of f is defined as

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

where, as usual, M and N are the number of image rows and columns, respectively. Mostly, we work with normalized histograms, which we refer to simply as *histograms* or *image histograms*. The sum of $p(r_k)$ for all values of k is always 1. The components of $p(r_k)$ are estimates of the probabilities of intensity levels occurring in an image. As you will learn in this section, histogram manipulation is a fundamental tool in image processing. Histograms are simple to compute and are also suitable for fast hardware implementations, thus making histogram-based techniques a popular tool for real-time image processing.

Histogram shape is related to image appearance. For example, Fig. 1 shows images with four basic intensity characteristics: dark, light, low contrast, and high contrast; the image histograms are also shown. We note in the dark image that the most populated histogram bins are concentrated on the lower (dark) end of the intensity scale. Similarly, the most populated bins of the light image are biased toward the higher end of the scale. An image with low contrast has a narrow histogram located typically toward the middle of the intensity scale, as Fig. 1(c) shows. For a monochrome image, this implies a dull, washed-out gray look. Finally, we see that the components of the histogram of the high-contrast image cover a wide range of the intensity scale, and the distribution of pixels is not too far from uniform, with few bins being much higher than the others. Intuitively, it is reasonable to conclude that an image whose pixels tend to occupy the entire range of possible intensity levels and, in addition, tend to be distributed uniformly, will have an appearance of high contrast and will exhibit a large variety of gray tones. The net effect will be an image that shows a great deal of gray-level detail and has a high dynamic range. As you will see shortly, it is possible to develop a transformation function that can achieve this effect automatically, using only the histogram of an input image.

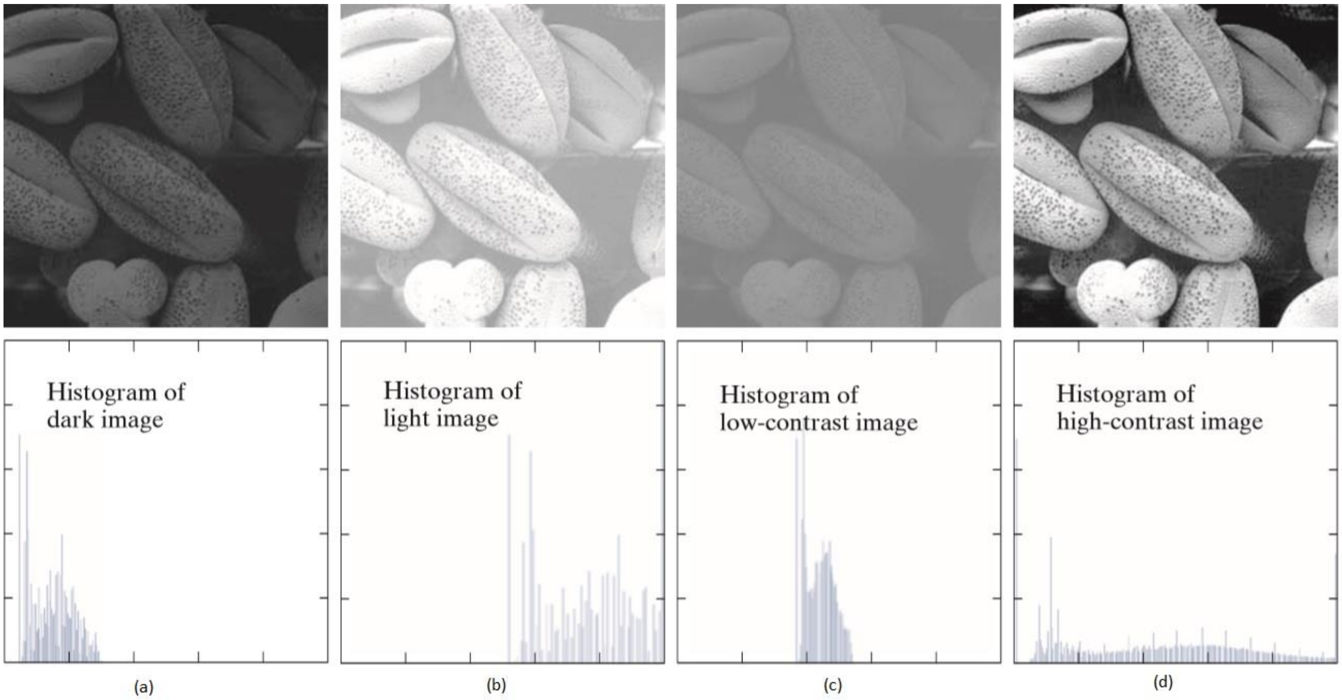

Figure 1: Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$

HISTOGRAM EQUALIZATION

Histogram equalization is a technique for adjusting image intensities to enhance contrast. Assuming initially continuous intensity values, let the variable r denote the intensities of an image to be processed. As usual, we assume that r is in the range $[0, L-1]$, with $r = 0$ representing black and $r = L-1$ representing white. For r satisfying these conditions, we focus attention on transformations (intensity mappings) of the form

$$s = T(r) \qquad 0 \le r \le L-1$$

that produce an output intensity value, s, for a given intensity value r in the input image. We assume that

a) $T(r)$ is a monotonic increasing function in the interval $0 \le r \le L-1$; and
b) $0 \le T(r) \le L-1$ for $0 \le r \le L-1$.

