

Global and Localized Histogram Equalization of an Image

Pradeep¹, Namratha M², Manu G V³

^{1,2}(Mtech in Software engineering, Department of Information Science, PESIT, Bangalore,India)

³(BE, MBA, Mtech in Computer Science, SJCIT, Chikkabalapur, Technical Lead-Testing, Calsoft Labs, Bangalore,India)

Abstract

The paper mainly deals with processing an image as well as enhancing it. To achieve the before said target we grab the pixel information from the input image, convert that information into RGB model, then using that model we process and/or enhance it based on the algorithms. The processing of the image is done by modifying the pixel data in relative context while enhancing the image is done based on enhancement algorithms. We analyse the enhancement of the image by comparing the enhanced image with the initial image using their respective histograms. The coding part of the project uses JAVA language on PC platform with Linux/Windows operating system. We have used an integrated development environment called Netbeans which is written in Java and runs everywhere where a JVM is installed, that is on almost all operating systems including Windows and Linux.

Keywords: Global Histogram Equalization, Image enhancement, Image processing, Linear equalization, Pixel grabber, Resolution modification, Scaling and Shearing

1. Introduction

The problem is to select an input image which is user dependent and store the pixel values of this image in a buffer. Using these pixel values, various enhancement and modification techniques are to be applied. Also the histogram[2] of the initial and final image are to be drawn and compared for analysis.

Our project has two main objectives:

1. Image Processing:

To design and develop a program that can process the image inserted by the user and read the pixel values and store it in a buffer. These values are modified in relative context and using these values the final image is displayed.

2. Image Enhancement:

To design and develop a program that can process the image inserted by the user and read the pixel values and store it in a buffer. These values are enhanced based on image enhancement algorithms and the final image is displayed. The initial and final images are compared and the enhancement is analyzed using histograms.

Our project deals with taking an image as input and then process it thus enhancing its tonal(intensity) details using histogram algorithm. The input is an image, such as photographs or frames of video. The output of image processing is an image as well as a graphical representation of the tonal distribution(color intensity) in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the number of pixels in that particular tone. The left side of the horizontal axis represents the black and dark areas, the middle represents medium gray and the right hand side represents light and pure white areas. The vertical axis represents the size of the area that is captured in each one of these zones. Image enhancement technique is also used to improve the clarity and the quality of the final image. The initial and final images are displayed and the differences are noted.

2. Design Considerations

In order to design the solution we first need an analysis of the problem. The ACSS is an incremental model. The telemetry data display, retrieval is initial version for the ACSS system. This section describes many of the issues, which need to be addressed or resolved before attempting to devise a complete design solution.

2.1 Assumption and Dependencies

The main assumptions and dependencies identified are as below:

1. It is assumed that the input image is JPEG format.
2. The operating system is Linux/ Windows.

2.2 General Constraints

The main general constraints identified are as below:

1. The program works only for fixed images.
2. The program works only for unenhanced, low quality images.

2.3 System Architecture

The initial design process of identifying these sub-systems and establishing a frame work for sub-system control and communication is called Architecture design and the output of this design process is a description of the software architecture. The architectural design process is concerned with establishing a basic structural framework for a system. It involves identifying the major components of the system and communications between these components.

2.4 Data flow diagram

The data flow diagrams specify a major transform in the approach to be followed for producing the software. A data flow diagram is a graphical representation of the flow: of the data through an information system. A data flow diagram can also be used for the visualization of data processing (structured design). It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context level DFD is then “exploded” to show more detail of the system being modeled. In this case data flow diagram has been split into three levels level 0, level 1, and level 2.

2.4.1 Pixel Grabber and image enhancement (Level 0)

Context level data flow diagram represents the fundamental system model. The entire software element is represented as a single bubble. In this level 0 data flow diagram ,we are having three external entities like TM Acquisition System, TM Archival System, and Client Retrieval. The Centralized System is main system where the telemetry data distribution, display and retrieval processes carried out very efficiently. We are actually receiving the input from the user as a JPEG image. This image is selected by the user among the several images provided. The image selected for enhancement is then passed for the appropriate enhancement algorithm. Based on the pixel values of the image provided, the enhancement is done. The image can be linearly or non-linearly stretched, exponential enhancement, logarithmic enhancement, sharpening, blurring, contrast, etc.

Equalization[3] is a technique where by the pixel values of the initial image is taken and these values which are randomly distributed are made to be uniformly distributed in such a way that the image has equalized value of the colored pixels. An equalized image is a better version of the initial image. But some images are equalized already and it is not necessary to perform any enhancement on it. In such cases enhancement may reduce the quality of the image and also it's resolution. The various algorithms used are piecewise linear stretching, linear equalization, contrast, global histogram equalization. A histogram is a graph which shows the pixel intensity variation[3],[4]. A histogram is initially drawn for the image before enhancement and even after enhancement. After enhancement, by observing the histogram we can realize the smoothness of the curve indicating that the image is equalized. All the ups and downs are uniformly distributed and makes the image a better quality one. Finally the initial and final images are compared to note the differences.

