

SOFTWARE IMPLEMENTATION OF INTRUDER DETECTION

Submitted in partial fulfillment of the Degree of
Bachelor of Technology



May – 2014

Under the supervision of

Dr. D. S. Saini

By

Isha Bhardwaj (101108)

Shikha Puri (101113)

Sumit Madaan (101140)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY,
WAKNAGHAT

CERTIFICATE

This is to certify that the work titled “**SOFTWARE IMPLEMENTATION OF INTRUDER DETECTION**” submitted by Isha Bhardwaj (101108), Shikha Puri(101113) and Sumit Madaan(101140) in partial fulfillment for the award of degree of B.Tech in the Discipline of Jaypee University of Information Technology, Wagnaghat, has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Supervisor Dr. D. S. Saini

Associate Professor, Department of ECE, JUIT Wagnaghat

Date : _____

ACKNOWLEDGEMENT

We would like to thank our supervisor **Dr. D. S Saini** (Associate Prof, ECE) for the invaluable advice and positive encouragement he has provided us throughout the course of this project. We would like to express our deep and sincere gratitude to our project guide for being a staunch supporter and motivator of this project. Right from the inception of this project work, our supervisor guided us till the very end in the true sense of the word.

We would also like to express my gratitude to this alma mater **JUIT, Wagnaghat** for providing proper resources as and when required such as an all-time internet facility and other resources.

Date:

Name of Students

Isha Bhardwaj (101108)

Shikha Puri (101113)

Sumit Madaan (101140)

Table of Contents

Chapter Number	Topic	Page No.
	Summary	vii
	List of Figures	viii
Chapter 1	Introduction	1
1.1	Objective	1
1.2	Aim	1
1.3	The Main Features of the Project	2
1.3.1	Face Detection	2
1.3.2	Face Recognition	2
1.3.3	Contacting the user	2
1.4	Technical Details	3
Chapter 2	Introduction to Digital Image Processing	4
2.1	What is an Image	4
2.2	Pixels	4
2.3	Digital Image Processing	4
2.3.1	Image Acquisition	5
2.3.2	Image Enhancement	5
2.3.3	Image Restoration	6
2.3.4	Color Image Processing	7
2.3.4.1	Hue	7
2.3.4.2	Saturation	7
2.3.4.3	Intensity	8
2.3.4.4	Radiance	8
2.3.4.5	Luminance	8
2.3.4.6	Brightness	8
2.3.5	Image Coding and Compression	9
2.3.6	Image Segmentation	9

2.3.6.1	Thresholding	10
2.3.7	Representation and Description	10
2.3.8	Recognition	11
Chapter 3	Introduction to MATLAB	12
3.1	Introduction	12
3.2	Advantages of MATLAB	13
3.3	Disadvantages of MATLAB	13
3.4	Some Features of MATLAB	14
3.5	MATLAB for Image Processing	14
3.5.1	Binary Images	15
3.5.2	Grayscale Images	15
3.5.3	Truecolor Images or RGB	16
3.5.4	Indexed Images	17
3.6	General Commands in MATLAB	18
Chapter 4	Algorithms	19
4.1	Face Detection	19
4.2	Face Recognition	19
4.3	Algorithms Used	19
4.3.1	Viola Jones Face Detection Algorithm	19
4.3.1.1	The Scale Invariant Detector	20
4.3.1.2	Feature Selection	21
4.3.1.3	AdaBoost	21
4.3.1.4	Cascading Classifiers	22
4.3.1.5	Result of Face Detection	23
4.3.2	Face Recognition using PCA	24
4.3.2.1	PCA	24
4.3.2.2	Eigen Face Approach	24
4.3.2.3	Eigen Values and Eigen Vectors	25
4.3.2.4	Face Image Representation	25
4.3.2.5	Mean and Mean Centered Images	26
4.3.2.6	Covariance Matrix	26
4.3.2.7	Eigen Face Space	27

4.3.2.8	Recognition Step	27
4.3.2.9	Result of Face Recognition	29
Chapter 5	Simulations and Results	30
5.1	Block Diagram	30
5.2	Activation of webcam	32
5.3	Capturing Image	32
5.4	Conversion	32
5.5	Comparison	35
5.6	Result of Comparison	35
5.7	Face Detection	36
5.8	Face Recognition	36
5.9	Result	36
Chapter 6	Applications	37
6.1	Security Issues	37
6.2	Face Detection and Recognition	37
Chapter 7	Conclusion and Future	38
7.1	Conclusion	38
7.2	Future Scope	38
	References	39

SUMMARY

The Project deals with the monitoring of the users surroundings/premises and detecting any human presence in his absence. If that human being is a known of the user, no specific action is taken but if that human is some unknown, then the software immediately sets an alarm and the user is informed of the breach in his premises. We are achieving this via using MATLAB software. The code written on it will activate the cameras to take snapshots of the surroundings. If any human is detected then the image is saved for further analysis. After saving the image we are performing face detection and recognition to find out if the person detected is known or unknown.

Isha Bhardwaj(101108)

Shikha Puri (101113)

Sumit Madaan(101140)

Date:

Dr. D. S. Saini

Date:

List Of Figures

Number	Figure	Page
2.1	Image Processing Model	5
2.2	Original Image	6
2.3	Enhanced Image	6
2.4	Categories of Degradation	7
2.5	RGB Image Components	8
2.6	Example of Image Segmentation	9
2.7	Example of Global Thresholding	10
2.8	Recognition of Specific Chips on Circuit Board	11
3.1	MATLAB Logo	12
3.2	MATLAB Window	14
3.3	Binary Image	15
3.4	Grayscale Image	16
3.5	Truecolor Image	16
3.6	Indexed Image	17
4.1	Integral Image	20
4.2	Sum Calculation	20
4.3	Types of Features	21
4.4	Working Of Cascading Classifier	22
4.5	Original Image	23
4.6	After Face Detection	23
4.7	Result 1	29
4.8	Result 2	29
5.1	Block Diagram of Project	30
5.2	Flow Chart	31
5.3	RGB Mode	33
5.4	YCbCr Mode	34
5.5	Black and White Mode	35

Chapter 1

Introduction

1.1 Objective

Security of one's belongings has become a very important part in this world. There are various security devices which are used nowadays to prevent theft and robbery. Security cameras have become ubiquitous in many countries. Whereas before they appeared only in banks and at high-security areas, they are now entering public places such as malls, streets, stadiums, transport and at homes also. These cameras have several benefits. An obvious benefit is that the police can catch criminals in the act, thus reduce crime. But these cameras need 24*7 monitoring to keep a check on what is happening where.

