**Image Enhancement Using Fuzzy Logic**

# Project Report

*Submitted in partial fulfillment of the requirements for the award of the degree of*

Bachelor of Technology

in

Department of Applied Mathematics by

**Pankaj** (2K17/MC/071)

**Naveen Kumar** (2​K17/MC/067)

*under the guidance of*

**Ms. Tanay Malhotra** (Assistant Professor)​

Department of Applied Mathematics​



DEPARTMENT OF APPLIED MATHEMATICS

**DELHI TECHNOLOGICAL UNIVERSITY**

Bawana road, Delhi-110042

# LIST OF CONTENTS

**Chapter 1: Problem Statement**

1.1 Objective

1.2 Motivation

**Chapter 2: Motivation**

2.1 What are Recommendation Systems

2.1.1 Why Recommendation systems

2.1.2 What can be Recommended

2.2 Types of Recommendation System

2.2.1 Content-Based Filtering

2.2.2 Collaborative Filtering

2.2.3 Hybrid Models

**Chapter 3: Introduction**

3.1​ Content-Based Modelling

3.1.1 Advantages

3.2 Clustring

3.2.1 Kmeans

**Chapter 4: preliminaries**

4.1​ Data cleaning

4.2 Model Making

4.3 Hosting over a Local Server

4.3.1 Introduction to Flask

**Chapter 5: Methodology and Result**

**Chapter 6: Conclusion**

**CHAPTER 1: PROBLEM STATEMENT**

In this report, we address the problem of bad quality and bad contrast of an image, to resolve this issue we are using the Support Fuzzification method with the Framework of the Logarithmic Model.

**CHAPTER 2 : MOTIVATION**

Fuzzy set theory has obtained great popularity in the image processing community in the last decades. Fuzzy techniques are nonlinear and knowledge-based. They can process imperfect data if this imperfection originates from vagueness and ambiguity rather than randomness.

We propose a fuzzy rule-based approach to image enhancement to address its seemingly conflicting goals:

(a) removing impulse noise,

(b) smoothing out nonimpulse noise,

(c) enhancing edges

**CHAPTER 3: INTRODUCTION**

* 1. **Image enhancement**

Image enhancement is simply a technique which improves the quality of the image, increases the perceptibility of the image which is quintessential in the fields such as medical imaging, surveillance, remote sensing etc. Further this acts as a preprocessing for applications like segmentation, recognition etc.

**3.2. Histogram**

Histogram is important in image processing as it acts as a graphical representation of the tonal distribution in a digital image. It is a graph showing the number of pixels in an image at each different intensity value found in that image.

**3.3 Fuzzy Logic**

Human brain is capable of making excellent decisions using imprecise & incomplete sensory information provided by the perceptive organs. Fuzzy theory provides a systematic calculus to deal with such information linguistically and perform numerical computations using linguistic labels in the form of membership functions. Fuzzy inference system (FIS) when selected properly can effectively model the human expertise in the specific application

This paper proposes an FIS which can enhance the contrast of low resolution images effectively, these results are then compared with the traditional histogram equalization and various evaluation metrics are tabulated.

**CHAPTER 4: PRELIMINARIES**

**Literature**

**Color Image Enhancement**

**Image enhancement** is used when we need to focus or pick out some important features of an image. For example, we may want to sharpen the to bring out details such as car license plate number or some areas of an X-ray film. In aerial photographs, the edges or lines may need to be enhanced to pick outbuildings or other objects. Certain sectoral components of an image may need to be enhanced in images obtained from telescopes. In some cases, the contrast may need to be enhanced.

**How can We achieve this?**

Colour image enhancement is achieved by enhancing first the transformed grayscale image and, then, transforming back the grayscale image into the colors. The color image enhancement is done on the transformed 2-D grayscale image rather than on the color image.

The particularity of the approach is that

* The **logarithmic representation** of images, i.e. the image values are elements of another (Euclidean) space, not R(the real line) in the case of grey images.
* And the image is structured using **fuzzy partitions**.