Steps:

1. First every image given by the user is checked whether it's a fixed image and the pixel values are stored.
2. After the image is selected, one or/and more of the enhancement algorithms is applied to improve the image quality.
3. The pixel values are manipulated based on the enhancement algorithm used.

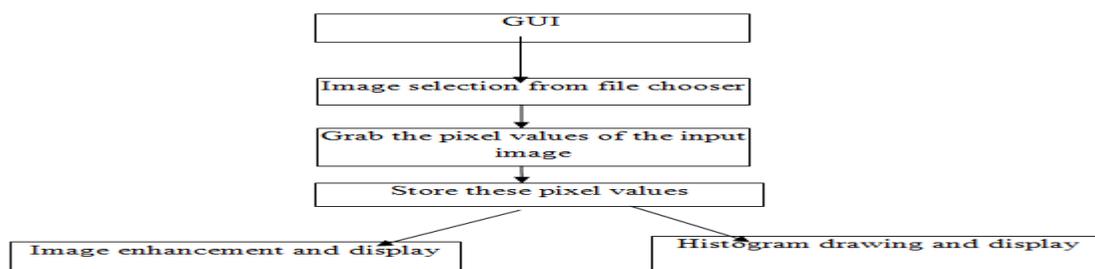


Figure 1: Level 0 data flow diagram

2.4.2 Histograms (Level 1):

An image histogram is a type of histogram which acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. Image histograms are present on many modern digital cameras. Photographers can use them as an aid to show the distribution of tones captured, and whether image detail has been lost to blown-out highlights or blacked-out shadows.

The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the number of pixels in that particular tone. The left side of the horizontal axis represents the black and dark areas, the middle represents medium gray and the right hand side represents light and pure white areas. The vertical axis represents the size of the area that is captured in each one of these zones. Image editors[5] typically have provisions to create a histogram of the image being edited. The histogram plots the number of pixels in the image (vertical axis) with a particular brightness value (horizontal axis). Algorithms in the digital editor allow the user to visually adjust the brightness value of each pixel and to dynamically display the results as adjustments are made. Improvements in picture brightness and contrast can thus be obtained in image processing and photography, a color histogram is a representation of the distribution of colors in an image. For digital images, it is basically the number of pixels that have colors in each of a fixed list of color ranges, that span the image's color space, the set of all possible colors. The color histogram can be built for any kind of color space, although the term is more often used for three-dimensional spaces like RGB or HSV. For monochromatic images, the term intensity histogram may be used instead. For multi-spectral images, where each pixel is represented by N of measurements, each within it's own wavelength of the light spectrum, some of which may be outside the visible spectrum, the colour histogram is N-dimensional. If the set of possible color values is sufficiently small, each of those colors may be placed on a range by itself; then the histogram is merely the count of pixels that have each possible color. Most often, the space is divided into an appropriate number of ranges, often arranged as a regular grid, each containing many similar color values. The color histogram may also be represented and/or displayed as a smooth function defined over the color space, that approximates the pixel counts.

The histogram provides a compact summarization of the distribution of data in an image. The color histogram of an image is relatively invariant with translation and rotation about the viewing axis, and varies only slowly with the angle of view. By comparing histograms signatures of two images and matching the color content of one image with the other, the color histogram is particularly well suited for the problem of recognizing an object of unknown position and rotation within a scene. Importantly, translation of an RGB image into the illumination invariant rg-chromaticity space allows the histogram to operate well in varying light levels. The main drawback of histograms for classification is that the representation is dependent of the color of the object being studied, ignoring its shape and texture. Color histograms can potentially be identical for two images with different object content which happens to share color information. Conversely, without spatial or shape information, similar objects of different color may be indistinguishable based solely on color histogram comparisons. There is no way to distinguish a red and white cup from a red and white plate.

JFreeChart[1],[6] is a free 100% Java chart library that makes it easy for developers to display professional quality charts in their applications. JFreeChart's extensive feature set includes:

1. a consistent and well-documented API, supporting a wide range of chart types;
2. a flexible design that is easy to extend, and targets both server-side and client-side applications;
3. support for many output types, including Swing components, image files (including PNG and JPEG), and vector graphics file formats (including PDF, EPS and SVG);
4. JFreeChart is "open source" or, more specifically a free software.

Steps:

1. Read and store the pixel values of the input image selected by the user.
2. Pass these values of pixels to the JFreeChart.
3. The JFreeChart draws the histogram.
4. This histogram is then passed to the display module.
5. The histograms for initial and final images are displayed.