While deciding about the major project, the main consideration was to make something that would be practically useful and could be utilized in our daily lives as well as various diverse fields. Theft has increased a lot in today's world and not everyone can afford costly security system. We have tried to make cheap and effective security system.

1.2 AIM

1. To set up the equipment so that it can be used at its best.
2. To find the best techniques for image comparison.
3. To understand the different methods of face detection and recognition available, under which various techniques are used.
4. After the analysis of different method a suitable technique is proposed which is more accurate and have high detection and recognition ratio.
5. Compare the proposed technique of face detection and recognition with the other techniques for better evaluation.

1.3 The Main Features of the project are:

1.3.1 Face detection

The demand for Face Detection is growing with a rapid speed because of its vast application in the field of computer vision. For the implementation of different security methods as such face tracking, face recognition and expression recognition and pose estimation all these methods require face detection algorithm. Though it has got a lot of challenges as in variation in poses, illumination, facial expression and occlusion the demand of face detection is still high because of its non-intrusive nature. Because unlike other bio-metric, face detection method gives better identification for security and best result than any other bio-metric methods. It has many benefits. It opens many opportunities in the field of human computer interactions, object detection and recognition and many other surveillance systems. It has a huge scope in future.

1.3.2 Face recognition

It has become a very active area of research in recent years mainly due to increasing security demands and its potential commercial and law enforcement applications. The last decade has shown dramatic progress in this area, with emphasis on such applications as human-computer interaction (HCI), biometric analysis, content-based coding of images and videos, and surveillance. Although a trivial task for the human brain, face recognition has proved to be extremely difficult to imitate artificially, since although commonalities do exist between faces, they vary considerably in terms of age, skin, colour and gender. The problem is further complicated by differing image qualities, facial expressions, facial furniture, background, and illumination conditions

1.3.3 Contacting the User

Today cell phone has become an integral part of our life and through this security system we can get security alert on our mobile phone in case of any incident. Since everyone has the knowledge of using them and various applications provided by the companies it becomes easy to contact to the user automatically because everyone keeps them updated with technology. We will use this technique to inform the user if any illegal activity is going on in his premises.

1.4 Technical Details

For the project we need:

- Laptop or a computer- which supports MATLAB software. MATLAB software for writing and testing of the code, it can be of any version after R20013a. A thorough knowledge of the software and various tools provided by it for simplifications and user friendly methods.
- Webcam of minimum 1.3 mega pixel value for capturing of images.
- Internet connection
- Basic knowledge of Digital Image Processing

Chapter 2

Introduction to Digital Image Processing

2.1 What is an Image?

It is an optical reproduction of an object. An image can be defined as a two-dimensional function, $f(x,y)$, where x and y are the spatial (plane) coordinates, and the amplitude off at any pair of coordinates (x, y) is called the intensity of the image or the gray level of the image at that point. When x , y and the amplitude values of f are all finite, discrete quantities, we call the image a digital image.

2.2 Pixels

The digital image is composed of a finite number of elements, each of which has a particular location and a finite value. These elements are referred to as picture elements, image elements and Pixels.

2.3 Digital Image Processing

Digital Image Processing refers to processing digital images, the two dimensional data by means of a digital computer. The digital image refers to an array of real or complex numbers represented by a finite number of bits. This processing is done to restore, enhance and improve an image for better analysis.

Digital Image Processing has plethora of applications, such as medical processing, robotics, radar and acoustic image processing, it is used in remote sensing via satellites and spacecraft.

This organization is summarized in the Fig 2.1. It does not imply that every process in done on an image but to convey the idea of all the methodologies that can be applied to an image for different purposes and objectives.

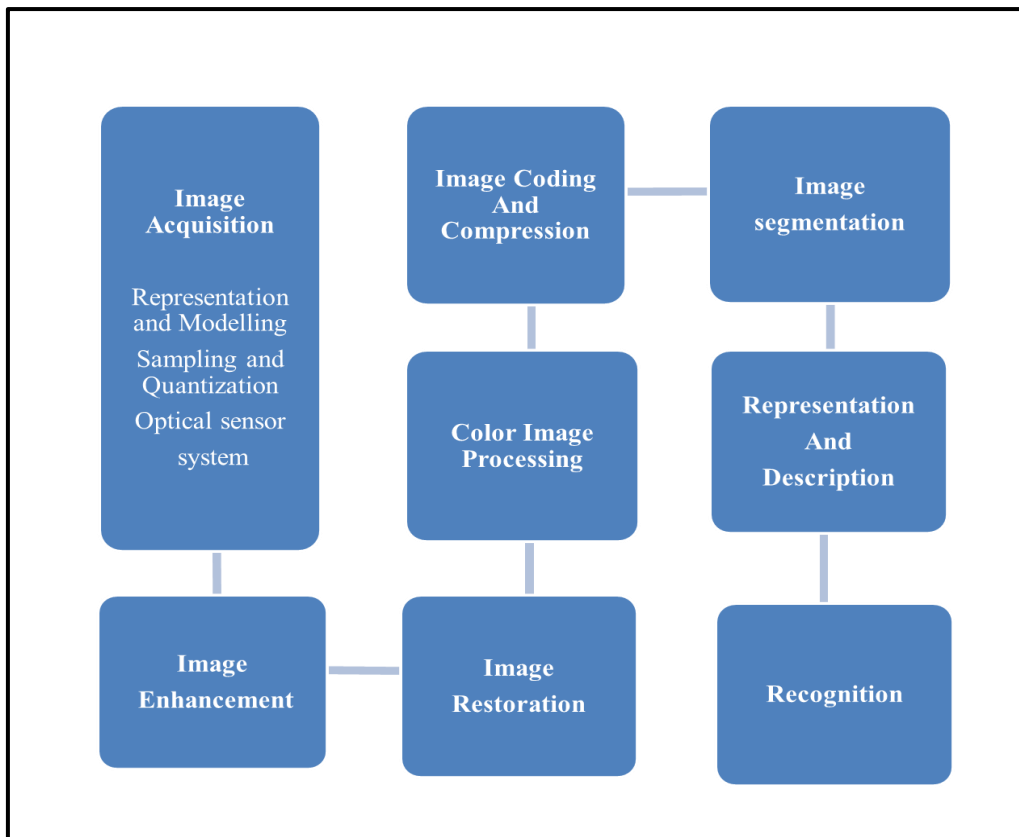


Figure 2.1: Image Processing Model

2.3.1 Image Acquisition

Optical sensor captures the image. It requires a detector, a cmos detector is used. After capturing the image sampling and quantization is done. It is then converted to a two-dimensional pixel array.

