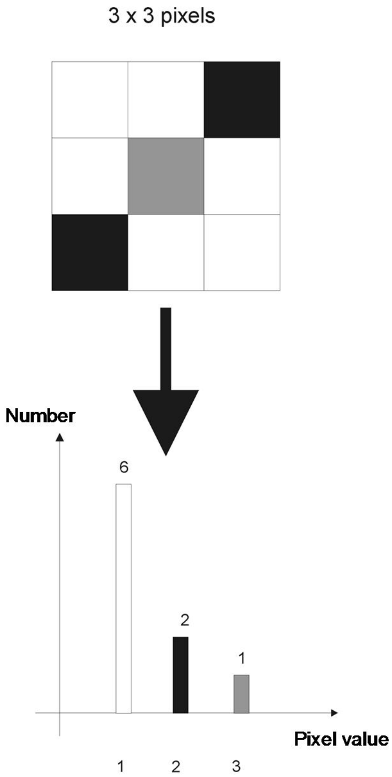
3 Image enhancement

3.1 The histogram

To compute the histogram of an image the pixels are classified into a number of distinct classes according to their value. The histogram shows how many pixels belong to each class. An example: for 8-bit (uint8) greyscale images each grey value is a separate class. The histogram then shows for each grey value how many pixels have that exact value. The picture below clarifies the idea



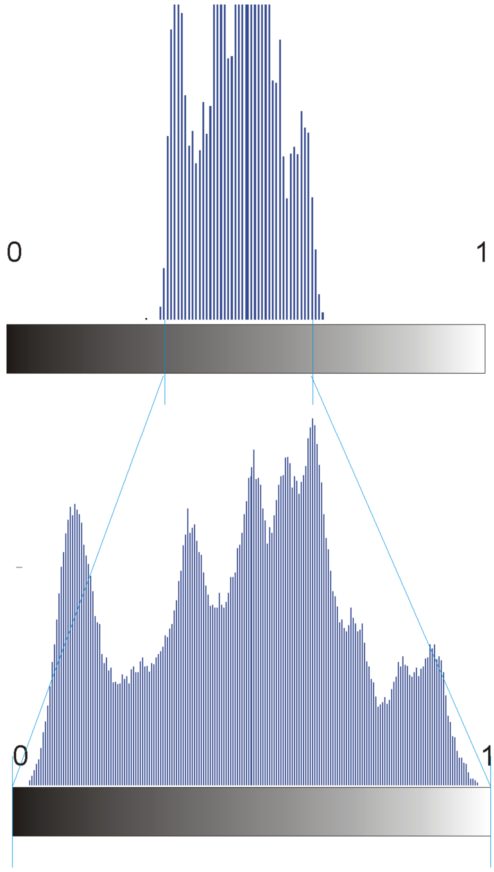
The histogram of a greyscale image is shown on screen with the `imhist(image,bins)` command. `bins` is the number of classes used for classification and `image` is the image from which we want to compute the histogram.

3.1.1 Exercise 6

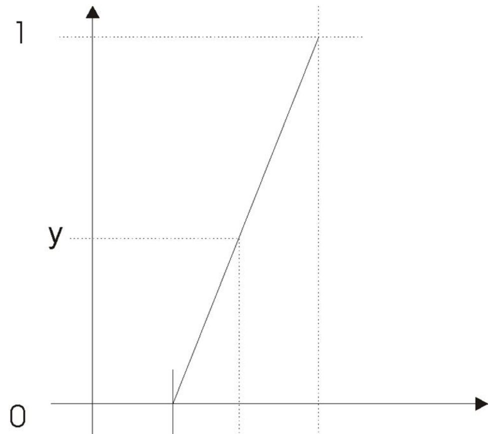
Open the greyscale images *lenad.tif* and *lenal.tif*. Convert both to double format (`im2double`). Use `imshow` to show the images. What do you notice? How does this translate into the histogram? Use `imhist` with 64 bins to view the histograms. Now open the greyscale images *lenagray.tif* en *lenalc.tif* and answer the same questions again.

3.2 Contrast enhancement

We assume greyscale images with intensities of the double type which have values between 0 and 1. We shall discuss two ways of contrast improvement. The first is histogram stretching. The main idea is the following: if we look at the histogram of an image with low contrast we notice (see previous exercise) that the biggest part of the pixel values are situated in a very small interval of intensities. Histogram stretching stretches this small interval to the interval [0,1]. The stretching operation is linear. The following figure clarifies this:



How do the pixel values have to be adjusted to get this effect?