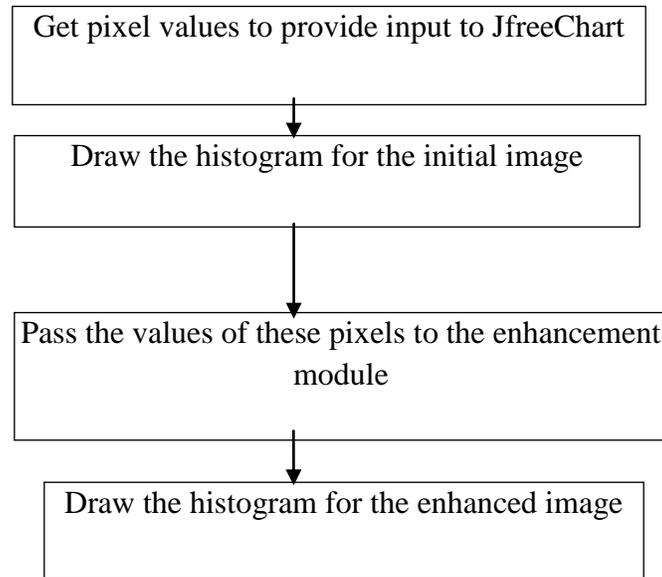


Figure 2: Level 1 data flow diagram

2.4.3 Display of the image(Level 2):

Steps:

1. The image after the enhancement must be displayed along with the initial image back to the user.
2. Also the histograms of both the images must be appropriately displayed.
3. Various options are to be provided at the front end to choose which enhancement technique to be used for the image selected to display.
4. The gray scale image also may be displayed when required.

Pixel Grabber is used to grab the pixel values from the image and it is stored in the buffer. Memory image source[1],[7] is used to obtain these values. Frames and panels are the basic units where an image is displayed in java. The java.awt.image class is the superclass that represents graphical images as rectangular arrays of pixels. The java.awt.image.BufferedImage class, which extends the Image class to allow the application to operate directly with image data (for example, retrieving or setting up the pixel color). Applications can directly construct instances of this class. The Buffered Image class[1],[8],[9] is a cornerstone of the Java 2D immediate-mode imaging API. It manages the image in memory and provides methods for storing, interpreting, and obtaining pixel data. Since Buffered Image is a subclass of Image it can be rendered by the Graphics and Graphics2D methods that accept an Image parameter. A Buffered Image is essentially an Image with an accessible data buffer. It is therefore more efficient to work directly with Buffered Image. A Buffered Image has a Color Model and a Raster of image data. The Color Model provides a color interpretation of the image's pixel data. The Graphics.drawImage method draws an image at a specific location. The GUI provides the panel necessary for the display of the initial and final images.

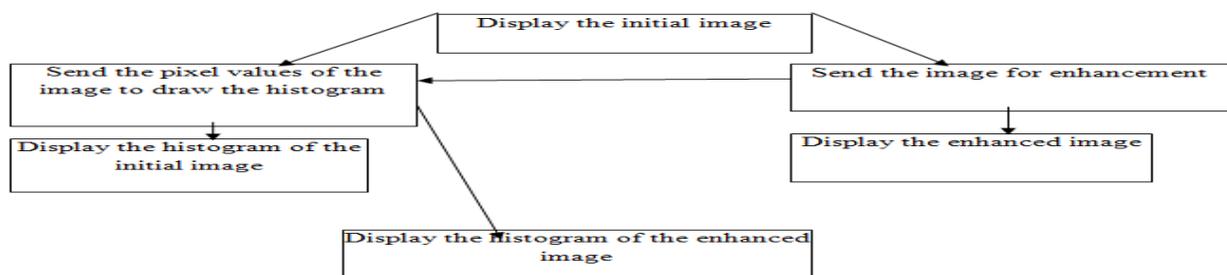


Figure 3: Level 2 data flow diagram

2.5 Structured Chart

This structural model shows how a function is realized by a number of other functions which it calls; structural charts are a graphical way to represent a decomposition hierarchy. Like data-flow diagrams, they are dynamic rather than static system models. They show how function calls others. They do not show the static block structure of a function or procedure. A function represented on a structure chart as a rectangle. The hierarchy is displayed by linking rectangles with lines. Inputs and outputs are indicated with annotated arrows. An arrow entering a box implies input leaving a box implies output. Data stores are shown in the rounded rectangles and user inputs are circles. The structure Chart is made up of the modules of the project together with the interconnections between modules. In this structural chart, the GUI is a main module for all sub modules. In this case we receiving input image from the user is a submodule. Within this GUI only we are implementing these sub-modules. Actually the user interacts through this GUI. The various enhancement algorithms are submodules too. The submodules of the enhancement algorithms can be linear and non-linear. Drawing and displaying the histogram also is a part of the submodule of the GUI.

