# Department of Computing

# Name: Ali Hassaan Mughal Reg: 173627

**EE433: Digital Image Processing**

# Lab 3 : Basic Image Processing 2

**CLO1: Acquire the fundamental concepts of a digital image processing system**

**CLO2: Identify and exploit analogies between the mathematical tools used for 1D and 2D image analysis and processing**

# 

# 

# Lab 3: Basic Image Processing 2

**Introduction**

This lab is an introduction to basic image processing.

**Objectives**

This lab will provide the concepts of image processing to the students by doing various tasks.

**Lab Tasks**

**Task #1: Image Binarization using a predefined global threshold.**

Take an RGB image (preferably from the provided ones) and convert it to binarized form (in 0/1 form) by defining a single global threshold. Repeat the experiment with the three provided images and identify why a single global binarization threshold may not be applicable in a wide variety of application scenarios.

from PIL import Image

img1 = Image.open("images\\B1.png").convert("L")

img2 = Image.open("images\\B2.jpg").convert("L")

img3 = Image.open("images\\B3.jpg").convert("L")

images = [img1,img2,img3];

for imgg in images:

width,height = imgg.size

pixels = imgg.load()

for x in range(0,width):

for y in range(0,height):

if pixels[x,y] < 128:

pixels[x,y] = 0;

else:

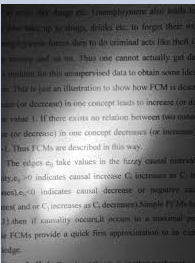
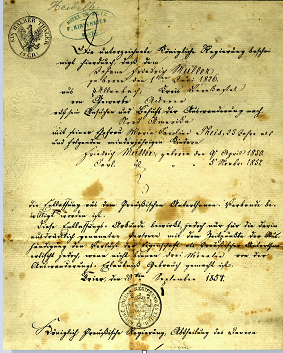
pixels[x,y] = 255;

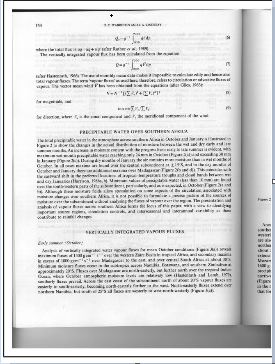
imgg.show()

**Analysis and Solution:**

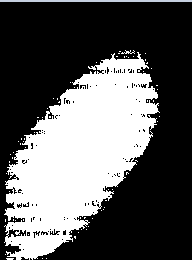
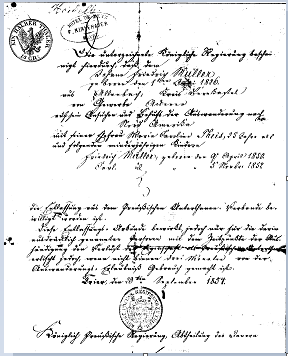
This technique is not very good for a fixed value for all images, as it is not necessary that all the images are uniformly between 0,255 making the middle range 128 a good point for binarization. It is possible that an image has all the pixels under 128 but still having different range, so for that we need to set a new threshold. It can be done by making a frequency graph or some analytical techniques differently for different images.

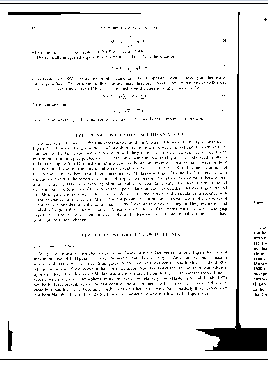
**Original Images**



**Binarized Images/Output Images**



**Task #2: Create Intensity Histogram from a Greyscale image**

**Code:**

**from PIL import Image**

**import matplotlib.pyplot as plt**

**img1 = Image.open("images\\B1.png").convert("L")**

**width,height = img1.size**

**intensities = [0 for y in range(0,256)]**

**print(intensities)**

**for x in range(0,width):**

**for y in range(0,height):**

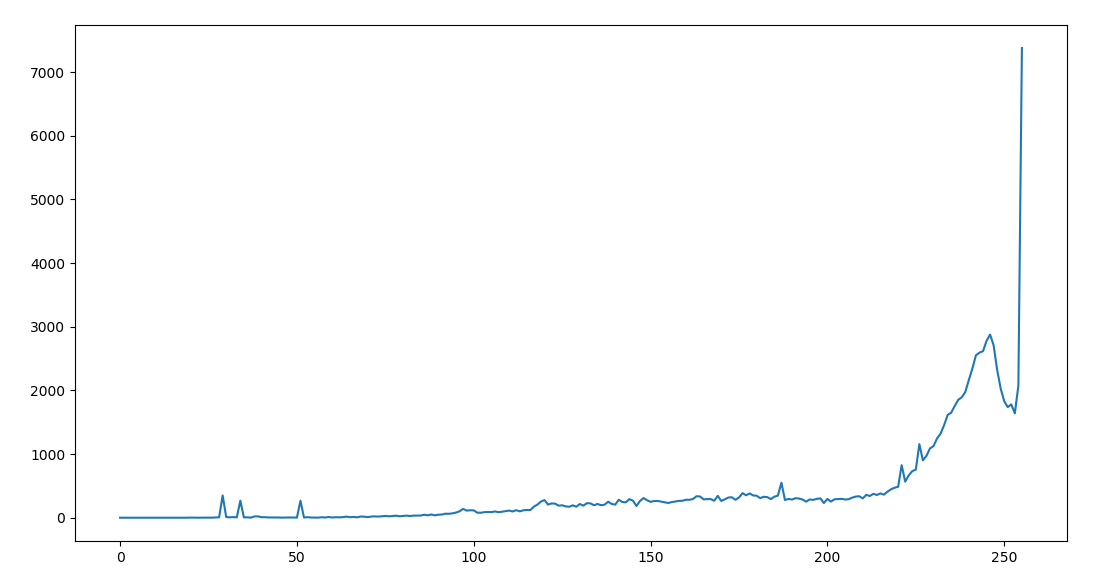
**intensities[img1.getpixel((x,y))]+=1;**

**print(intensities)**

**plt.plot(intensities)**

**plt.show()**

**Output:**



**Task #3: Recursive XY-cut algorithm**

**Code:**

**from PIL import Image,ImageDraw**

**import matplotlib.pyplot as plt**

**img1 = Image.open("images\\XY-cutss.png").convert("L")**

**width,height = img1.size**

**intensities = [0 for y in range(0,height)]**

**for x in range(0,width):**

**for y in range(0,height):**

**if (img1.getpixel((x,y)) < 128):**

**intensities[y]+=1;**

**threshold = 30;**

**draw = ImageDraw.Draw(img1)**

**for y in range(0,height):**

**if intensities[y] < 20:**

**draw.line((0,y) + (width-1,y) , fill=0)**

**img1.show()**

**Output:**



Hint:

Hand in the source code from this lab at the appropriate location on the blackboard system at LMS. You should hand in a single compressed/archived file named Lab\_1\_<your reg. No. ABC without angle brackets>.zip that contains the following.

1. All resulting image files representing the work accomplished for this lab.
2. A plain text file named OUTPUT.txt that includes a) author information at the beginning, b) a brief explanation of the lab, c) any comments, or suggestions, d) your response to all the tasks.

To Receive Credit

1. By showing up on time for lab, working on the lab solution, and staying to the end of the class period, only then you can receive full credit for the lab assignment.

Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.

**Deliverable**

Hand in the source code from this lab at the appropriate location on the blackboard system at LMS.