2.3.2 Image Enhancement

The main objective of image enhancement is to improve the quality of the image for better human visualization and for analysis so that it can be used suitably for various other applications. It is used to bring out detail that is obscured, or simply highlight certain features of interest in an image. For image enhancement prior knowledge of degradation phenomenon is not required. Some of the operations of image enhancement are removing blurring, increasing contrast and removing noise etc. Image enhancement can be done by two different categories; they are spatial domain method and frequency domain method.



Figure 2.2: Original Image



Figure 2.3: Enhanced Image

2.3.3 Image Restoration

It is the areas that also deal with improving the appearance of an image, unlike enhancement which is subjective, image restoration is objective, in a sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation. It is used to fix color, the contrast, the tone and to repair damages which are caused due to non-linear filtering produced by geometry distortion correction, noise filtering, sensors and environmental factors. Thus restoration properties are used to remove the degradation and putting on inverse process in order to get back the original image.

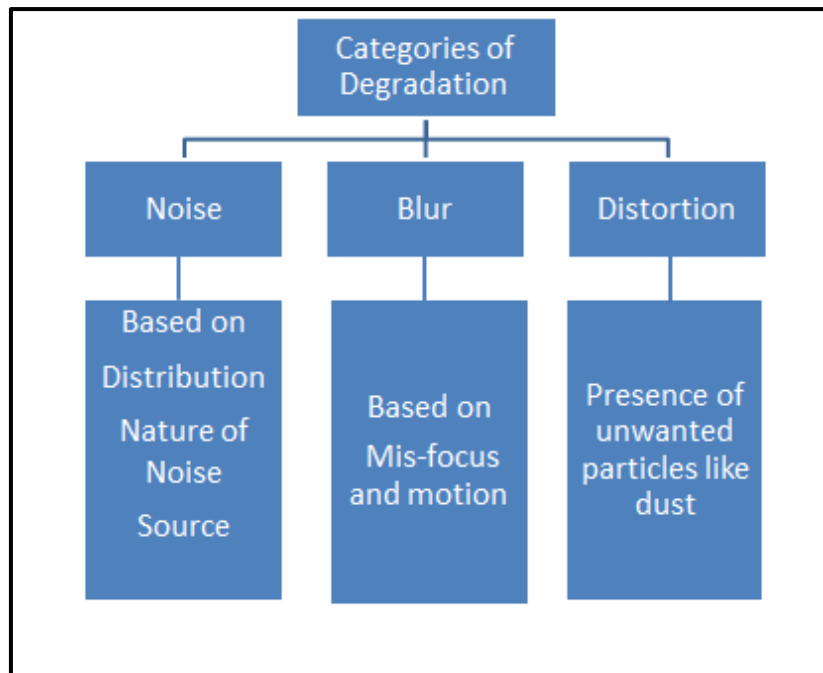


Figure 2.4: Categories of Degradation

2.3.4 Color Image Processing

Color image processing has gained importance because of the significant increase in the use of digital images over the internet. The use of color image processing is encouraged by two factors. Color helps in object identification and extraction from images. The other factor is that human eye can distinguish thousands of color shades and intensities and only few shades of gray. This information can be used to simplify image analysis. We can use this for object identification and feature extraction based on color.

Three quantities are used to describe any particular color. They are Hue, Saturation and Intensity.

2.3.4.1 Hue

It represents the dominant color as perceived by the observer. It is the property associated with the color which has dominant wavelength in a mix of light waves.

2.3.4.2 Saturation

It is determined by the excitation property, it gives the information about the purity or the amount of white color mixed with hue. A pure hue is fully saturated i.e. no white light is

mixed in it. The degree of saturation and the amount of white color added in it are inversely proportional to each other.

Hue and Saturation combined together are termed as Chromaticity and hence all colors are characterized by their brightness and chromaticity.

2.3.4.3 Intensity

It is determined by the actual amount of light present. More light corresponds to more intense colors. It is determined by the energy therefore it is physical quantity.

2.3.4.4 Radiance

It is the net amount of energy that comes out from a light source.

2.3.4.5 Luminance

It is the amount of energy an observer perceives from a light source.

2.3.4.6 Brightness

It is a subjective signifier and practically it is not possible to measure the brightness. It is determined by the perception of the color and therefore is psychological.

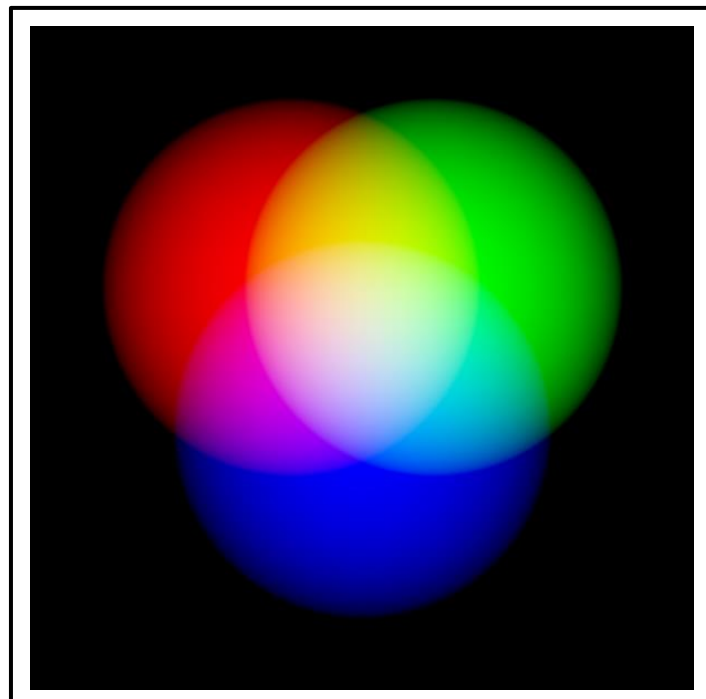


Figure 2.5: RGB Image Components

2.3.5 Image Coding and Compression

Image coding deal with how the image data can be represented.

Image Compression deals with reducing the amount of data required to represent an image and to enable efficient image storing and transmission. This can be done by removing the redundant information. There are various techniques that can be used for image compression like Huffman Coding, Shannon Fano coding, Arithmetic Coding, Lempel-Ziv-Welch coding for Error free or lossless Compression and, Lossy predictive coding for Lossy Compression.

2.3.6 Image Segmentation

Image Segmentation is a process of subdividing the image into multiple parts. The image is divided into constituent regions or objects so that the minute details of the image can be read and analyzed. In it the image is changed and the representation of the image is modified into meaningful and simplified way. This process comes into picture when we want to detect and identify a certain object in an image and when the object is identified the process is terminated. There is no need to continue the process past the level of detail required to identify the object.