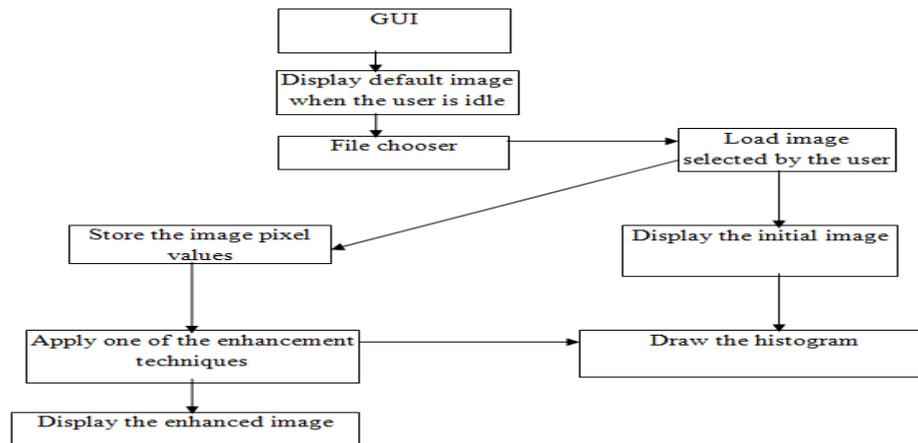


Figure 4: Structure chart

2.5.1 Module 1: GUI

Introduction

This module represents the main part of the GUI and the flow of the entire project begins here. This mainly involves the GUI i.e User and Administrator Interface to manage the user input and selection of the images.

Purpose

The main purpose of the GUI module is to give access to Administrator, Users and Super Users to display the image in real-time, and also selection among the various images provided. A default image is always displayed when the GUI is idle and user interaction is happening. The GUI must be interactive and responsive.

Functionality

This module receives the input from the user. The received input image is sent to the corresponding functions for processing. This module will act as a main system for other functions and from this system we are sending the input image for further enhancements. The various steps are below.

Step 1: Start

Step 2: Select image name to be provided as the input for enhancement from the file chooser provided.

Step 3: Display the initial image that is selected by the user.

Step 4: Send the received input image to the further module for enhancements.

Step 5: The further modules are responsible for the enhancement.

Step 6: The output of this stage is an input to the next module which does the required enhancement and then displays the image along with it's histogram.

Step 7: It is used for display of the images and the histogram and also an interface between the user input and the enhancement algorithms

Step 8: Stop

Input

This module accepts the inputs from the user in the form of the image selected from the file chooser.

Output

This module send the selected image to the further modules for enhancement.

Inter component Relationship

This module receives inputs from the user which is received by subsequent modules for further enhancements and display.

2.5.2 Module 2: Input image Retrieval

Introduction

The particular image input have to be retrieved from the GUI where the user interacts. The module is responsible for checking whether the image retrieved as per the specifications(fixed image).

Purpose

This module will be responsible for getting inputs from user for retrieving data and report it to display module.

Functionality

This module accepts input from user through the GUI.

The various steps are as below.

Step 1: Start

Step 2: Select a file name which contains the input image.

Step 3: Display the initial image.

Step 4: Grab the pixel values of the input image and store it in a buffer for modification during enhancement.

Step 5: Send the retrieved input to the Display module through the main GUI for different displays.

Step 6: If we want retrieve pixel values for another input image than go to Step 2 otherwise go to Step 7

Step 7: Stop

Input

This module accepts input from the user as an image and sends it for displaying and drawing of the histogram.

Output

This module returns the enhanced image as the output.

Inter component Relationship

This retrieves the input from the user and produces the retrieved inputs as an output to the Display module through the GUI.

2.5.3 Module 3: Image Display

Introduction

In image display, the particular image to be enhanced must be retrieved and based on the pixel values stored, the image is appropriately drawn. There are various inbuilt functions for drawing the image from the pixel values. The appropriate packages must be imported. The histogram for the image must also be displayed.

Purpose

The main purpose of the image Display module is to display the initial and final image after enhancement.

Functionality

his module accepts inputs as pixel values from the buffer where it is stored. Based on the value of the pixels, the image is drawn onto the display. The pixel values determine the resolution of the image being displayed.

Steps:

The various steps are as below:

Step 1: Start

Step 2: Select the image name for display

Step 3: Select the image format for display

Step 4: Start to receive pixel data from the buffer and start drawing the image onto the screen.

Step 5: Display the image in the required proper format.

Step 6: Send the display details to Display_Mode module for different displays.

