Assignment-1
Basic Image Editor

GOVARDHAN REDDY(213070008)

August 2021

1 Abstract

Image processing involves enhancing the pictorial information of the original image which will be very helpful for human interpretation. There are several such image processing techniques available. The main aim of the paper is to construct a simple *GUI based Basic Image Editor*. It should be able to load images and apply certain image processing techniques on them so that the hidden information becomes more clear. Also it has to perform certain additional functionalities like loading the image onto GUI etc. The observation of the images after applying image processing techniques has been studied in the paper. It has been observed that the GUI based Basic Image Editor has been able to perform these image processing techniques and functionalities very effectively.

2 Introduction

When we capture a photo with a camera, there is some information within the image which will be hidden from the human eye. So by applying certain Image processing techniques, we can enhance the image so that the hidden information becomes visible to the human eye.

The main objective of the paper is to build a Basic Image Editor that performs the some image processing techniques and functionalities.After applying the techniques we can observe the hidden information in the images. We have used tkinter from python for implementing GUI. There are certain important things to be taken care of.

Handling both the colour and gray images for applying image processing techniques is important. The image processing techniques implemented should be well coded to see the results. After applying the techniques we have to handle the range of intensities in the resultant images. Also it is equally important that both the front-end and back-end are in functioning in synchronization.

In this paper we will see image processing techniques are implemented along with implementation of GUI and finally we will see the results and conclude,

3 GUI design

3.1 Overall approach

GUI is implemented by using tkinter in python. There are two main things to be taken care here. One is front-end and the other is back-end.

When we press the buttons in the front-end it will perform the actions back-end to which the user don't have any access.

The front-end is implemented using tkinter in python in the paper and the back-end through certain functions in python. Also as mentioned the users can only access the front-end and they don't have any idea about what is going on back-end. Hence it is important that they both are performing well simultaneously.

The front-end displays the buttons, entries, labels and images. These labels, buttons, entries are visible to the user. The functionalities and the techniques implemented by GUI are mentioned below.

3.2 GUI features:

It can perform the following image processing techniques: These image processing techniques are well discussed in the Image processing operations.

1. *Gamma correction*
2. *Logarithmic transformation*
3. *Global Histogram Equalization*
4. *Local Histogram Equalization*
5. *Sharpening*
6. *Blurring*
7. *Negative of Image*

It has the following functionalities:

1. *Loading an image*

2. *Displaying Image Area —item undo*
3. *undo all*
4. *Saving an image*
5. *Exit*

As the names themselves say, when we press the above buttons, they respectively perform their actions like Loading the image onto GUI, Displaying the area of image in pixels, Undo upto two images, Undo all to get to the original image, saving an image in the system and exiting the GUI.

4 Image Processing Operations

Transformation is denoted by

$$s = T(r)$$

where r is related to the pixel intensities in original image
T is transformation acting on the original image and
s is the related to the resultant pixel intensities.

4.1 Gamma correction

4.1.1 Purpose

Gamma correction is also called power-law transformation. Gamma correction can control the overall brightness of an image. Here we have a parameter called $\gamma>0$ which controls the over all brightness of image.

4.1.2 Mathematical formula

If r is the original image intensity (let r takes intensity values between 0 and L-1, $\forall$ L$\in\mathbf{N}$), then the transformed intensity, s(s also takes intensity values between 0 and L-1, $\forall$ L$\in\mathbf{N}$) is given by(c=constant)

$$s = cr^{\gamma}$$
$$s = (L-1)(\frac{r}{L-1})^{\gamma}$$

where $\frac{r}{L-1}$ takes values between 0 and 1 and hence
s=(L-1)$(\frac{r}{L-1})^{\gamma}$ takes values between 0 and L-1.
Hence c=(L-1)$^{1-\gamma}$

4.1.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.2 Logarithmic transformation

4.2.1 Purpose

Logarithmic transformation is an image processing technique used to expand the dark pixels of the images as compared to higher pixels thus enhancing the image.

4.2.2 Mathematical formula

If r is the original image intensity (let r takes intensity values between 0 and L-1, $\forall$ L$\in$**N**), then the transformed intensity, s(s also takes intensity values between 0 and L-1, $\forall$ L$\in$**N**) is given by(c=constant)

$$s = c\log_{10}(1 + r)$$

where c = $\frac{L-1}{\log_{10}(1+max(r))}$ where max(r) is the maximum intensity value in the original image.

