**IMAGE PROCESSING**

Deepthi Thalagundamatada

CSC 532- Design and Analysis of Algorithm

Dr. Gene Tagliarini

**ABSTRACT**

Image processing Assignment renders us to explore the concepts of image processing and understand the image processing concepts with the following Questions

Use a convenient context such as either the [Algoritharium](http://people.uncw.edu/tompkinsj/algoritharium/index.html) (the [updated jar file is here](http://people.uncw.edu/tagliarinig/courses/532/Lectures/algoritharium.jar)), the [Picture](http://people.uncw.edu/tagliarinig/courses/532/Lectures/Picture.java) class, or the necessary Python resources (DO NOT simply apply the library functions in PIL) to implement and demonstrate algorithms for the following questions mentioned.

**BACKGROUND**

(1)As truly said “Picture is worth a Thousand Words”, An image is a picture representing visual information. A digital image is an image that can be stored in digital form. Digital image processing is a field of study that seeks to analyze, process, or enhance a digital image to achieve some desired outcome.

Image processing basically includes the following three steps:

* Importing the image via image acquisition tools;
* Analyzing and manipulating the image;
* Output in which result can be altered image or report that is based on image analysis.

The detailed steps of image processing can be defined as below:



**INTRODUCTION**

(2) An image is a two-dimensional signal. The image can be defined by mathematical function f(x,y) where x and y are the two co-ordinates horizontally and vertically. The value of f(x,y) at any point is given the pixel value at that point of an image.

RGB colors: RED, GREEN, BLUE in each pixel have an 8-bit value which

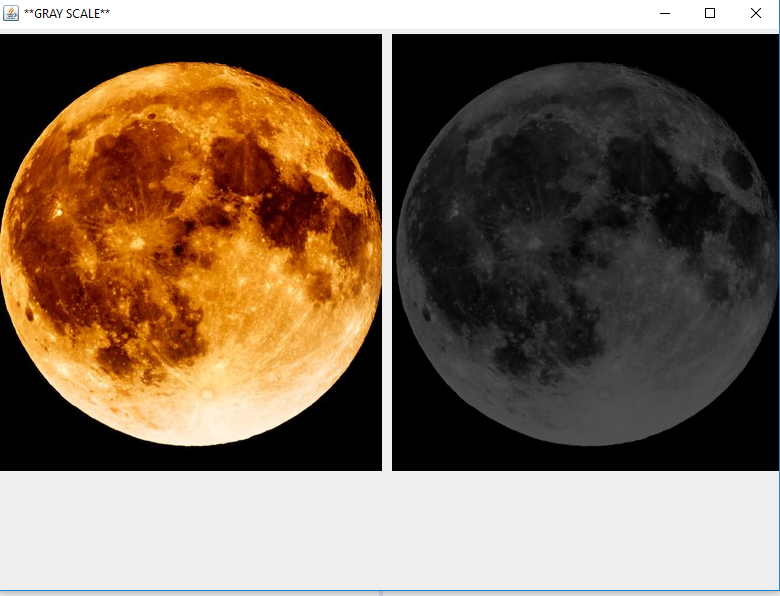
means RED-255,GREEN-255,BLUE-255

For Digital Image processing project implementation has several steps according to the need of the question which has been given. The main common operations in the image processing project for each question include certain steps which are common:

