Image Processing: Point Operations

- Point operations = method of pixel value modification that size, geometry, and local structure is not change.
- There are many ways to operate, one is *Global* or *Homogeneous*
- Global or Homogeneous operation is the operation that independent with image coordination. For example, adjusting brightness or contrast.

Modifying image intensity

Contrast and Brightness

- Contrast is the difference of color in the image.
- Brightness is the quantity of light in an image. In histogram, if most of data is in right, high brightness. If most of data is in left, low brightness.

Limiting the Results by Clamping

- Clamping is one way to limit range of data by replace over information by highest or lowest of the range, for example, range of 8 bit per pixel is 0 to 255 but one of information is 300, replace it with 255.

Inverting Images

- Inverting images is reverse the order of pixel values and add contrast value to map the result to acceptable range again. Then clamping is not essential for this way.

Threshold Operation

- Thresholding is a way to separate the pixel values into 2 classes and given a constant to be a representative.

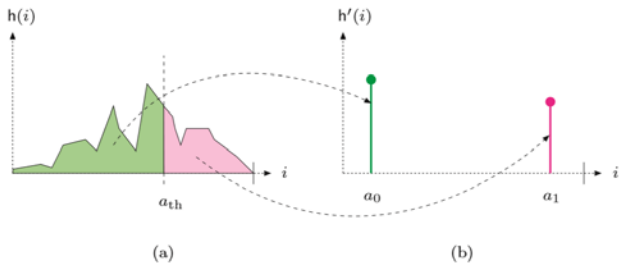
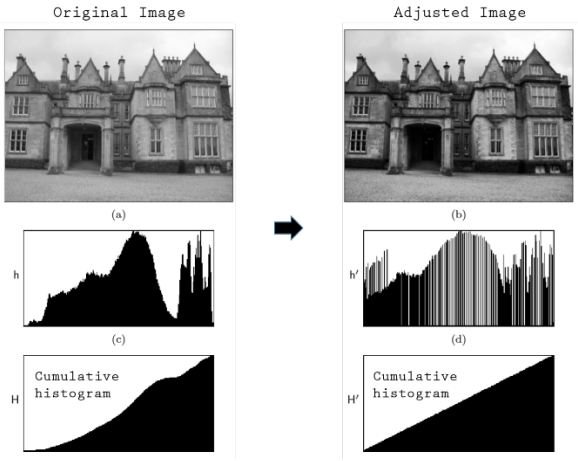


Figure 4.2 Effects of thresholding upon the histogram. The threshold value is a_{th}. The original distribution (a) is split and merged into two isolated entries at a_0 and a_1 in the resulting histogram (b).

Histogram Equalization

- Histogram equalization is method to improve histogram to be a strength line to adjust the contrast of an image.
- How to histogram equalization
  - Find histogram of image *since histogram is a discrete distribution, you can count the number of pixel of each pixel value*
  - Cumulative histogram *by add the amount of previous pixel value with current amount of pixel value*
  - Find the suitable amount of pixel and replace it into old pixel value compute

    *Amount of pixel = (amount of pixel value / size of image) * the highest pixel value*



Histogram Specification

Frequencies and Probabilities

- Frequency , $M*N$ = amount of all pixel
- Probability , $p(i)$ = amount of interested pixel divide by frequency , $p(i) / (M*N)$

  *** sum of all $p(i) = 1$
- PDF; probability density function
- CDF; cumulative density function

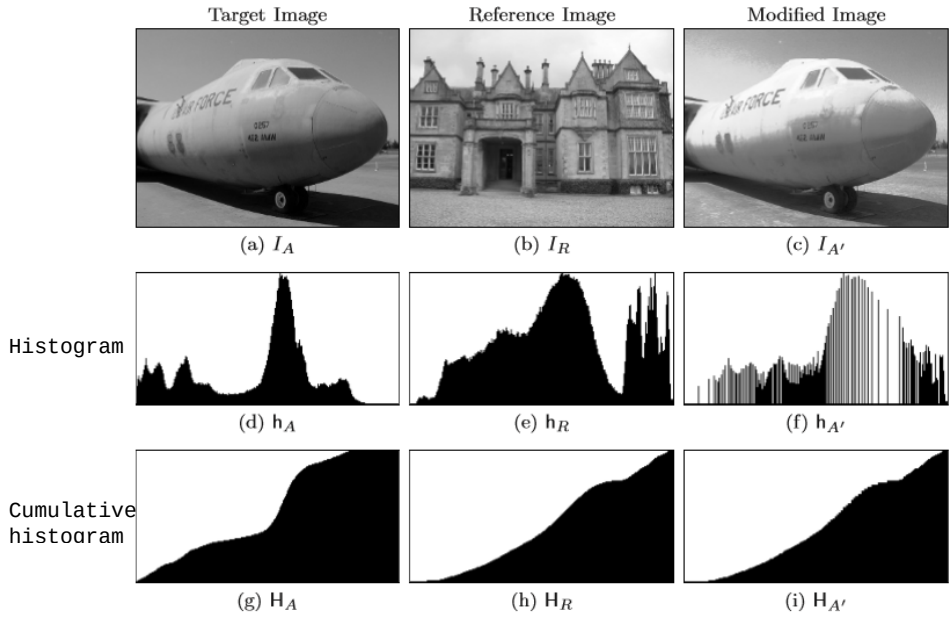
Principle of Histogram Specification

Another name of histogram specification is histogram matching because it is matching the pixel value of an input histogram to reference histogram, reference can be either normal distribution histogram or another image.