4.2.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.3 Global Histogram equalisation

4.3.1 Purpose

Global histogram equalisation is an image processing technique used to improve the contrast in the original image. It accomplishes this by effectively spreading out the most frequent intensity values, i.e. stretching out the intensity range of the image

4.3.2 Mathematical formula

Let the range of intensities in the image be 0 to L-1, where L-1 is maximum intensity value in the image. Let r_k be the frequency of intensity values k in the original image for $k \in \{0, \ldots, L-1\}$. Let MN be the total number of pixels in the original image. Let s_k be the mapped intensity level of the corresponding intensity level k in the original image where $k \in \{0, \ldots, L-1\}$. The discrete version of equalising transformation is given by:

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \tag{1}$$

$$= \frac{L-1}{MN}\sum_{j=0}^{k} n_j \quad \text{for k} \in \{0, \ldots, L-1\} \tag{2}$$

where n_j is the number of pixels with intensity level j
$p_r(r_k)$ is probability mass function of r_k

4.3.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.4 Local Histogram equalisation

4.4.1 Purpose

Histogram equalization is an image processing technique widely used image contrast enhancement method. While global histogram equalization enhances the contrast of the whole image, local histogram equalization can enhance many image details by considering different sub image matrices and applying equalization on them. This is achieved by taking a sliding window.

4.4.2 Mathematical formula

Let W be a sliding window of size 3x3 with all 1's in it. Now we will slide the window through the image and we will get 3x3 sub image matrices. For every such sub image matrix we will apply the same formulas (1) and (2) with r_k, where $k \in \{0, \ldots, L-1\}$, now equals the frequency of intensity level k in the sub image matrix instead of the whole matrix and replace the intensity of only the central pixel in original image with the mapped intensity s_k, where $k \in \{0, \ldots, L-1\}$, from the above formulas. We will add $\boldsymbol{\alpha}$ percent of original image intensity and $\boldsymbol{\beta = 1 - \alpha}$ percent of the locally equalized image intensity to get the resultant image intensity $\boldsymbol{o_k}$.

$$\begin{aligned}\boldsymbol{s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)} \\ \boldsymbol{= \frac{L-1}{MN}\sum_{j=0}^{k} n_j} \quad \text{for k} \in \boldsymbol{\{0,\dots,L-1\}} \\ \boldsymbol{o_k = \alpha(centralpixelintensity) + \beta(s_k(1,1))}\end{aligned} \tag{3}$$

where $\boldsymbol{n_j}$ is the number of pixels with intensity level j in the sub image matrix of size 3x3
$\boldsymbol{s_k}$(1,1)= modified intensity of central pixel value
$\boldsymbol{p_r}(\boldsymbol{r_k})$ is probability mass function of $\boldsymbol{r_k}$
$\boldsymbol{\alpha = 0.8}$ and $\boldsymbol{\beta}$=0.2 are considered for this paper.

4.4.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.5 Sharpening

4.5.1 Purpose

Sharpening is an image processing technique which enhances edges and fine details in an image. It is widely used in different industries for increasing the local contrast and sharpening the images. Sharpening is a kind of high pass filter which preserves edges and fine details by removing lower frequency components.

4.5.2 Mathematical formula

The method used for sharpening the images is unsharp masking. First the image is smoothened with the help of gaussian filter. Then mask is obtained by subtracting the smoothened image from original image. Now we will add certain portion of the mask to the original image which provides sharpness to the image.

$$\begin{aligned}
w(x,y) &= \exp(-\frac{(x^2+y^2)}{2\sigma^2}) \\
\bar{f}(x,y) &= \frac{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)f(x+s,y+t)}{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)} \\
g_{mask}(x,y) &= f(x,y) - \bar{f}(x,y) \\
g(x,y) &= f(x,y) + \beta(g_{mask}) \\
\bar{g}(x,y) &= \frac{(g(x,y) - min(g)) * (L-1)}{max(g) - min(g)}
\end{aligned}$$

where w(x,y) for this paper is a 4x4 gaussian filter
$\bar{f}(x,y)$ is the correlated value at the central pixel
f(x,y) is the original intensity level at the central pixel
g_{mask}(x,y) = masked intensity level at the central pixel
g(x,y) is resultant intensity value at the centre and
$\bar{g}(x,y)$ is to bring the central pixel value into the range 0 to L-1
min(g) and max(g) are minimum and maximum intenstiy values in the entire resultant image.