Step 7: Stop

Input

This module accepts the input data as the pixel values of the input image which is stored in the buffer..

Output

This module returns the pixel data that is raw data display to Display_Mode.

Inter component Relationship

This function receives input from the Retrieval module to produce the image and histogram display as output. That displayed output send to Display_Mode module.

2.5.4 Module 4: Histogram display Mode[10]

Introduction

Already we know about full frame mode display in where both the initial and final images are displayed together along with their histogram respectively.

Purpose

This module is responsible for the displaying only selected single frame from the master frame either only the image or the histogram.

Functionality

This module is responsible to displaying selected single frame and drawing the histogram. JfreeChart is used to plot the graph by taking pixel values as the input.

The steps are as below.

Step 1: Start

Step 2: Make sure that the appropriate libraries are included that the JfreeChart requires for plotting the histogram.

Step 3: The pixel values of the image for which the histogram is to be drawn are pass to the JFreeChart.

Step 4: Then it reads these pixel values and plots the graph.

Step 5: The histogram is displayed for the image selected.

Step 6: Stop

Input

It receives pixel values from the image for which the histogram is to be plotted.

Output

The graphical representation of the intensity of the image by drawing the histogram for the image selected by the user.

Inter component Relationship

It mainly depends on the intensity of the pixels of the image selected. Depending on pixel values, a smooth or sharp or dense or exponential or linear or non-linear curve may be obtained.

2.5.5 Module 5: Stepwise Linear stretching

Introduction

This is an enhancement algorithm which takes the input image given by the user and enhances the image and displays it along with it's histogram.

Purpose

This module is responsible for stretching the pixel values so that the image intensity becomes more uniform.

Functionality

This module is responsible to displaying the selected image and compare it before and after enhancement . The various steps are as below.

Step 1: Start

Step 2: The image to be enhanced is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform. The intensity is high at the peaks. All the pixel values at peaks are stretched so that the histogram curve becomes smooth and uniform.

Step 4: The modified pixel values are then used to draw the enhanced image.
Step 5: This image is now sent to the display module.
Step 6: Stop

Input

It receives pixel values grabbed from the selected input image.

Output

The selected input image is modified and the final image is displayed along with the histogram.

Inter component Relationship

It mainly depends on the pixel values of the input image and the intensity curve distribution.

2.5.6 Module 6: Linear equalization[11]

Introduction

This is an enhancement algorithm which takes the input image given by the user and enhances the image and displays it along with its histogram.

Purpose

This module is responsible for equalizing the pixel values so that the image intensity becomes more uniform and is equally distributed.

Functionality

This module is responsible for displaying the selected image and compare it before and after enhancement . The various steps are as below.

Step 1: Start

Step 2: The image to be enhanced is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform. The intensity is high at the peaks.

Step 4: The maximum and minimum values of the occurrences of the image pixel values are identified and the value less than minimum is considered as zero and the value greater than maximum is 255.

Step 5: The modified pixel values are then used to draw the enhanced image.

Step 6: This image is now sent to the display module.

Step 7: Stop

Input

It receives pixel values grabbed from the selected input image.

The selected input image is modified and the final image is displayed along with the histogram.

Inter component Relationship

It mainly depends on the pixel values of the input image and the intensity curve distribution.

2.5.7 Module 7: Modification of the resolution

Introduction

This is a modification technique which takes the input image given by the user and enhances the image and displays both the images onto the frame.

Purpose

This module is responsible for modifying the pixel values so that the image intensity becomes as desired by the user.

Functionality

This module is responsible for displaying the selected image and compare it before and after modification . The various steps are as below.

Step 1: Start

Step 2: The image to be modified is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform.

Step 4: It contains six options:

3. Brightness
4. Darkness
5. Edge detection
6. Negative

7. Reset

Step 5: Based on the option selected, the pixel values are modified

Brightness is used to increase the resolution of the pixels so that it looks more bright where as darkness does the opposite of this function. Negative inverts the value of each and every pixel value. Reset is used to get back the initial image after applying one of these modification techniques so that the next modification can be applied later.

Step 6: The modified pixel values are then used to draw the enhanced image.

Step 7: This image is now sent to the display module.

Step 8: Stop

Input

It receives pixel values grabbed from the selected input image.

Output

The selected input image is modified and the final image is displayed. The initial image can be got back by clicking on the reset option.

Inter component Relationship

It mainly depends on the pixel values of the input image and the resolution of the pixel values of the image being used.

2.5.8 Module 8: Global Histogram equalization[11]

Introduction

This is an enhancement algorithm which takes the input image given by the user and enhances the image globally and displays both the initial and final images.