1. Set the frame
2. Obtain the image
3. Set the transformed image in the frame

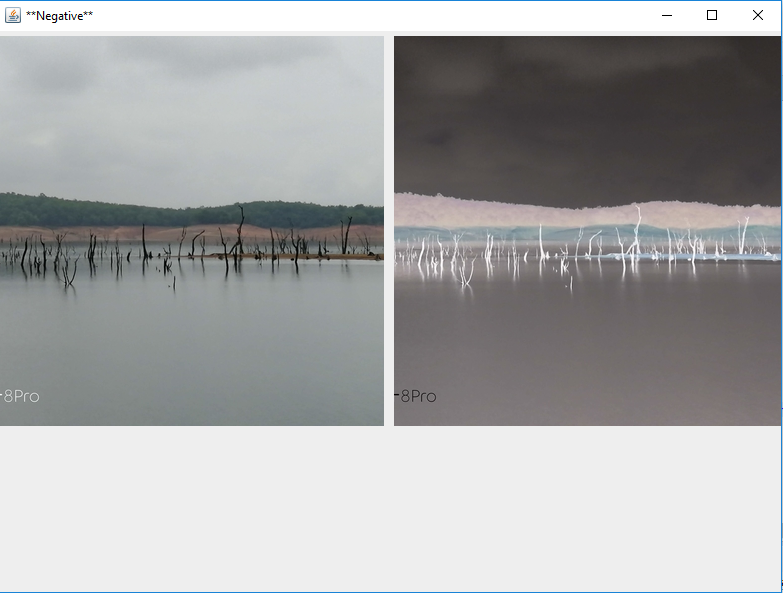
**Experimental Design**

1. Color-to-gray-scale conversion. Is this an invertible operation?

* Set the Average of Red, Green, Blue gives grayscale: R+G+B/3 to each pixel in the Image.
* 

1. **Convert an image to its negative. Is this an invertible operation?**

* An image can be converted to negative by
  + 255-Red,255-Green,255-Blue
  + No, it is not an invertible operation



1. **Find and “enhance” boundary contours.**

Detect the edges inside the image

Initialize the colors as leftColor, Rightcolor and Bottomcolor to null Step 3: check the values of rows and columns from image pixels for the length,

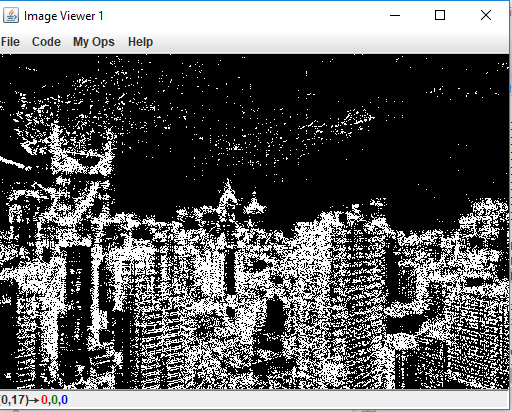
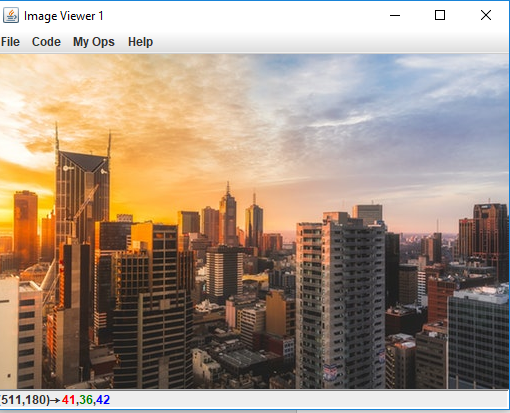
**if row values is less than length of images**

**then execute as Bottomcolor= c[row+1]+col]**

**if column values is less than length then do rightcolor=c[row]+[col+1]**

**elseif execute as leftcolor=c[row]+[col] or**

**elseif**

**Execute the color what is given by users either black or white**

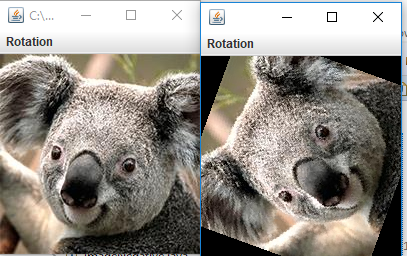
1. Rotating parts of an image.

**Get the coordinates of the current image**

**use formula for new coordinates as**

**x=acos(angle)+bsin(angle)+x0**

**y=asin(angle)+bcos (angle)+y0**



1. **Warping parts of an image.**

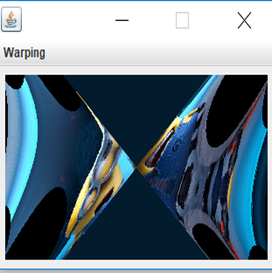
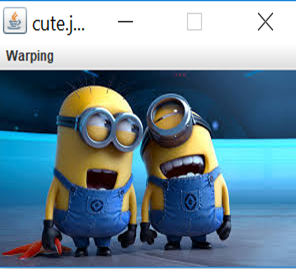
**for row and column to warp the image with angle**