4.5.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.6 Blurring

4.6.1 Purpose

Blurring, also known as smoothing is mainly used for removing outlier pixels in the original image. Blurring is a kind of low pass filter which smoothens the image by removing higher frequency components like edges.

4.6.2 Mathematical formula

The filter used for blurring is box filter or weighted average filter. The mathematical formula for obtaining a single pixel in the output image is

$$g(x,y) = \frac{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)f(x+s,y+t)}{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)}$$

where f(x,y) is original image with x,y as spatial co-ordinates and
w(x,y) is a box filter with all 1's in it divided by the total sum of the filter coefficients.
The above formula is extended for all the pixels in the input image.

4.6.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

4.7 Negative of an image

4.7.1 Purpose

Some cameras encode the image data in negatives. Negative images are generally used to make positive prints on photographic paper. Positive prints are made by projecting the negative on the paper with a photographic enlarger or making contact print.

4.7.2 Mathematical formula

The transformation function used in image negative is:

$$\boldsymbol{s = (L - 1) - r}$$

where L-1 is the maximum intensity value
s is the intensity value at the output pixel and
r is the intensity value at the input pixel

4.7.3 Citation

Don't know the exact citaion of earlier use in image processing. These mathematical formulas are taken from he book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**

5 Experiments and results

5.1 GUI Screenshot

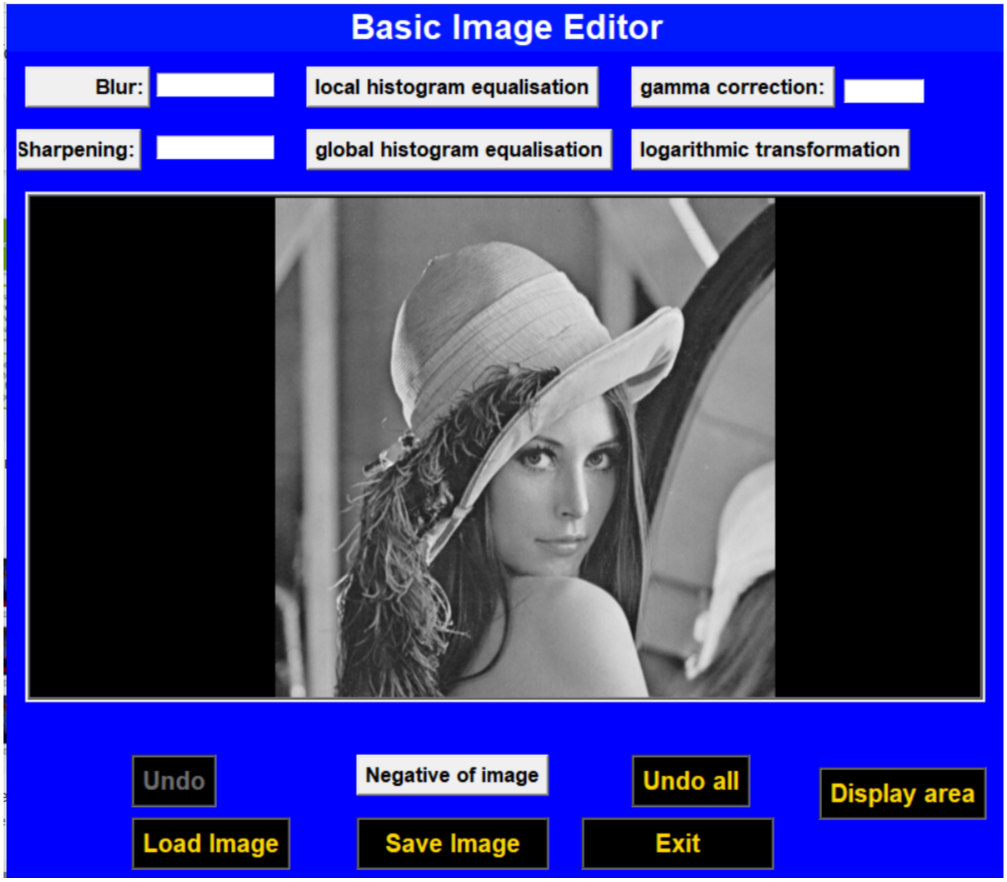


Figure 1: Screenshot of the tkinter GUI.