Better results can be obtained if the image can be divided into statistically uniform sub-images i.e. defining a partition on the image support and allowing a different transform in each sub-images of the partition. Using classical partitions, we are faced with a block effect at the border of the sub-images. To avoid this drawback the classical partitions can be replaced by fuzzy partitions. Their elements will be the “fuzzy windows”. In each of them, an affine transform will be defined using the fuzzy mean and fuzzy variance computed for the pixels of the analyzed window. The final image is obtained by summing up the images of every fuzzy window in a weight way. The weights used are membership degrees, which define the fuzzy partition.

The logarithmic models have created a new environment for developing some new methods of image enhancement. The special thing in our model is that this one uses both positive and negative values for grey level representation. More, the grey level set is bounded and it is a support for a structure of a real vector space. Thus, it is possible to have the grey level addition and subtraction plus the multiplication with positive and negative scalars. This model can be easily extended from grey levels to colors. Moreover, the affine transforms defined for grey level images can be used for image color components without any changes.

**THE FUNDAMENTALS OF THE LOGARITHMIC MODEL**

A monochrome or grey level image is described by a real and bounded intensity function

f: Ω →V

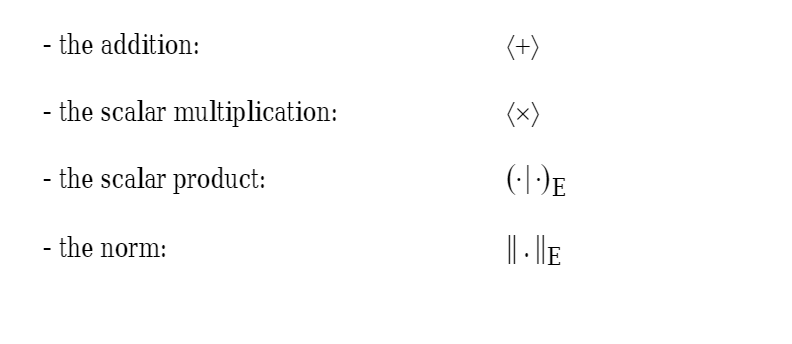
The two parameters ( Ω, V) have the following meaning:

1) Ω is a region in the real plane such as rectangle and a ≤ x ≤ b,c ≤ y ≤ d, and represents the image support;

2) V is a real and bounded set and represents the grey level space.

The value f (x, y) of the intensity function represents the grey level at the point (x, y) ∈ Ω. The problems appear when processing an image: the mathematical operations concerning the real functions implicitly use the real number algebra and, thus results are spread on the whole real axis. This is in contradiction with the fact that the values of the intensity function must be bounded, so, at the end of the procedure, it is necessary to truncate the results. The mathematical model used in this project has a different behaviour compared to the classic model and removes the difficulty shown before: all the results are inside the domain of the expected values.