**coordinates and use formula for radius as square root of(dx^2-dy^2)**

**Set new coordinates as**

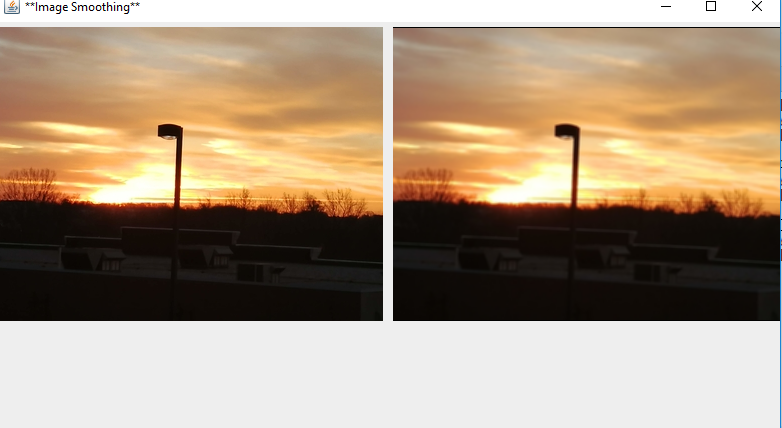
**x=cos(angle)dx + sin(angle)dy + constant\*x**

**y=sin(angle)dx+cos (angle)dy+constant\*y**



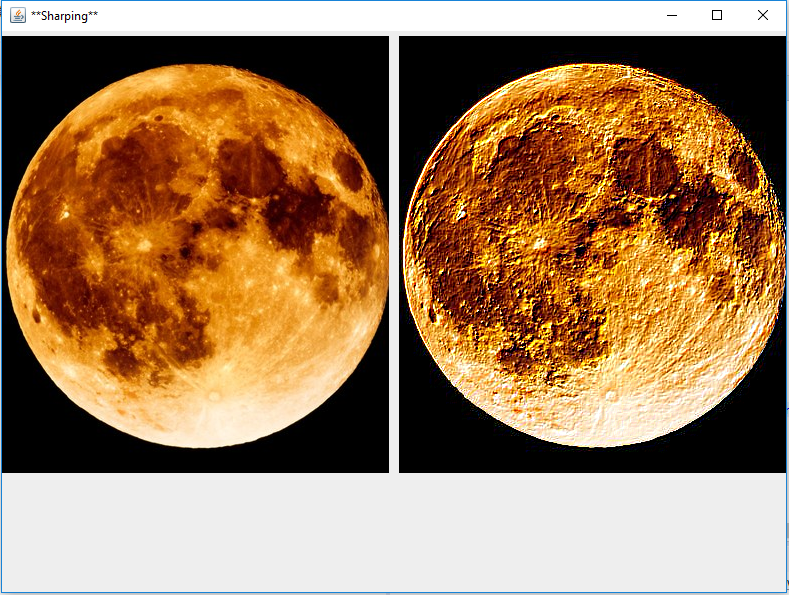
6.  Implement image filters for smoothing and sharpening, as well as removing speckle

* **Smoothing Operation(3)**
  + Get the image and Pass filter [3\*3 linear filter] to each pixel of the image
  + Smooth of image happens by the mean of each pixel ( red/9,green/9,blue/9)



**Sharping operation(3)**

* + **The 3\*3 linear filter is passed to each pixel where**
  + **if the RGB values if more than 255(Max Threshold) they are set to 255**
  + **if less than 0(min threshold) are they are set to 0.**



7.  Contrast enhancement (Is this an invertible operation?) and compare the results obtained via the histogram equalization method using:

        RGB space

Find histograms in the red, green, and blue color spaces and use the histograms to remap the RGB levels. Use the remapped values to assign pixel values in a contrast-enhanced image. How would you describe the qualitative differences between the original image and the contrast-enhanced image?