5.2 Global Histogram Equalisation Results

Here we can see that original figure has higher frequency of higher intensities. Hence if we perform global histogram equalisation, the frequency of intensities get distributed equally.

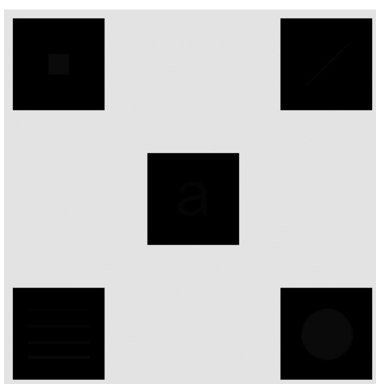
(a) Original Image

(b) Global Histogram equalized image

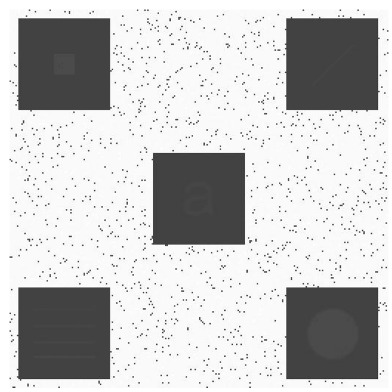
Figure 2: Global Histogram results

5.3 Local Histogram equalisation results

If we enhance the below original image with Global histogram equalization we might not get good results as shown below. But if we apply Local histogram equalisation it performs better than the Global Histogram equalisation.

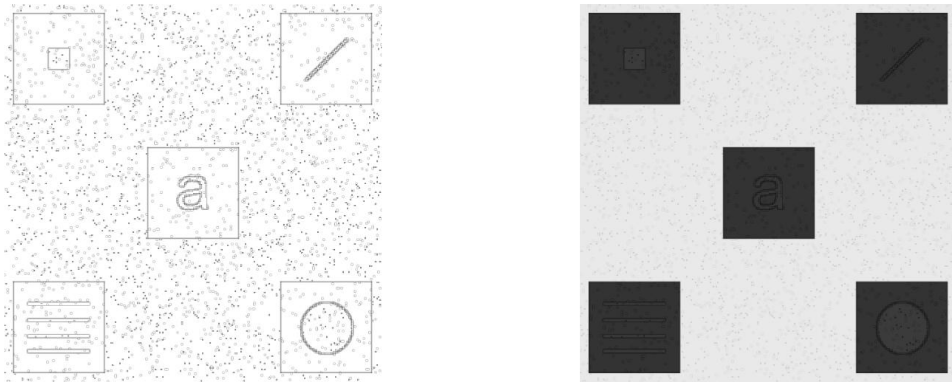


(a) Original Image



(b) Global Histogram equalized image

Figure 3: Global Histogram result



(a) Image when $\boldsymbol{\alpha}$=0 and $\boldsymbol{\beta}$=1

(b) Image when $\boldsymbol{\alpha}$=0.8 and $\boldsymbol{\beta}$=0.2

Figure 4: Local Histogram results

5.4 Gamma Correction results

As discussed gamma can control the brightness of image. If the image is darker we can enhance it using lower gamma to see the finer details in the image. We will see two cases: a) When the image is darker, lower gamma gives us a good view of images b) When the image is brighter, higher gamma gives us good view of image.

(a) Original Image

(b) Image when $\boldsymbol{\gamma}$=0.6

Figure 5: Gamma Correction results when $\boldsymbol{\gamma}$<1

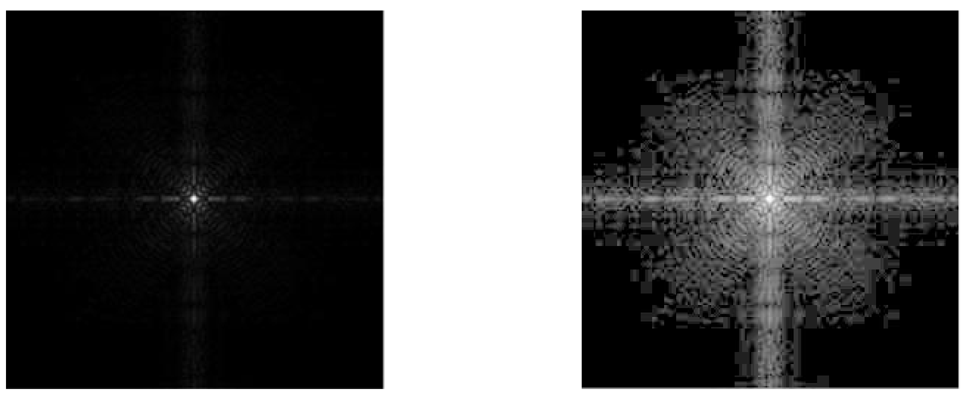
(a) Original Image

(b) Image when γ=4

Figure 6: Gamma Correction results when γ>1

5.5 Logarithmic transformation results

Logarithmic transformation is used to enhance the darker details of the image. So in the below original image we can't see the information with much clarity, but when we apply logarithmic transformation, we get better results.