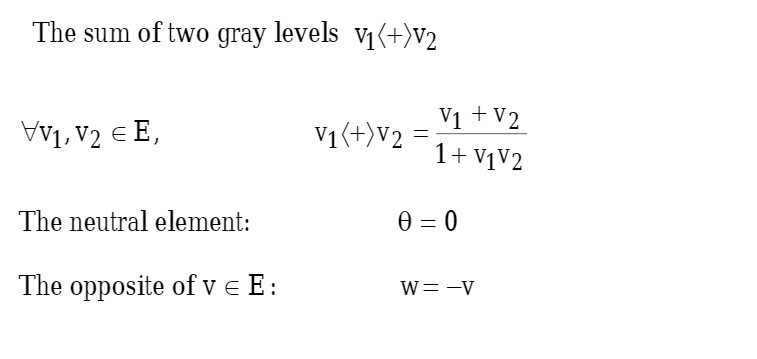
**Notations**



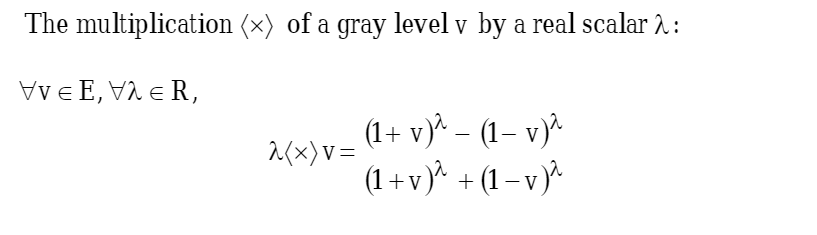
**The vector space of gray levels**

Let us consider the space of gray levels as the set E = (−1,1) . In the set of gray levels we will define the addition 〈+〉 and the multiplication 〈×〉 by a real scalar and then, defining a scalar product ( ) E ⋅| and a norm ⋅ || , we shall obtain a Euclidean space of .||V gray levels.

**The addition of gray**

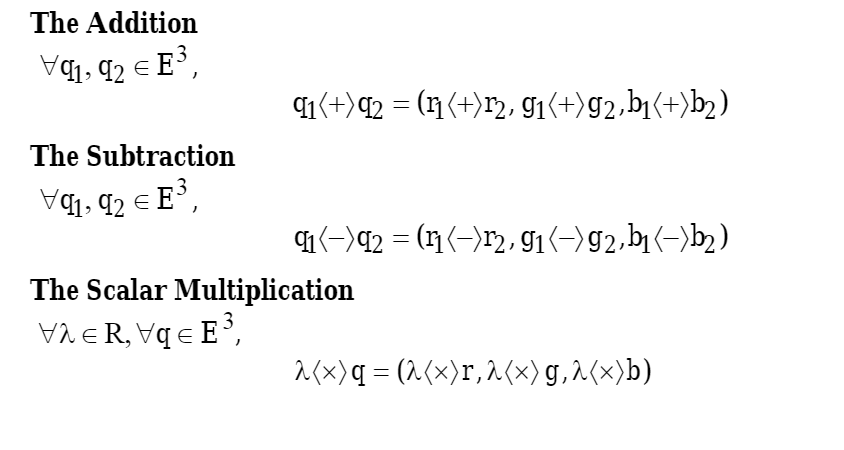


**THE MUTLIPLICATIONS BY SCALER**

****

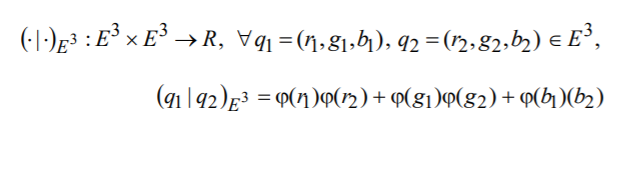
**The Logarithmic Model for Color image**

Consider the cube E^3 as the color space. Let be Q = (r, g, b)∈E^3, a color having the components r(red), g(green) and b(blue).

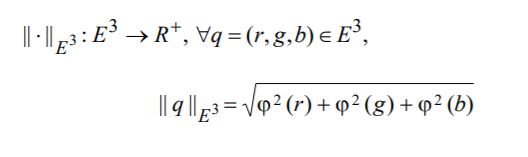
****

**The Euclidean Space of the Colors**

The Scaler product:



The Norm**:**

****

**The Fuzzification of the Image Support**

A gray level image is described by its intensity function:

f: D - > E

where

D **⊂** R ^2

is the image support.

Without loss of generality, the rectangle

D = [x0, x1] x [ y0, y1]

can be considered as the image support.

The coordinates of a pixel within the support

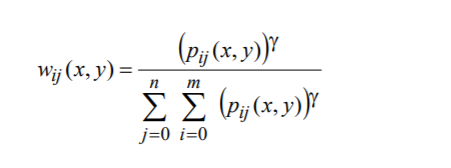
D will be noted (x, y).