Purpose

This module is responsible for equalizing the pixel values so that the image intensity becomes more uniform and is equally distributed. This is based on the global values of the pixels

Functionality

This module is responsible to displaying the selected image and compare it before and after enhancement . The various steps are as below.

Step 1: Start

Step 2: The image to be enhanced is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform.

Step 4: The pixel values of the image to be enhanced are used to get the RGB (Red Green Blue) components. Using this, the HSV(Hue Saturation Value)

components of the image are obtained. The hue and saturation remain the same, where as the value is modified to enhance the image intensity. Then the modified value is used and the new HSV values are converted back into their corresponding RGB values and this value of pixels are used to draw the enhanced image.

Step 5: The modified and enhanced pixel values are then used to draw the enhanced image.

Step 6: This image is now sent to the display module.

Step 7: Stop

Input

It receives pixel values grabbed from the selected input image.

Output

The selected input image is globally equalized based on the RGB values of the pixels and the final image is displayed along with the histogram.

Inter component Relationship

It mainly depends on the pixel values of the input image and the intensity curve distribution and mainly on the RGB values and the corresponding HSV values.

2.5.9 Module 9: Scaling and shearing of images

Introduction

This is a processing algorithm which takes the input image given by the user and modifies the image for scaling and shearing[12] effects and displays both the initial and final images.

Purpose

This module is responsible for modifying the pixel values so that the image intensity becomes as desired by the user.

Functionality

This module is responsible to displaying the selected image and compare it before and after enhancement . The various steps are as below.

Step 1: Start

Step 2: The image to be enhanced is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform.

Step 4: If it is scale value then matrix becomes

[sx 0 0] sx-The factor by which the co ordinate are scaled along x axis direction.

[0 sy 0] sy-The factor by which the co ordinate are scaled along y axis direction.

If it is shear value matrix becomes

[1 shx 0] shx-Multiplier by which co-ordinates are shifted in the direction of +ve X axis as factor of their y co-ordinates.

[shy 1 0] shy- Multiplier by which co-ordinates are shifted in the direction of +ve Y axis as factor of their X co-ordinates.

Step 5: Draw the processed image on buffer using graphics function.

The modified and enhanced pixel values are then used to draw the enhanced image.

Step 6: This image is now sent to the display module.

Step 7: Stop

Input

It receives pixel values grabbed from the selected input image.

Output

The selected input image is modified and the final image is displayed. The initial image can be displayed again and compared with the final image..

Inter component Relationship

It mainly depends on the pixel values of the input image and the resolution of the pixel values of the image being used.

2.5.10 Module 10: Filters

Introduction

This is a processing algorithm which takes the input image given by the user and modifies the image for scaling and shearing effects and displays both the initial and final images.

Purpose

This module is responsible for modifying the pixel values so that the image intensity becomes as desired by the user.

Functionality

This module is responsible to displaying the selected image and compare it before and after enhancement . The various steps are as below.

Step 1: Start

Step 2: The image to be enhanced is selected by the user and the pixel values are grabbed.

Step 3: The pixel values of the initial image are not uniform.

Step 4: Here we are doing three kinds of filters, blue, green and red. In green filter, we consider the RGB color matrix and in that we make only the value of green as 1 and the others 0. Similarly for red and blue.

Step 5: The modified and enhanced pixel values are then used to draw the enhanced image.

Step 6: This image is now sent to the display module.

Step 7: Stop

Input

It receives pixel values grabbed from the selected input image.

Output

The selected input image is modified and the final image is displayed. The initial image can be displayed again and compared with the final image..

Inter component RelationshipIt mainly depends on the pixel values of the input image and the resolution of the pixel values of the image being used.

3. TESTING:

Test Cases for Unit Testing:

Module under test	Image retrieval
Description	Retrieving the pixel values from the input image
Sample input	fish.jpg
Expected output	It should store the pixel values of this image
Actual output	The pixel values of the input image are stored
Remarks	Test is successful

Module under test	Image display
Description	Display the images in the required format
Sample input	fish.jpg
Expected output	It should display the initial and final image in the required fashion and format
Actual output	The initial and final images are displayed as required
Remarks	Test is successful

Module under test	Histogram display
Description	Display the histogram for the initial and final images
Sample input	fish.jpg
Expected output	It should draw the histogram based on the pixel values passed to the JFreeChart
Actual output	The histogram is drawn and displayed
Remarks	Test is successful

Module under test	Stepwise linear stretching
Description	It equalizes the pixel values by stretching the values present at the peak
Sample input	fish.jpg
Expected output	The enhanced image along with the histogram
Actual output	The enhanced image is displayed
Remarks	Test is successful

Module under test	Linear equalization
Description	It equalizes the pixel values by finding the minimum and maximum occurrences of the pixel values
Sample input	fish.jpg
Expected output	The enhanced image along with the histogram
Actual output	The enhanced image is displayed
Remarks	Test is successful