There are many applications of image segmentation. It is used for Object location in satellite images, Diagnosis, for study of anatomical structure, military imaging and many more. There are many different ways to perform image segmentation such as Otsu's Method, Watershed Transformation.

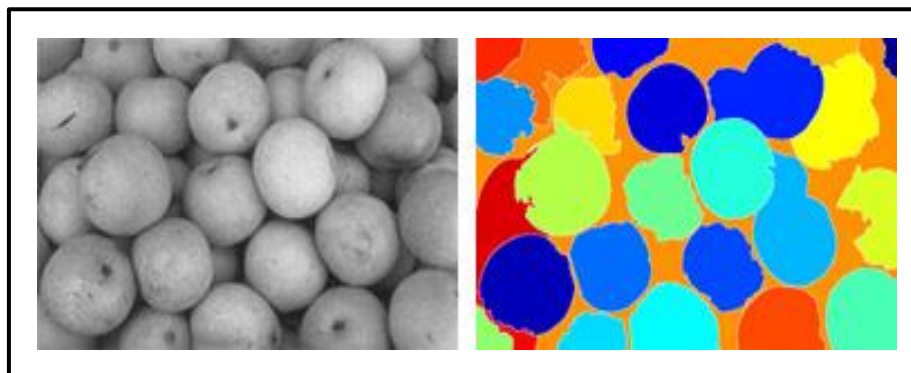


Figure 2.6: Example of Image Segmentation

(www.mathworks.in)

2.3.6.1 Thresholding

Thresholding is one of the important applications of Image segmentation because of its intuitive property and ease of implementation. In this method we try to find the object from its background. It is done on the basis of intensity of the pixels. In a normal way we segment such regions through separation of light and dark regions and this is called thresholding. After doing this a binary image is created from grey level by bending pixels below some threshold level to zero and other pixels which are above the grey level to one. There are various techniques of thresholding, but the simplest of them all is global thresholding where image histogram is partition by using a single global threshold T .

The figure given below shows the result of thresholding.



Figure 2.7: Example of Global Thresholding

(www.mathworks.in)

2.3.7 Representation and Description

The output of image segmentation stage is usually a raw pixel data, constituting either the boundary of a region i.e. the set of pixels separating on image region from another or all the points in the region itself. Converting the data to form suitable for computer processing is necessary. The decision to be made is whether the data should be represented as a boundary or as a complete region. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. A method must also be specified for describing the data so that features of interest are highlighted.

Description is also called feature selection, deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another.

2.3.8 Recognition

It is in which the object in the image is identified and labeled based on its description. This concept is used in many applications like surveillance systems, toll booth monitoring etc.

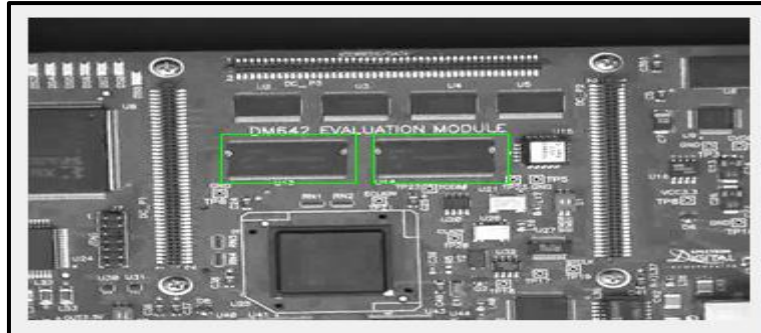


Figure 2.8: Recognition of specific chips on Circuit Board

Chapter 3

Introduction to MATLAB

MATLAB is an integrated technical computing environment that combines numeric computation, advanced graphics and visualization, and a high- level programming language.



Figure 3.1: MATLAB Logo

(www.mathworks.com/products/matlab)

3.1 Introduction

MATLAB stands for “MATrix LABoratory”. It is a high performance language for technical computing. It integrates computation, visualization and programming environment. It has data structures, contains built-in editing and debugging tools, and supports object-oriented programming. We can use it to graph functions, solve equations, perform statistical tests, and do much more operations. Its performs its tasks faster than traditional programming languages such as C, C++, and Fortran.

MATLAB has a wide range of applications. It can be used for signal processing, control design, finance modeling and analysis, biological computation, mathematical computation and last but not the least Image Processing.

It has many features at the disposal of the user. The tool boxes provided in it are one of the most important features of MATLAB. They allow the user to learn and apply many specialized technology hence making it easy for the user to directly use inbuilt codes rather than making them. Toolboxes are available for the area: Control System, Image Processing, Fuzzy Logic and many more.

3.2 Advantages of MATLAB

- a) It is user Friendly. The programs can be written and easily modified with built in integrated development and can be debugged using MATLAB Debugger.
- b) It is Platform Independent. It can run on many computer operating systems like Windows, Linux, Mac OS etc.
- c) Presence of Predefined Functions in it. MATLAB has a large number of predefined functions which can be directly used. They give tested and pre-packaged solutions to many basic technical problems such as square root, trigonometric functions of cosine and sine, mean, return absolute values etc. and to solve more complex problems there are the toolboxes.
- d) The feature of Device-Independent Plotting in MATLAB. It has integral plotting and imaging commands which cannot be found in other computer languages.
- e) The Graphical User Interface, MATLAB comprise of a tool that permit the user to construct a GUI. It is a powerful mechanism such that even a beginner can design a data analysis program.
- f) The MATLAB Compiler. In it firstly the program is compiled into a device independent p-code after that the p-code is interpreted at runtime.

3.3 Disadvantages of MATLAB

MATLAB does not have many disadvantages except one. The execution of the program takes more time than the compiled language. The reason behind it is that it is an interpreted language. MATLAB is costly software. Its cost is more than C or Fortran compiler, but the student version of MATLAB is inexpensive and is same as the full edition of MATLAB.

3.4 Some features of MATLAB

The following screen opens when we run the MATLAB software.