Let there be

**P = {Wij | (i, j) ∈ [0, m] x [0, n]}** a fuzzy partition of the support D.

The membership degrees of a point (x, y)∈ D to the fuzzy window Wij are given by the functions:

Wij: D - > [0,1],



The membership degrees wij (x, y) describe the position of the point (x, y) within the support D. The parameter γ ∈(0,∞) has the role of a tuning parameter offering a greater flexibility in building the fuzzy partition P. In other words, γ controls the fuzzification-defuzzification degree of the partition P.

**ALGORITHM**

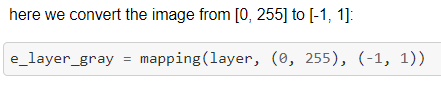
We can enhance the colors in an image by tunning two things brightness and contrast, this algorithm divided the image into fuzzy windows and every pixel has a membership degree to every window, the membership degrees are calculated depending on the distance between the window and the pixel, then the means and variances are calculated with respect to the membership degrees, the final image is obtained by summing up the images of every fuzzy window in a weight way, the weights used are membership degrees

**Algorithm Steps:**

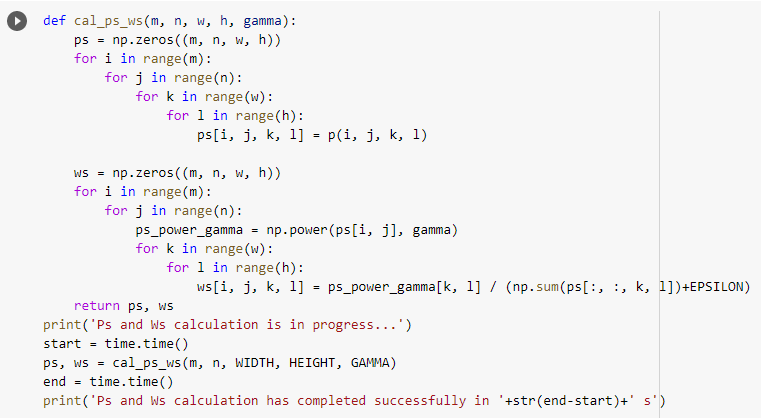
* Convert the image to gray level space (if it is rgb convert every channel and work on one channel at a time then re-merge them).
* Calculate the pixels weights for every window.
* Calculate windows means and variances in a wieghted way.
* Do b\_ij\*f + a\_ij where "a\_ij" is the mean of ij window, "b\_ij" is the variance of ij window, "f" is the image.
* Summing up the images of every fuzzy window in a weighted way.

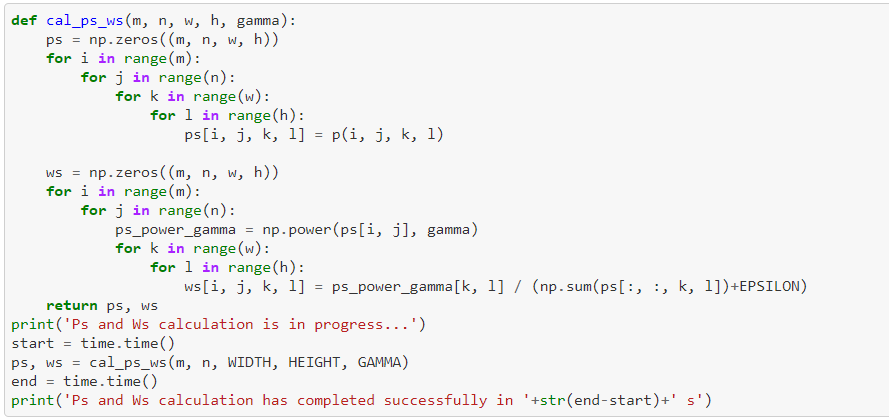
**Detailed Steps:**

**1.** Convert the image to gray level space (if it is rgb convert every channel and work on one channel at a time then re-merge them).