Module under test	Modification of resolution
Description	It modifies the pixel values either brighten or darken or form the negative or contrast
Sample input	fish.jpg
Expected output	The enhanced image must be displayed
Actual output	The enhanced image is displayed
Remarks	Test is successful

Module under test	Global Histogram equalization
Description	It equalizes the pixel values globally by converting the RGB values to HSV. Then the value is enhanced and converted back to RGB
Sample input	fish.jpg
Expected output	The enhanced image along with the histogram
Actual output	The enhanced image is displayed
Remarks	Test is successful

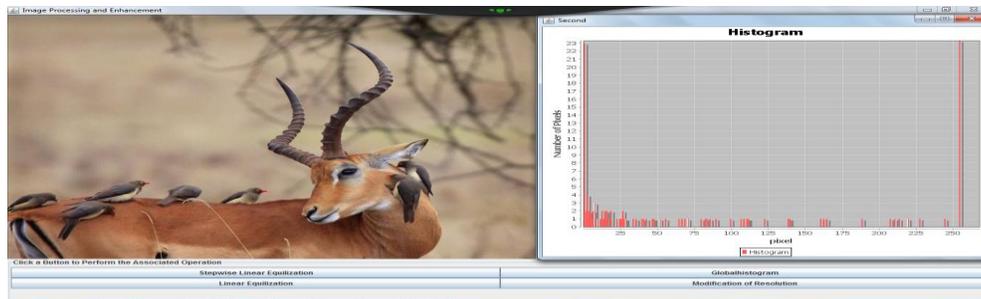
Module under test	Scaling and Shearing images
Description	It scales or shears the input image by a given value along the x or y axis according to a matrix. The modified pixel values are used to draw the final image.
Sample input	fish.jpg
Expected output	The modified image along with the histogram
Actual output	The modified image is displayed
Remarks	Test is successful

Module under test	Filters
Description	It uses three different filters blue, green and red. A matrix is used for modification of the pixel values. These modified pixel values are used for the final output.
Sample input	fish.jpg
Expected output	The modified image along with the histogram
Actual output	The modified image is displayed
Remarks	Test is successful

4. SCREEN SHOTS

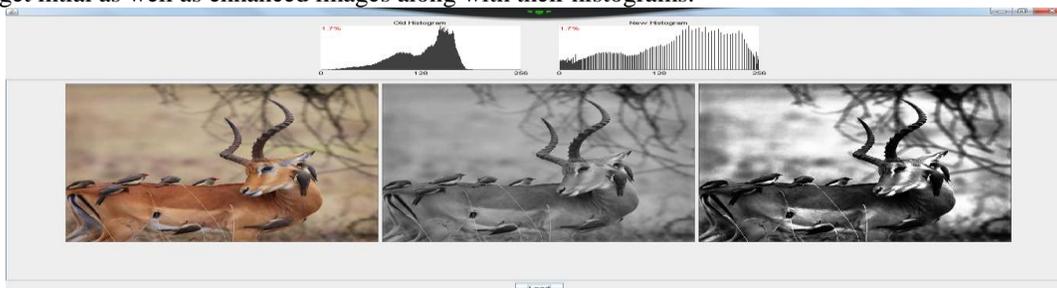
Run the Main program.

We will get the intial image and its histogram.



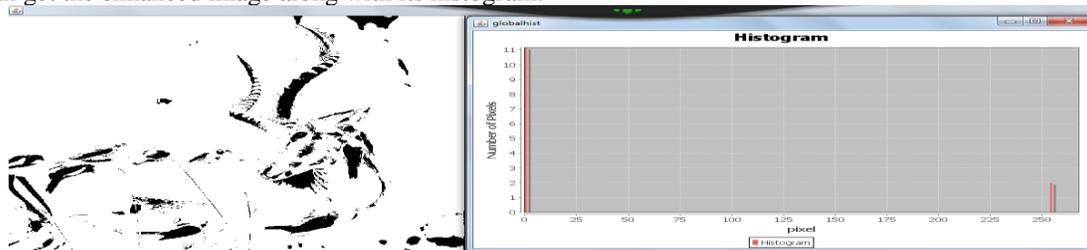
Initial image and its histogram.

- Choose the option **Stepwise Linear Equalization**.
We shall get initial as well as enhanced images along with their histograms.