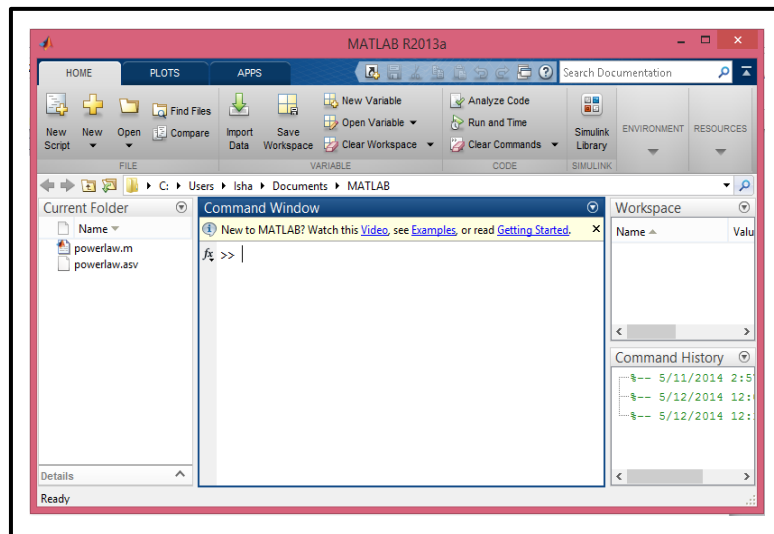


Figure 3.2: MATLAB Window

This screen shows the Command window, Workspace window and the command history window.

MATLAB Window

- Command Window: In it we issue command to MATLAB for processing.
- Command History: it shows the running history of prior commands issued in the command window.
- Workspace: it is the GUI for viewing, loading and saving MATLAB variables.
- Help: it is the GUI for finding and viewing on-line documentation. Help not only contains the theoretical background, but also shows demos for implementation.
- Editor: it is the text editor and debugger for MATLAB text files.

3.5 MATLAB for Image Processing

MATLAB is one of the best software's for Image processing. It stores images in the form of a two-dimensional matrix. In the matrix each element represents a single pixel of an image. The true color images use three dimensional matrixes. In it the first plane represents intensity

of red pixels, second plane represents the intensity of green and the third plane represents the intensity of blue pixels.

Image formats supported by MATLAB:

- a) BMP(Bitmap)
- b) HDF(Hubble Deep Field)
- c) JPEG(Joint Picture Expert Group)
- d) PCX(Picture Exchange Format)
- e) TIFF(Tagged Exchange Format)

MATLAB supports four types of images. They are:

3.5.1 Binary Images

In a binary image each pixel is represented as black and white. Since there are only two possible values for each pixel 0 and 1, we only need one bit per pixel.

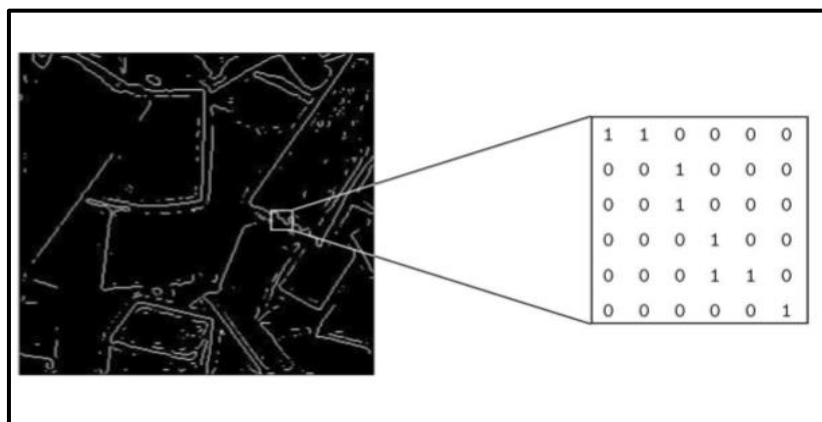


Figure 3.3: Binary Image
(haris@geod.baug.ethz.ch)

3.5.2 Grayscale Images

In this the pixels are in a shade of gray. They range from 0 (black) to 255 (white). This means that each pixel can be represented by eight bits or exactly one byte.

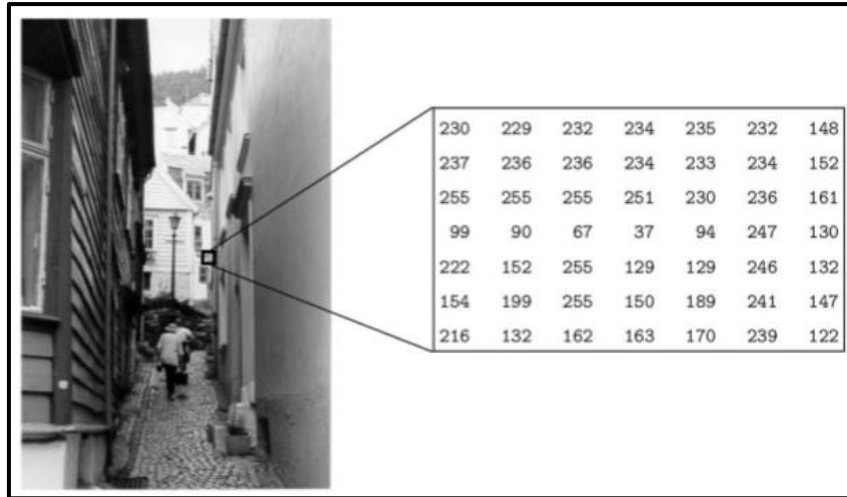


Figure 3.4: Grayscale Image
(haris@geod.baug.ethz.ch)

3.5.3 Truecolor Images or RGB

Here each pixel has a particular color and that color is described by the amount of red, blue and green present in it. Each of these components has a range from 0 to 255, which gives a total of cube of (256) different possible colors. Such an image is a three dimensional image and each pixel corresponds to three values.



Figure 3.5: Truecolor Image
(haris@geod.baug.ethz.ch)

3.5.4 Indexed images

These are also called as pseudo-color images. They comprise of an array and color map matrix. The pixel values in the array are direct indices into a color map. Array is represented as X and color map which is an array of m -by-3 with a floating point in the range of $[0\ 1]$ is represented as map . The m -rows of color map give the information about the values of components (red, green and blue) of a single color. An indexed image uses direct mapping of pixel values to color map values.