(1)

**Step 1. create an array for HistRed, HistGreen, HistBlue of size 256 and initialize it with 0.  
Step 2: Create the histogram of the image by scanning each pixel of the image and then by incrementing related array member.  
            HistRed[Red\_Value(pixel)] = Hist[Red\_Value(pixel)]+1**

**HistGreen[Green \_Value(pixel)] = Hist [Green\_Value(pixel)]+1**

**HistGreen [Blue\_Value(pixel)] = Hist [Blue\_Value (pixel)]+1  
Step 3: Form a cumulative histogram CHR,chG,chB of size 256  
      chRed=0, chGreen=0 and chBlue=0**

**For i from 0 to 255**

**chRed[i] = chRed [i-1] + chRed [i]**

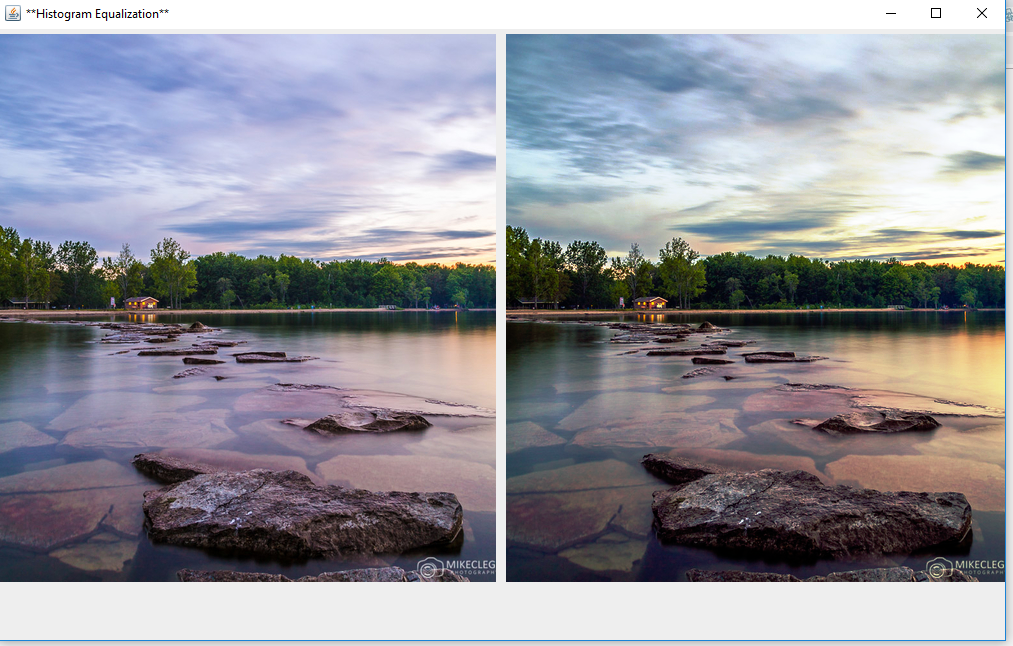
**chGreen[i] = chGreen [i-1] + chGreen [i]**

**chBlue[i] = chBlue [i-1] + chBlue [i]  
Step 4: Round((255\*chRed[i])/(Total Pixel))**

**Round((255\*chGreend[i])/(Total Pixel))**

**Round((255\*chBlue[i])/(Total Pixel))**

**Step 4: Rescan image and create a new image with new Red, Green and Blue values**



        HSB space

Transform the RGB values into HSV values (Hue, Saturation, and Brightness) to enhance an image using histogram equalizations over seven subsets HSB values (H, S, or B only, H and S, H and B, S and B, and HSB).

For RGB to HSB conversion with H,S,B as different values. The picture depicts where subsets are taken and are experimented. Making HSB histogram equalization is still pending.