(a) Original Image

(b) Image after log transformation

Figure 7: Logarithmic transformation

5.6 Negative of Image results

Negation is just the inverted version of original image. It was used to capture images from the camera in olden days and then converted into original. Following figure shows the negative of original image.

(a) Original Image (b) Negative of original

Figure 8: Negative of an image

5.7 Blurring results

Blurring is used to smoothen the images. Any image with edges can be smoothened. So here is an example of blurring an image.

(a) Original Image (b) Blur with box filter of size 5

Figure 9: Blurring of an image

5.8 Sharpening results

Sharpening is used to enhance edges. In the below we show that how unmasking looks like and what happens if mask is added to original.

Figure 10: Original Image

(a) Gaussian mask

(b) Sharpening with gaussian filter of sigma=5

Figure 11: Sharpening an image

5.9 Composition of global histogram equalisation and negative of image techniques

We have seen in Figure 2 that Global Histogram equalisation alone can't reveal the details of image. Hence we will use negation of original first and then apply Global

Histogram Equalisation. Like this we can use any combination of mentioned image processing techniques to get the desired output.

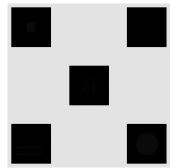
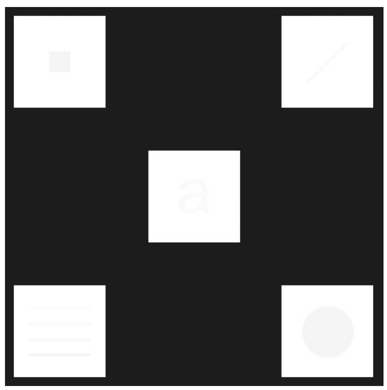
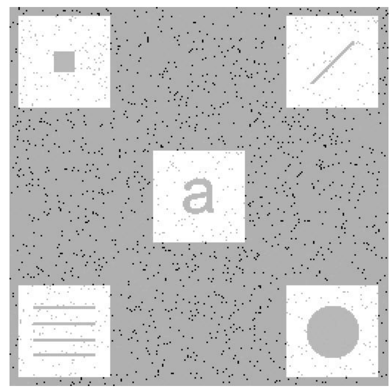


Figure 12: Original Image



(a) Negative of Image



(b) Global Histogram Equalisation on negative image

Figure 13: Combination of techniques

6 Conclusion and Discussion

Creating the GUI itself is a main challenge in itself, since this is the first time building a GUI. Then while performing transformations, checking whether the pixels are in the range 0 to L-1=255 is another challenge. I felt implementing sliders for sharpening and blurring is the most difficult challenge. Also error handling is another thing that I have learned in the process.

Given more time I would have implemented sliders for sharpening and blurring and also would have added more features to the basic image editor.Also I would have improved the sharpening process. To conclude, overall the image editor is working fine for both colour and gray images.

7 References

1. https://en.wikipedia.org/wiki/Negative_(photography)

2. https://www.youtube.com/watch?v=YXPyB4XeYLA
3. https://en.wikipedia.org/wiki/Photographic_film
4. https://www.sciencedirect.com/science/article/abs/pii/S1077314298907238#:~:text=Histogram%20equalization%20is%20a%20widely,places%20in%20the%20original%20image.
5. https://nptel.ac.in/content/storage2/courses/117104069/chapter_8/8_32.html#:~:text=Image%20sharpening%20refers%20to%20any,contrast%20and%20sharpening%20the%20images.&text=Increasing%20yields%20a%20more%20sharpened%20image.
6. The book named **"Digital Image Processing, Gonzalez and Woods, Prentice Hall"**
7. Geeks for Geeks
8. Stack Overflow
9. tex.stackexchange.com
10. Images are taken from the google website