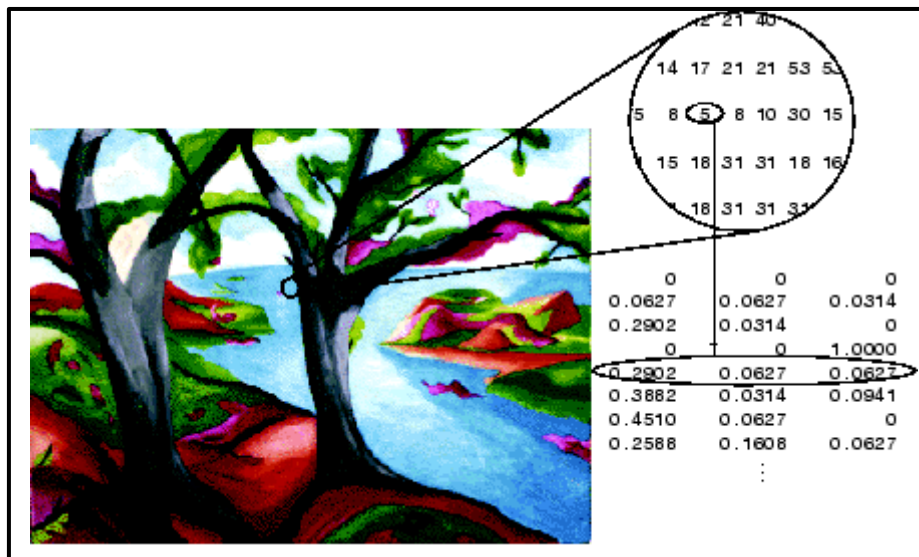


Figure 3.6: Indexed Image

(www.mathworks.in)

3.6 General Commands in MATLAB

These commands are related to dealing with images in MATLAB.

1. `imread`: It reads an image.
2. `figure`: It creates a figure on the screen on which we can display the image.
3. `imshow(g)`: This command displays the matrix g as an image.
4. `pixval on`: This turns on the pixel values in our figure.
5. `impixel(i,j)`: This command returns the value of the pixel at the position (i,j) .
6. `iminfo`: This returns the information about the image.
7. `ind2gray(x,map)`: This converts indexed image to grayscale.

8. `gray2ind(x)`: This converts grayscale to indexed image.
9. `rgb2gray(x)`: This converts RGB to grayscale.
10. `gray2rgb(x)`: This converts gray to RGB.
11. `rgb2ind=[y,map]`: This converts RGB to Indexed image.
12. `ind2rgb(x,map)`: This command converts indexed to RGB image.

Chapter 4

Algorithms

4.1 Face Detection

Face detection is defined as finding the number of face present in an image and locating their position and size. Generally the process of face detection is partition in two steps. They are

- In the first step the whole image is scanned to find the region (features) that can be identified as face, skin colour is the most common among all.
- And the next is localisation which gives more accurate estimation of size and position of the faces. Face detection is the first step of human identification and recognition system

4.2 Face Recognition

We do face recognition almost on a daily basis. Most of the time we look at a face and are able to recognize it instantaneously, if we are already familiar with that face. This natural ability, if possible imitated by machines can prove to be invaluable and may provide for very important in real life applications. Face Recognition is the task of identifying an already detected face as a known or unknown face, and in more advanced cases, telling exactly who's it is!

- Firstly, the face of the person is detected in the image and then stored in the database.
- After storing the image, the face is matched with the others present in the database and the closest resemblance is resulted.

4.3 Algorithms used:

4.3.1 Viola Jones Face Detection Algorithm

The basic principle behind the Viola-Jones algorithm is to scan a sub-window capable of detecting faces across a given image. The standard image processing approach would be to rescale the given image to different sizes and then run the fixed size detector through these images. Viola-Jones re-scale the detector instead of the input image and run the detector many times through the image, each time with a different size. This detector is constructed

using a so-called integral image and some simple rectangular features reminiscent of Haar wavelets.

4.3.1.1 The Scale Invariant Detector

The first step of the Viola-Jones face detection algorithm is to turn the input image into an integral image. This is done by making each pixel equal to the entire sum of all pixels above and to the left of the concerned pixel.

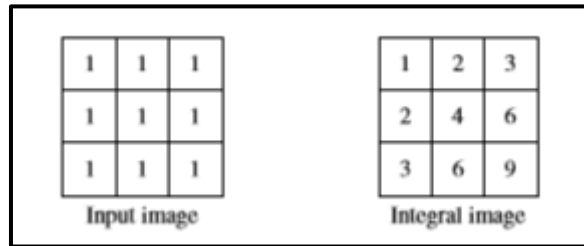


Figure 4.1 Integral Image

This allows for the calculation of the sum of all pixels inside any given rectangle using only four values. These values are the pixels in the integral image that coincide with the corners of the rectangle in the input image.

$$\text{Sum of grey rectangle} = D - (B + C) + A$$

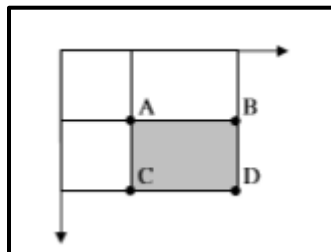


Figure 4.2: Sum Calculation

Because both rectangle B and C include rectangle A the sum of A has to be added to the calculation. The sum of pixels within rectangles of arbitrary size can be calculated in constant time.

4.3.1.2 Feature Selection

The Viola-Jones face detector analyzes a given sub-window using features consisting of two or more rectangles. For a 24x24 detection region, the number of possible rectangle features is approximately 180,000.

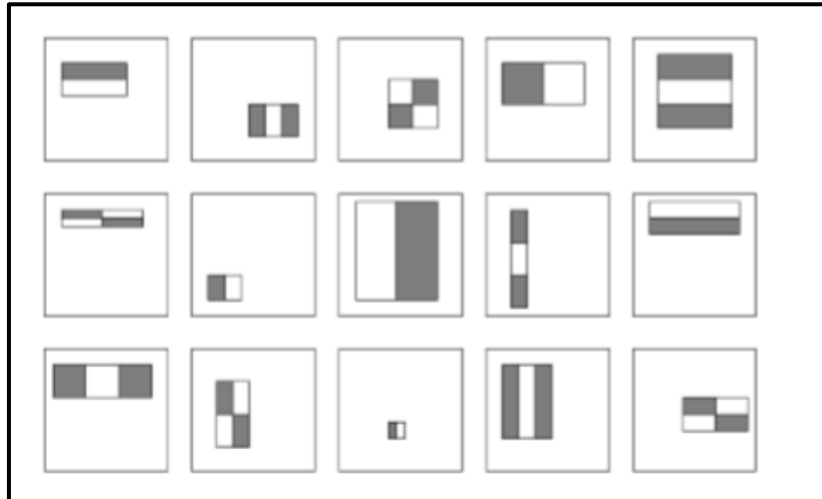


Figure 4.3: Types of features

Each feature results in a single value which is calculated by subtracting the sum of the white rectangles from the sum of the black rectangles.

At test time, it is impractical to evaluate the entire feature set. Therefore, we can create a good classifier using just a small subset of all possible features.