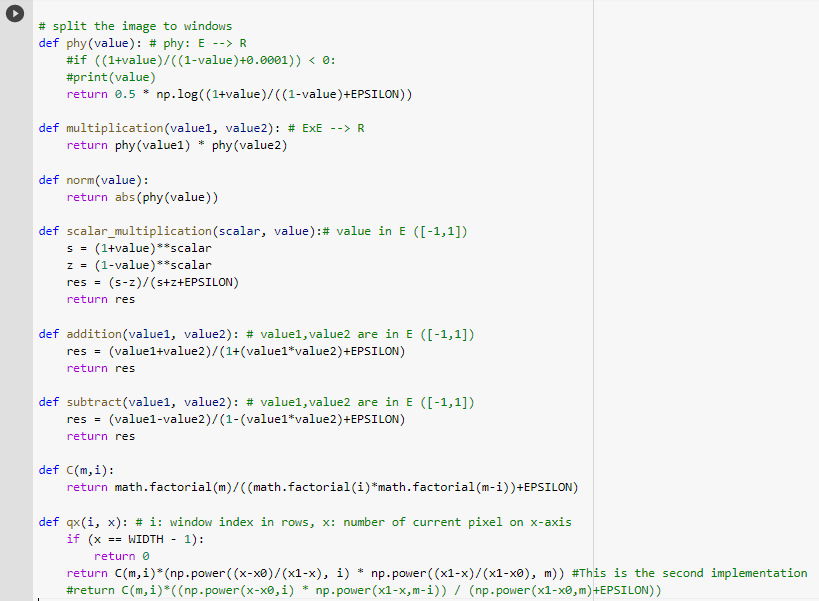
**2.** Calculate the pixels weights for every window.





**3.** Calculate windows means and variances in a weighted way.

**4.** Do b\_ij\*f + a\_ij where "a\_ij" is the mean of ij window, "b\_ij" is the variance of ij window, "f" is the image.



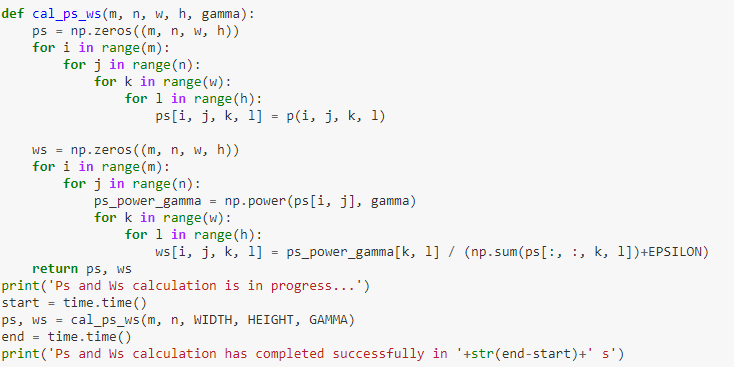
## The enhancement phases are:

* Image Fuzzification.
* Calculate Ps and Ws.
* Calculate cards, means, variances, lamdas.
* Implement windows enhacement.
* Implement Image Enhacement.