How to histogram specification

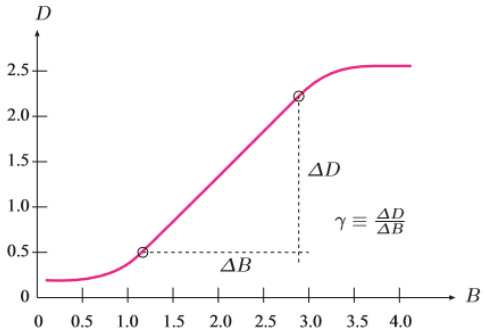
- Get input image and reference image.
- Find histogram of both input image and reference image.
- Compute cumulative of both histograms.
- Matching amount of pixel between input and reference in each pixel value and replace it into input image.

  Find which pixel value of reference has minimum difference of number of pixel from input's pixel value. Then fill it into output image at the same position of an input.



Gamma Correction

- Gamma is relationship between light intensity (B) and film density (D).



- Gamma correction is changing non-linear part into linear, almost found in analog photography and old television (cathode ray tube version).

  For example, *In CRT monitor,* γ *is in range of 1.8 to 2.8*

  In TV analog NTCS, γ *is 2.2*

  In TV PAL system, γ *is 2.8*

- In binary image, output come from power function

  $Output = pixel\ value^{\gamma}$

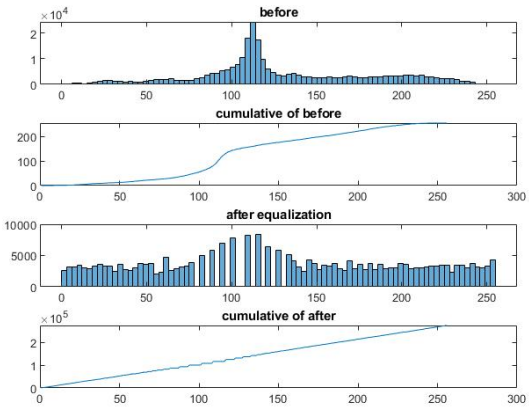
- In general, output come from the below relationship

  $Output = (pixel\ value^{\gamma} \times pixel\ value\ max) \div pixel\ value\ max^{\gamma}$

Reference

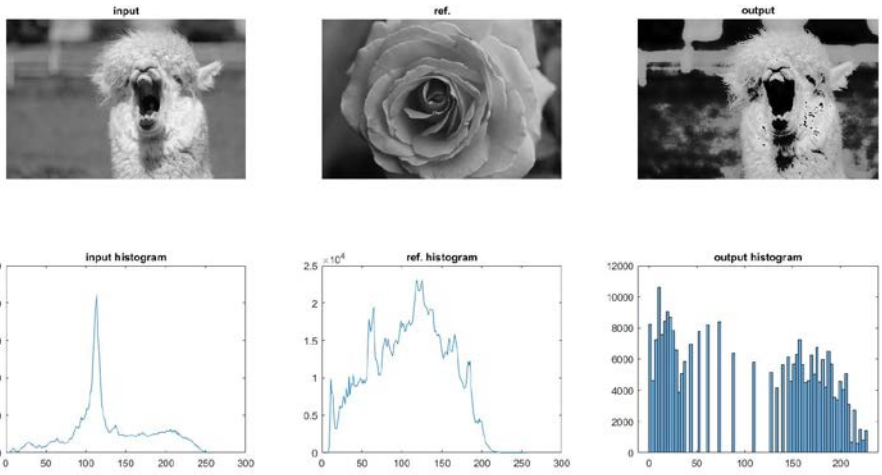
Wilhelm Burger, Mark J. Burge. Principles of Digital Image Processing Fundamental Techniques. Springer. 2009.