4.3.1.3 AdaBoost

AdaBoost is a machine learning boosting algorithm capable of constructing a strong classifier through a weighted combination of weak classifiers. A weak classifier classifies correctly in only a little bit more than half the cases. To match this terminology to the presented theory each feature is considered to be a potential weak classifier. A weak classifier is mathematically described as:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } pf(x) > p\theta \\ 0 & \text{otherwise} \end{cases}$$

- x is a 24*24 pixel sub-window
- f is the applied feature
- p the polarity

- θ is the threshold that decides whether x should be classified as a positive (a face) or a negative (a non-face)

Only small amount of the possible 180,000 feature values are expected to be potential weak classifiers, the AdaBoost algorithm is used to select only the best feature polarity and threshold.

4.3.1.4 Cascading Classifiers

Starting with the simple classifiers which reject many of the negative sub-windows while detecting almost all positive sub-windows. The basic principle behind the algorithm is not to find faces but to discard non-face region. Positive results from the first classifier trigger the evaluation of a second classifier, and so on. A negative outcome at any point leads to the immediate rejection of the sub-window i.e. the non-face region as shown.



Figure 4.4: Working of Cascading Classifier

4.3.1.5 Result of Face Detection

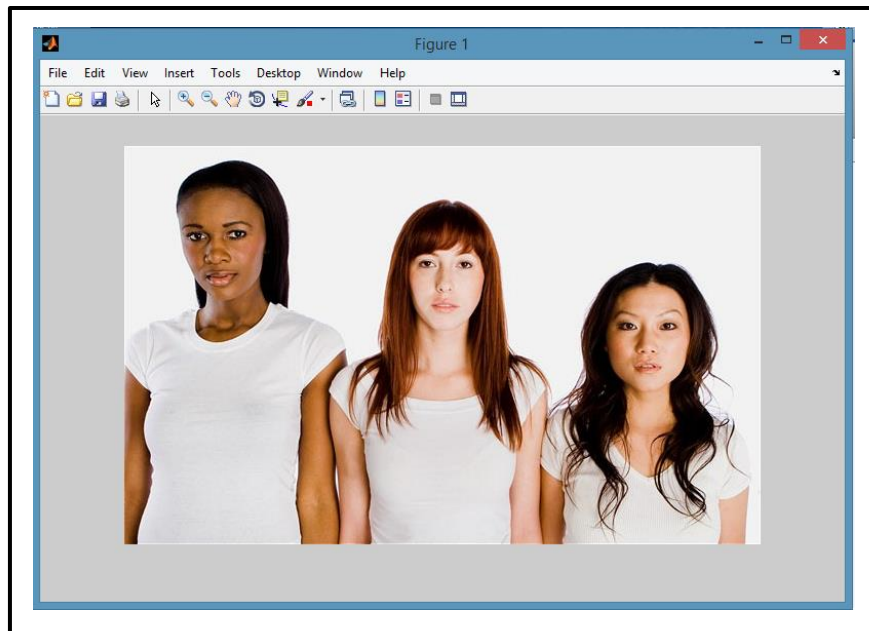


Figure 4.5: Original Image

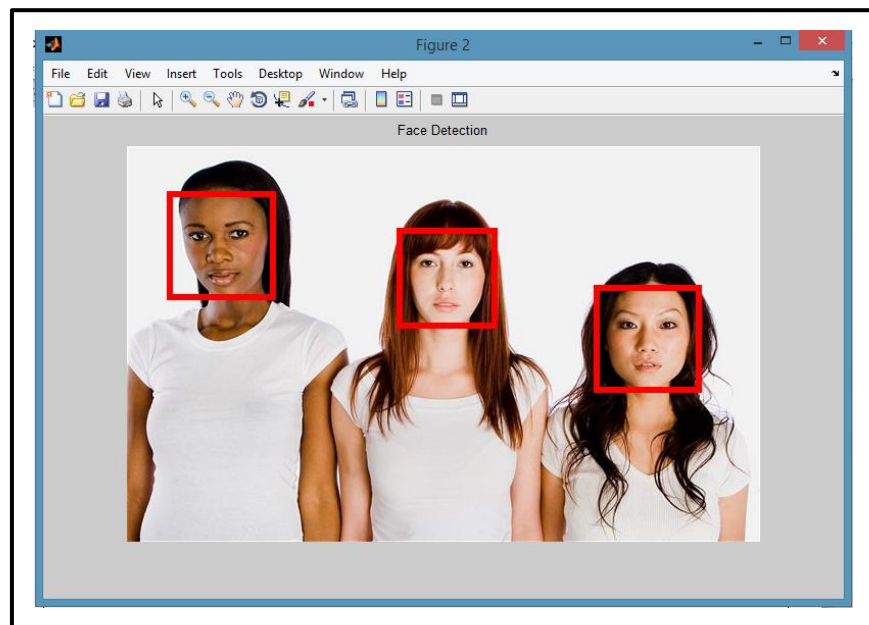


Figure 4.6: After face detection

4.3.2 Face Recognition using PCA

4.3.2.1 PCA

PCA is a variable reduction procedure and is useful if the obtained data has some redundancy. This will result into reduction of variables into smaller number of variables which are called Principal Components which will account for most of the variance in the observed variable.

The main goal of PCA is to reduce the dimensionality of the data by retaining as much as variation possible in our original data set. On the other hand dimensionality reduction results in information loss. The low-dimensional space can be determined best by principal components.

The major advantage of PCA is using it in Eigen face approach which helps in reducing the size of the database for recognition of a test image. The images are stored as their feature vectors in the database which can be seen projecting each and every trained image to the set of Eigen faces obtained. PCA is applied on Eigen face approach to reduce the dimensionality of a large data set.

4.3.2.2 Eigen Face Approach

It is an adequate and efficient method to be used in face recognition due to its simplicity, speed and learning capability. Eigen faces are a set of Eigen vectors which are used in the Computer Vision problem of human face recognition. They refer to an appearance based approach to face recognition that captures the variation in a collection of face images and use this information to encode and compare images of individual faces in a holistic manner.

The Eigen faces are Principal Components of distribution of faces, i.e. the Eigen vectors of the covariance matrix of the set of the face images, where an image with N by N pixels is considered a point in N^2 dimensional space. We want to extract the relevant information in a face image, encode it efficiently and compare one face encoding with a database of faces encoded similarly. A simple approach to extracting the information content in an image of a face is to capture the variation in a collection of face images.

We want to find Principal Components of the distribution of faces, or the Eigenvectors of the covariance matrix of the set of face images. Each Eigen vector is contributed by each image location, so that we can display the Eigen vector as a face. Each face image can be

represented in terms of linear combination of the Eigen faces. The total number of possible Eigen faces is equal to the number of face images in the training set. The faces can also be approximated by using the best Eigen face, which is the one having the largest Eigen values, and which therefore account for most variance between the set of face images. The primary reason for using fewer Eigen faces is computational efficiency.