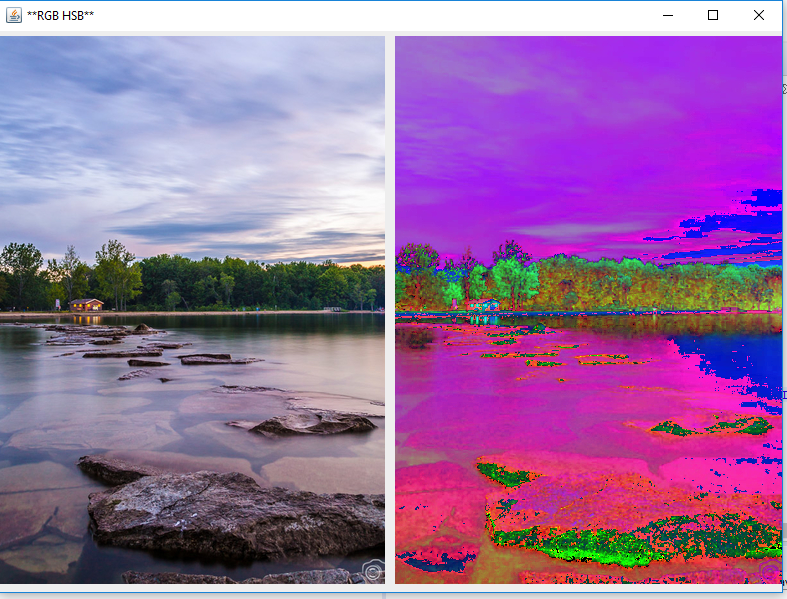


Image for HB:



Image for HS:

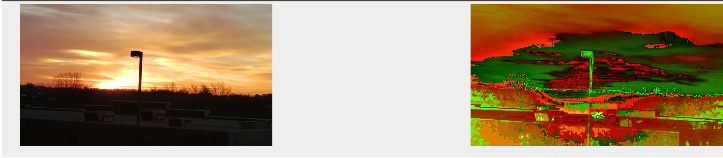


Image for HSB:

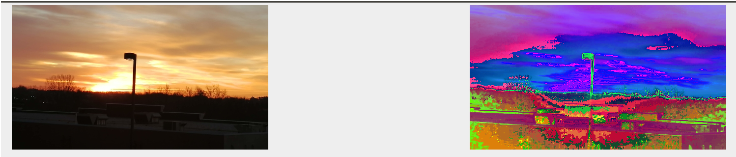


Image for HS:



Image for S:



Image for SB

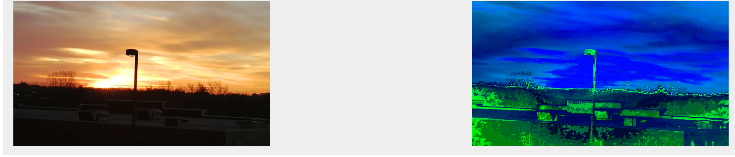


Image for B

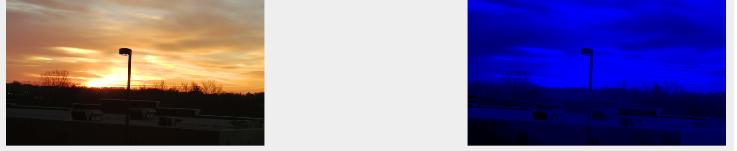
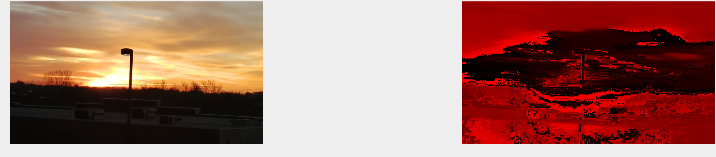


Image for H:



10. Fading:  Create and demonstrate the team’s implementation of algorithms to fade from one image into another. For example, team member portraits could be used to form a fade sequence from one to another.

**Step1: Give one image as input**

**Step 2: pass another image pixels to the current image**

**Step 3: Display the image**

**FINDINGS**

Image Flipping: from the mid of the width

Get x,y of the pixel with the color of the image

Flip the x and y value (width - x - 1, y)



**FUTURE WORK**

Future work will be mainly on the Compositing and identify a region in one image and search for a similar region in other images.

**CONCLUSION**

Image Processing quite interesting subject where learning of the pixels and color on each pixel play a very important role. Learning of Digital image processing has given more insight into the images and their properties.

**ACKNOWLEDGEMENT**

I would like to acknowledge Ganga Poudel in this Project.

**REFERENCE**

(1) [**http://charity.cs.uwlax.edu/artofimageprocessing/ArtOfImageProcessingTrailer.pdf**](http://charity.cs.uwlax.edu/artofimageprocessing/ArtOfImageProcessingTrailer.pdf)

(2)http://web.ipac.caltech.edu/staff/fmasci/home/astro\_refs/Digital\_Image\_Processing\_2ndEd.pdf

(3) http://homepages.inf.ed.ac.uk/rbf/BOOKS/VERNON/Chap004.pdf