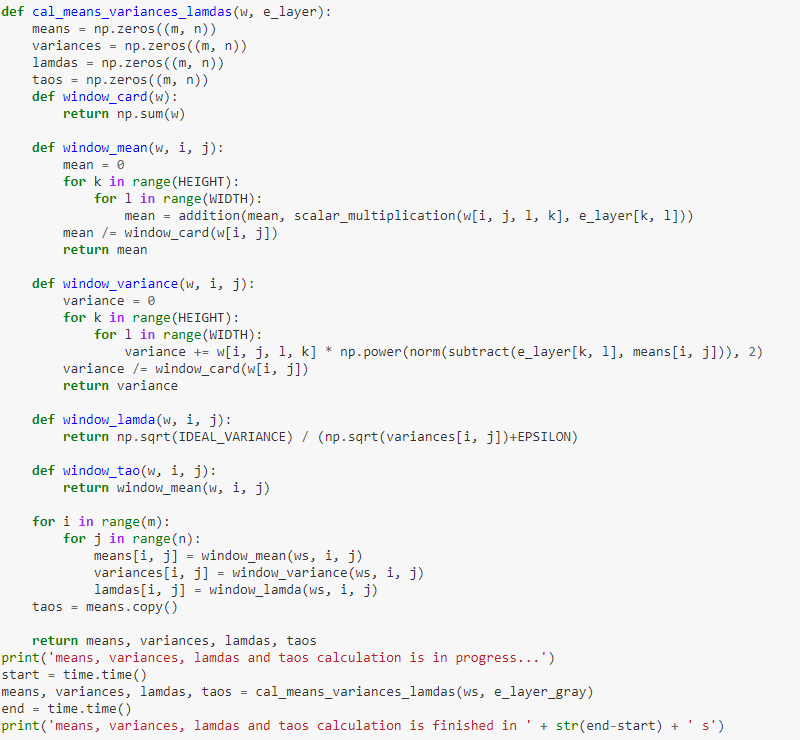
1**) Image fuzzification**

e\_layer\_gray = mapping(layer, (0, 255), (-1, 1)

2**) Calculate Ps and Ws**



### **3)** **Calculate cards, means, variances, lambdas:**

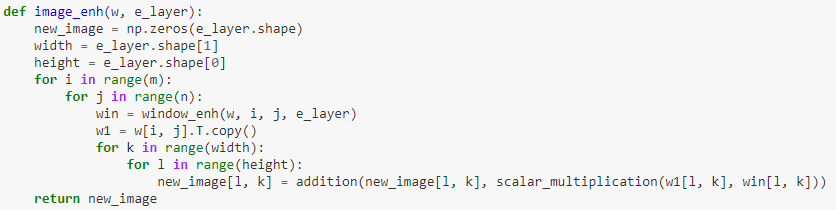


### **4) Implement window enhacement**

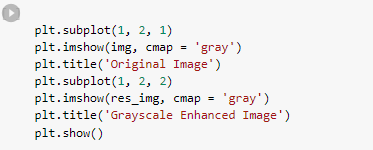
**def** window\_enh(w, i, j, e\_layer):

**return** scalar\_multiplication(lamdas[i, j], subtract(e\_layer, taos[i, j]))

### **5) Implement Image Enhacement:**



**Final Output**



Grayscale enhanced image shown below in the result section.

**RGB Color Space Image Enhancement:**



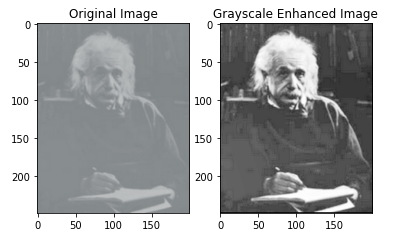
**Final Output**



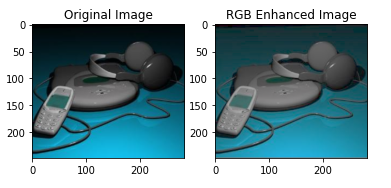
RGB enhanced image shown in below result section.

**CHAPTER 5: RESULT**

**5.1 Grayscale Enhanced Image**



**5.2 RGB Enhanced Image**



**CHAPTER 6: CONCLUSION:**

This project presented a method for color and grey image enhancement. The affine transform parameters are determinate using the mean and the variance of the color image components. Moreover, there are two parameters of the transformation function that offer more flexibility in the tuning of the brightness, contrast and saturation. Using Support of Fuzzification in the Framework of the Logarithmic Model. Simple though powerful methods for image enhancement can be obtained using affine transforms, defined in the framework of the logarithmic model.

Although the method for image enhancement based on Fuzzification in the Framework of the Logarithmic Model is sufficient but in the future, efficient methods can be developed for image enhancement which can give a more accurate result.

**Summary**

The image enhancement method presented here uses point operations. The particularity of the approach is that

* the logarithmic representation of images the image values are elements of another (Euclidean) space, not R (the real line) in the case of grey images.
* and the image is structured using fuzzy partitions.

The final image is obtained by summing up the images of every fuzzy window in a weight way. The weights used are membership degrees, which define the fuzzy partition.