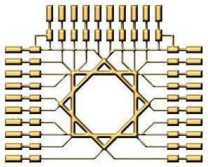
Higher Institute for Applied Sciences and Technology
Telecommunication Department
Multimedia Lab
2019 - 2020



Lab2

Intensity Transformation and Histogram Equalization and Matching

The objective of this experiment is to implement the fundamental intensity transformations and image enhancement techniques using MATLAB Image Processing Toolbox.

Intro to Matlab Image Processing Toolbox

Image Processing Toolbox provides a comprehensive set of reference standard algorithms, functions, and apps for image processing, analysis, visualization, and algorithm development.

- Get information about the image `lighthouse.jpg` using ***imfinfo*** function.
- Read the image `lighthouse.jpg` using ***imread*** function.
- Display the image using ***imshow*** function.
- What is the color type of this image?
- Show the histogram of this image using ***imhist*** function. Does it work? why?
- Use ***rgb2gray*** function to convert the image to grayscale. What is the algorithm used by Matlab for the conversion?
- Show the image histogram. What can you tell from its intensity distribution?

- What is its color type of the image `kids.tif`?
- Read the image `kids.tif` and display it.
- Convert `kids.tif` to a grayscale image using the suitable function.

- Read the image `marbles.tif` and display it.
- Convert the image to indexed image using ***rgb2ind*** function. Repeat the conversion with different number of colors and compare the results to the original.

- Read the image `circle.png` and convert it to grayscale and display it.
- Convert the image to binary using ***im2bw*** function and use the threshold computed by ***graythresh*** function.
- Repeat the previous step using a threshold that only keeps the orange person.

Intensity Transformation Functions

Making changes in the intensity is done through Intensity Transformation Functions. The four main intensity transformation functions are:

- Photographic Negative
- Contrast-stretching Transformation
- Gamma Transformation
- Logarithmic Transformation

1. Negative Transformation

- Read the image `breast.jpg`.
- What is the color type of this image?
- Convert it to a grayscale image and display it.
- Apply Photographic negative using matrix operations.

- Apply Photographic Negative transformation to the image using ***imcomplement*** function.
- Display the image before and after the transformation with the corresponding histograms in one plot using ***subplot***.
- Observe the transformation results. What can this transformation be used for?

2. Contrast-stretching Transformation

Contrast stretching (often called normalization) is a simple image enhancement technique that attempts to improve the contrast in an image by `stretching' the range of intensity values it contains to span a desired range of values.

- Read the image `house.jpg`.
- Display the histogram for this image. What do you note?
- Apply contrast stretching on this image using ***imadjust*** function. (You can use ***stretchlim*** function to get lower and upper limits that can be used for contrast stretching).
- Display the gray image before and after the transformation with the corresponding histograms in one plot.
- Observe the results. What is the effect of this transformation?

3. Gamma Transformation

- Read the image `raider.png` and display it.
- Convert the image to gray.
- Apply gamma transformation to this image using ***imadjust*** function for different values of gamma.

- Display the gray image and two other images with different gamma values and the corresponding histograms in one plot.

- What is the effect of gamma transformation?

4. Logarithmic Transformation or Dynamic Range Compression

Dynamic range in photography describes the ratio between the maximum and minimum measurable light intensities.

The dynamic range of an image can be compressed by replacing each pixel value with its logarithm. This has the effect that low intensity pixel values are enhanced. Applying a pixel logarithm operator to an image can be useful in applications where the dynamic range is too large to be displayed on a screen.

The most common application for the dynamic range compression is for the display of the Fourier Transform.

The logarithmic transformation of an image x is: $y = c*\log(1 + \text{double}(x))$

The constant ***c*** is usually used to scale the range of the log function to match the input domain. In this case `c=255/log(1+255)` for a uint8 image, or `c=1/log(1+1)` for a double image.

- Read the image `text.png` and display it.

- Apply Fourier transform using ***fft2*** function and display the magnitude of the result. Explain the results.

- Apply the logarithmic transform on Fourier magnitude matrix and convert the result to an intensity image using ***mat2gray*** function and display the result.

- What information about the image can you tell from the result?

- What other effect log transformation has on images?

Histograms

1. Histogram Equalization

- Read the image `flowers.jpg` and display it with the corresponding histogram. What do you note?
- Use ***histeq*** function to enhance the image.
- Display the image before and after transformation with the corresponding histograms in one plot.
- What is the effect of histogram equalization?

2. Histogram Matching

- Read the images `ribs_a.jpg` and `ribs_b.jpg` and display them.
- Enhance the image `ribs_b.jpg` by matching its histogram to the histogram of `ribs_a.jpg`.(use ***imhist*** and ***histeq*** functions.)
- Compare the histogram of the matched image with the histogram of the reference one. What is the effect of this transformation?