Experiment of Histogram Equalization Using MATLAB



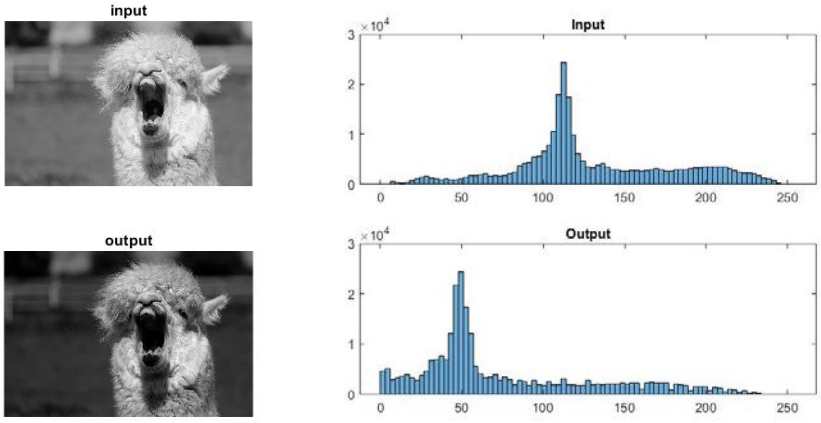

```
clear; clc;
%try by myself
I0 = imread ('alpaca.jpg');
I = rgb2gray (I0);

fre = zeros (256, 1);
prob_freq = zeros (256, 1);
cum_hist = zeros (256, 1);
I2 = uint8 (zeros (size (I, 1), size (I, 2)));
prob_cum = zeros (256, 1);
fre2 = zeros (256, 1);
prob_freq2 = zeros (256, 1);
cum_hist2 = zeros (256, 1);

for i=1: size (I, 1)
    for j=1: size (I, 2)
        number = I (i, j);
        fre (number+1) = fre (number+1) +1;
        prob_freq (number+1) = fre (number+1) / (size (I, 1)* size (I, 2));
    end
end

cum_hist (1) = fre (1);
for i = 2:256
    cum_hist (i) = fre (i) +cum_hist (i-1);
end
for i = 1:256
    prob_cum (i) = round (cum_hist (i) * 255 / (size (I, 1) *size (I, 2)));
end
for i=1: size (I, 1)
    for j = 1: size (I, 2)
        I2 (i, j) = prob_cum (I (i, j) + 1);
    end
end
for i = 1: size (I2, 1)
    for j = 1: size (I2, 2)
        number2 = I2 (i, j);
        fre2 (number2+1) = fre2 (number2 +1) + 1;
        prob_freq2 (number2+1) = fre2 (number2+1) / (size (I2, 1)* size (I2, 2));
    end
end
cum_hist2 (1) = fre2 (1);
for i = 2:256
    cum_hist2 (i) = fre2 (i) + cum_hist2 (i-1);
end
```

Experiment of Histogram Specification Using MATLAB




```
clear; clc;
I0 = imread ('alpaca.jpg'); I = rgb2gray (I0);
I1 = imread ('ref_image.jpg'); Ir = rgb2gray (I1);
%input
for i=1: size (I, 1)
    for j=1: size (I, 2)
        number = I (i, j);
        freq (number+1) = freq (number+1) +1;
        prob_freq (number+1) = freq (number+1) / (size (I, 1)* size (I, 2));
    end
end
cum_hist (1) = freq (1);
for i = 2:256
    cum_hist (i) = freq (i) + cum_hist (i-1);
end
for i = 1:256
    prob_cum (i, 1) = round (cum_hist (i) * 255 / (size (I, 1)*size (I, 2)));
end
%reference histogram
for i=1:size(Ir,1)
    for j=1:size(Ir,2)
        number_ref = Ir (i, j);
        freq_ref (number_ref+1) = freq_ref (number_ref+1) +1;
        prob_freq_ref (number_ref+1) = freq_ref (number_ref +1) / (size (Ir, 1)*size (Ir, 2));
    end
end
cum_hist_ref (1) = freq_ref (1);
for i = 2:256
    cum_hist_ref (i) = freq_ref (i) +cum_hist_ref (i-1);
end
for i = 1:256
    prob_cum_ref (i, 1) = round (cum_hist_ref (i) * 255 / (size (Ir, 1)*size (Ir, 2)));
end
%matching
for i = 1:256
    M = abs (prob_cum (i) - prob_cum_ref (1));
    for j = 1:256
        Mn = abs (prob_cum (i) - prob_cum_ref (j));
        if M < Mn
            C (i, 1) = j-1;
            break;
        elseif Mn == 255
            C (i, 1) = j-1;
        end
        M = Mn;
    end
end
for i = 1:256
    prob_cum_new (i, 1) = prob_cum (C (i, 1));
end
for i = 1: size (I, 1)
    for j = 1: size (I, 2)
        Io (i, j) = prob_cum_new (I (i, j) + 1);
    end
end
Io = uint8(Io);
```

Experiment of Gamma Correction Using MATLAB




```
clear all; clc;
I0 = imread ('alpaca.jpg');
I = rgb2gray (I0);
In = double (I);

gamma =2;
If = In.^gamma;
%power function
for i = 1: size (I, 1)
    for j = 1: size (I, 2)
        Io (i, j) = round ((If (i, j)*255) / (255^gamma));
    end
end
Io = uint8 (Io);
subplot(2, 1, 2); imshow (Io); title ('output')
subplot(2, 1, 1); imshow (I); title ('input')
figure;
subplot(2, 1, 1); histogram (I); title ('Input')
subplot(2, 1, 2); histogram (Io); title ('Output')
```