Recall that the probability of occurrence of intensity level r_k in a digital image is approximated by

$$p_r(r_k) = \frac{n_k}{MN}$$

The transformation is

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \qquad k = 0,1,2,\ldots,L-1 \qquad (1)$$

where, as before, L is the number of possible intensity levels in the image (e.g., 256 for an 8-bit image). Thus, a processed (output) image is obtained by using the equation above to map each pixel in the input image with intensity r_k into a corresponding pixel with level s_k in the output image, This is called a histogram equalization or histogram linearization transformation. It is not difficult to show that this transformation satisfies conditions (a) and (b) stated previously in this section.

HISTOGRAM MATCHING (SPECIFICATION)

As explained in the last section, histogram equalization produces a transformation function that seeks to generate an output image with a uniform histogram. When automatic enhancement is desired, this is a good approach to consider because the results from this technique are predictable and the method is simple to implement. However, there are applications in which histogram equalization is not suitable. In particular, it is useful sometimes to be able to specify the shape of the histogram that we wish the processed image to have. The method used to generate images that have a specified histogram is called *histogram matching* or *histogram specification*.

Given an input image, a specified histogram, $p_z(z_i), i = 0,1,2,\ldots,L-1$ and recalling that the $s_k{}'s$ are the values resulting from Eq. (1), we may summarize the procedure for discrete histogram specification as follows:

1. Compute the histogram, $p_r(r)$, of the input image, and use it in Eq. (1) to map the intensities in the input image to the intensities in the histogram-equalized image. Round the resulting values, s_k, to the integer range $[0, L-1]$.
2. Compute all values of function $G(z_q)$ using the Equation
$$G(z_q) = (L-1)\sum_{i=0}^{q} p_z(z_i)$$
for $q = 0,1,2,\ldots,L-1$ where $p_z(z_i)$ are the values of the specified histogram. Round the values of G to integers in the range $[0, L-1]$. Store the rounded values of G in a lookup table.
3. For every value of $s_k, k = 0,1,2,\ldots,L-1$ 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.

LOCAL HISTOGRAM PROCESSING

The histogram processing methods discussed thus far are global, in the sense that pixels are modified by a transformation function based on the intensity distribution of an entire image. This global approach is suitable for overall enhancement, but generally fails when the objective is to enhance details over small areas in an image. This is because the number of pixels in small areas have negligible influence on the computation of global transformations. The solution is to devise transformation functions based on the intensity distribution of pixel neighborhoods.

The histogram processing techniques previously described can be adapted to local enhancement. The procedure is to define a neighborhood and move its center from pixel to pixel in a horizontal or vertical direction. 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 used to map the intensity of the pixel centered in the neighborhood. The center of the neighborhood is then moved to an adjacent pixel location and the procedure is repeated. Because only one row or column of the neighborhood changes in a one-pixel translation of the neighborhood, updating the histogram obtained in the previous location with the new data introduced at each motion step is possible. This approach has obvious advantages over repeatedly computing the histogram of all pixels in the neighborhood region each time the region is moved one pixel location. Another approach used sometimes to reduce computation is to utilize nonoverlapping regions, but this method usually produces an undesirable “blocky” effect.

Figure 2(a) is an 8-bit, 512×512 image consisting of five black squares on a light gray background. The image is slightly noisy, but the noise is imperceptible. There are objects embedded in the dark squares, but they are invisible for all practical purposes. Figure 2(b) is the result of global histogram equalization. As is often the case with histogram equalization of smooth, noisy regions, this image shows significant enhancement of the noise. However, other than the noise, Fig. 2(b) does not reveal any new significant details from the original. Figure 2(c) was obtained using local histogram equalization of Fig. 2(a) with a neighborhood of size 3×3. Here, we see significant detail within all the dark squares. The intensity values of these objects are too close to the intensity of the dark squares, and their sizes are too small, to influence global histogram equalization significantly enough to show this level of intensity detail

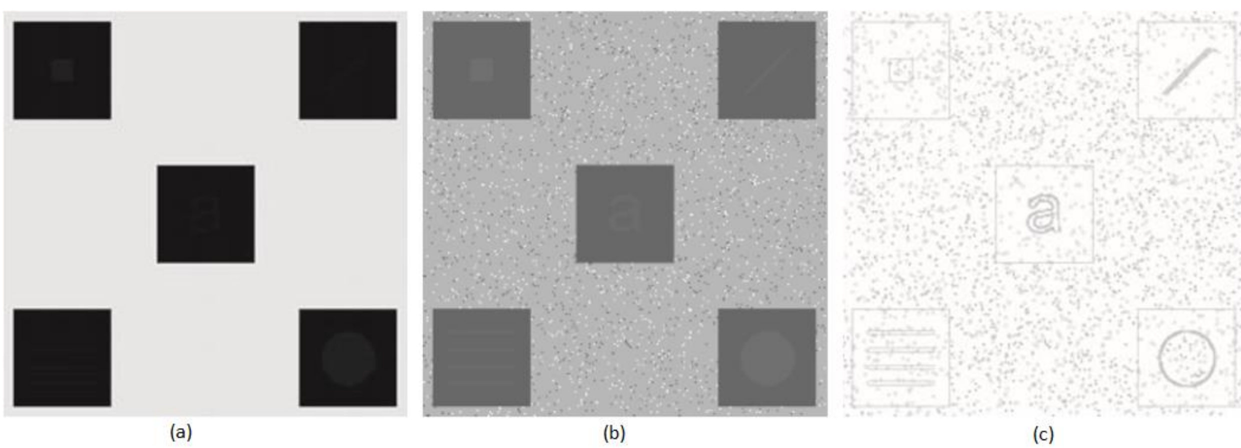

Figure 2: (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization.

References and further reading:

Digital Image Processing, 4th edition, Gonzalez, Rafael and Woods, Richard, 2018