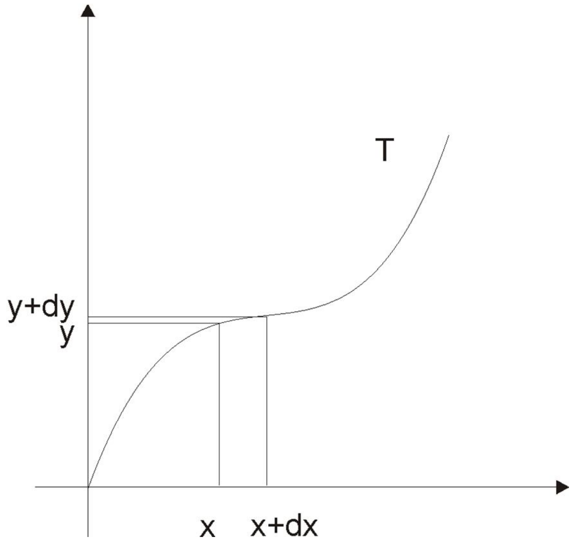


The link between x and y is (straight line through (min,0) and (max,1)):

$$\frac{y-0}{1-0} = \frac{x-\min}{\max-\min}$$

Applying the formula to all pixels delivers the new image.
The second way to resolve the lack of contrast is histogram equalisation. First we notice that if we normalize the values in the histogram by dividing them with the total number of pixels, the histogram approaches the probability density function (pdf) of the intensities. We assume that the image has the ideal contrast when the pdf is uniform. How do we have to transform the pixels to obtain a uniform pdf/histogram? We first make a couple of assumptions to simplify the problem:

X= original pixel value
Y= pixel value after transformation
Y=T(X)
Pixel values (intensities) are continuous quantities.
X lies between 0 and 1
T(X) also lies between 0 and 1
T(X) is a strictly ascending function, so if $X_1 > X_0$, then $T(X_1) > T(X_0)$.
The question is now, what is T() so that the pdf of Y would be uniform?



Look at the figure above. If X lies between *x* and *x+dx*, then Y=T(X) lies between *y* and *y+dy*.
From this follows:
P(y≤Y<y+dY)=P(x≤X<x+dX).
⇔
$p_y(Y)dY=p_x(X)dX$ with p_y and p_x the pdfs of X and Y
⇔
$$p_y(Y) = p_x(X)\cdot\frac{dX}{dY}$$
We know that the pdf of y has to be uniform so $p_y(Y)=cte$ and from $\int_0^1 p_y(y')dy'=1$ follows that $p_y(Y)=1$, so
$$\frac{dY}{dX} = p_x(X)$$
⇔
$$\frac{dT(X)}{dX} = p_x(X)$$
$$\int_0^{x'} \rightarrow T(x') - T(0) = \int_0^{x'} p_x(X)dX$$
Because of the fact that T() is a strictly ascending function, it must be that T(0)=0 so
$$T(x') = \int_0^{x'} p_x(X)dX$$
in other words, T() is the cumulative probability density function of X !

3.2.1 Exercise 7

Histogram stretching. Read in the greyscale image *baboon.tif* and convert to double. Look at the image and the histogram with 64 bins (`imhist`). Now use the functions `min()` `max()` and `imadjust` (with gamma=1) to perform linear histogram stretching. Use the MATLAB help (`help` command) for more information on these commands. Look at the result and its histogram with 64 bins (`imhist`). Compare with the histogram of the original.

3.2.2 Exercise 8

Gamma correction. With linear histogram stretching the transformation function is a straight line given by:

$$y = \frac{x - \min}{\max - \min}$$

With gamma correction the transformation is:

$$y = \left(\frac{x - \min}{\max - \min} \right)^{\gamma}$$

How does the transformation curve look for gamma >1 and gamma<1? What is the effect? Load the greyscale image *lenagray.tif* and convert to double format. Use `imadjust` with 3 and 0.33 for gamma. Look at the images and the histograms (64 bins).

3.2.3 Exercise 9

Histogram equalisation. Load the greyscale image *baboon.tif* again and convert it to double format. Apply histogram equalization with the use of the `histeq` command (64 bins). Look at the result and the histogram (64 bins) of the result. The histogram is not perfectly uniform. What is the reason for this?

3.2.4 Exercise 10

Thresholding. Load the image *medical.tif* and convert it to double format. Create a function called *my_threshold (I, T1, T2)* which takes as input an image and two double values between 0 and 1, and creates an output according to the following rule:

$$out\_value = \begin{cases} 1, if\ pixel\ value\ \epsilon\ [T1, T2] \\ 0, else \end{cases}$$

Threshold the loaded image and create 3 separate scenarios, such that **only** the following are visible: a) air; b) blood and c) bone