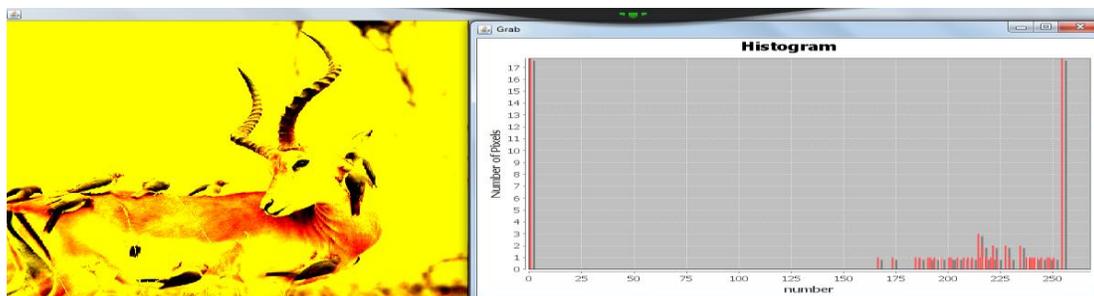
Stepwise Linear Equalization of the image and its histogram.

- Choose the option **Globalhistogram**.
We shall get the enhanced image along with its histogram.



Global Histogram Equalization of the image and its histogram.

- Choose the option **Linear Equalization**.
We shall get the enhanced image along with its histogram.



Linear Equalization of image and its histogram.

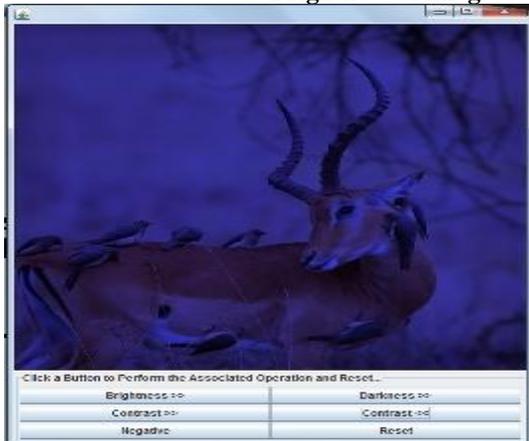
- Choose the option **Modification of Resolution**.
We shall get options to modify the Brightness, Darkness, Contrast and Negative of images.



a. Brightness of image



b. Darkness of image



c. Contrast of image



d. Negative of image

5. CONCLUSION AND FUTURE ENHANCEMENTS

Conclusions

The application deals with the retrieval of pixel values from the selected user image and applying one of the image enhancement algorithm. Then it displays the histogram. The major portion of the project is concentrated on “JAVA” on Linux platform. Various options are provided to implement one of the enhancement algorithm. These include stepwise linear stretching, linear equalization, global histogram equalization, modification of resolution.

The user selects one of the best-suited distribution algorithm for the image selected. The histogram is drawn for both the initial image and the final image. After enhancement, the histogram curve becomes smoother. The application is made interactive by allowing the user to select the image to be enhanced.

Limitations

The limitations of the product are not negative aspects of the work done. However, no product can fulfill all the needs of the user. With respect to this, the following might be the limitations of this package.

Some important limitations are as follows:

1. The package has been designed for only fixed images.
2. The application includes only a few of the algorithms for enhancement and does not cover all possible algorithms.
3. Few vulnerabilities associated with the Linux Operating system are as mentioned below:
 - 3.1 Superuser may misuse the privileges granted to him.
 - 3.2 Many system files can be changed easily
 - 3.3 Modules can be easily used to intercept the kernel.

Future enhancements

The software can be extended to support the following functionality:

- This application will be supported on Linux platform as well as windows operating system.
- This application, which currently supports fixed images can be extended for moving images such as videos and also for satellite images which are in the form of raw binary data.
- Color histograms can also be implemented along with intensity histogram.

References

- [1]. Java: The complete reference, Seventh Edition, Herbert Schildt, Tata McGraw Publishing Company Limited.
- [2]. Digital Image Processing-Concepts, Algorithms, and Scientific Applications, Jaehne. B, Springer, Berlin 1991
- [3]. Digital Image Processing, Third Edition, Rafael. C. Gonzalez and Richard. E. Woods, Prentice Hall 2002
- [4]. The image processing handbook, Fifth edition, John. C. Russ, IEEE Press, New York 1994
- [5]. Digital Image Processing: Principles and Applications, Baxes.G.A., Wiley, New York 1994
- [6]. Digital Image Processing, Gonzalez.R.C. and Wintz.P.A, Addison-Wesley 1977
- [7]. An Introduction to Digital Image Processing, Niblack.W., Prentice Hall 1986
- [8]. http://en.wikipedia.org/wiki/Digital_image_processing
- [9]. <http://www.cromwell-intl.com/3d/histogram>
- [10]. http://designer-info.com/Writing/image_enhancement.htm
- [11]. http://en.wikipedia.org/wiki/Histogram_equalization
- [12]. http://en.wikipedia.org/wiki/Color_model