4.3.2.3 Eigen Values and Eigen Vectors

In linear algebra, the eigenvectors of a linear operator are non-zero vectors, when they are operated by the operator, they result in a scalar multiple of them. Scalar is then called Eigen value (λ) which is associated with the eigenvector (X). Eigen vector is a vector that is scaled by linear transformation. It is a property of matrix. When a matrix acts on it, only the vector magnitude is changed not the direction.

$$AX = \lambda X, \text{ where } A \text{ is a vector function.}$$

$$(A - \lambda I)X = 0, \text{ where } I \text{ is the identity matrix.}$$

This is a homogeneous system of equations and form fundamental linear algebra. A non-trivial solution exists if and only if-

$$\text{Det}(A - \lambda I) = 0, \text{ where } \text{det} \text{ denotes determinant.}$$

When evaluated, it becomes a polynomial of degree n. This is called characteristic polynomial of A. If A is N by N matrix, then there are n solutions or n roots of the characteristic polynomial. Therefore, there are n Eigen values of A satisfying the equation.

$$AX_i = \lambda_i X_i, \text{ where } i = 1, 2, 3, \dots, n$$

If the Eigen values are all distinct, then there are n associated linearly independent eigenvectors, whose directions are unique and span an n dimensional Euclidean space.

4.3.2.4 Face Image Representation

Training set of m images of size $N \times N$ are represented by vectors of size N^2 .

Each face is represented by $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M$. Feature vector of a face is stored in a $N \times N$ matrix. Now, these two dimensional vectors are changed to one dimensional vector.

For Example- $\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$

Each face image is represented by the vector Γ_i .

$$\Gamma_1 = \begin{bmatrix} 1 \\ -2 \\ 1 \\ -3 \end{bmatrix} \Gamma_2 = \begin{bmatrix} 1 \\ 3 \\ -1 \\ 2 \end{bmatrix} \Gamma_3 = \begin{bmatrix} 2 \\ 1 \\ -2 \\ 3 \end{bmatrix} \dots \dots \dots \Gamma_M \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

4.3.2.5 Mean and Mean Centered Images

Average face image is calculates by

$$\psi = (1/M)\sum_{i=1}^M \Gamma_i$$

$$\begin{bmatrix} 1 \\ -2 \\ 1 \\ -3 \end{bmatrix} + \begin{bmatrix} 1 \\ 3 \\ -1 \\ 2 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ -2 \\ 3 \end{bmatrix} + \dots \dots \dots + \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} -1 \\ -1 \\ 2 \\ -3 \end{bmatrix}$$

$$\Psi = (\Gamma_1 + \Gamma_2 + \Gamma_3 + \dots \dots \dots + \Gamma_M)/M$$

Each face differs from the average by $\Phi_i = \Gamma_i - \Psi$ which is called mean centered image.

$$\Phi_1 = \begin{bmatrix} 2 \\ -1 \\ -1 \\ 0 \end{bmatrix} \Phi_2 = \begin{bmatrix} 2 \\ 4 \\ -3 \\ 5 \end{bmatrix} \Phi_3 = \begin{bmatrix} 3 \\ 2 \\ -4 \\ 6 \end{bmatrix} \dots \dots \dots \Phi_M = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 4 \end{bmatrix}$$

4.3.2.6 Covariance Matrix

A covariance matrix is constructed as:

$$C = AA^T, \text{ where } A = [\Phi_1, \Phi_2, \Phi_M] \text{ of size } N^2 \times N^2.$$

$$A = \begin{bmatrix} 2 & 3 \\ -1 & -2 \\ -1 & 1 \\ 0 & 2 \end{bmatrix} A^T = \begin{bmatrix} 2 & -1 & -1 & 0 \\ 3 & -2 & 1 & 2 \end{bmatrix}$$

Size of covariance matrix will be $N^2 \times N^2$

Eigen vectors corresponding to this covariance matrix is needed to be calculated, but that will be a tedious task therefore, for simplicity, we calculate $A^T A$ which would be a 2×2 matrix in this case.

$$A^T A = \begin{bmatrix} 6 & 7 \\ 7 & 18 \end{bmatrix}, \text{ size of this matrix is } M \times M.$$

Consider the eigenvectors v_i of $A^T A$ such that

$$A^T A X_i = \lambda_i X_i$$

The eigenvectors v_i of $A^T A$ are X_1 and X_2 which are 2×1 . Now multiplying the above equation with A both sides we get-

$$A^T A X_i = A \lambda_i X_i$$

$$A^T A (A X_i) = \lambda_i (A X_i)$$

Eigen vectors corresponding to $A A^T$ can now be easily calculated with reduced dimensionality where $A X_i$ is the Eigen vector and λ_i is the Eigen value.

4.3.2.7 Eigen Face Space

The Eigen vectors of the covariance matrix $A A^T$ are $A X_i$ and are denoted by U^i . U^i resembles the facial images which look ghostly and are called Eigen faces. Eigen vectors correspond to each Eigen face in the face space and discard the faces for which Eigen values are zero thereby, reducing the Eigen face space to an extent. Then the Eigen faces are ranked according to their usefulness in characterizing the variation among the images. A face image can be projected into this face space by

$$\Omega_k = U^T (\Gamma_k - \psi); \text{ Where } k=1, \dots, M, \text{ and } (\Gamma_k - \psi) \text{ is the mean centered image.}$$

Hence projection of each image can be obtained as Ω_1 for projection of first image and Ω_2 for projection of second image.

4.3.2.8 Recognition Step

The test image, Γ , is projected into the face space to obtain a vector, Ω as

$$\Omega = U^T (\Gamma - \psi)$$

The distance of Ω to each face is called Euclidean distance and defined by

$\epsilon_k^2 = \|\Omega - \Omega_k\|^2$ where $k=1, \dots, M$ and Ω_k is a vector describing the k th face class. A face is classified as belonging to class k when the minimum ϵ_k is below some chosen threshold Θ_c . otherwise the face is classified as unknown.

Θ_c , is half the largest distance between any two face images:

$$\Theta_c = (1/2 \max_{j,k} \|\Omega_j - \Omega_k\|) \text{ where } j, k = 1, \dots, M$$

We have to find the distance ϵ between the original test image Γ and its reconstructed image from the Eigen face Γ_f .

$$\epsilon^2 = \|\Gamma - \Gamma_f\|^2, \text{ where } \Gamma_f = U * \Omega + \psi$$

If $\epsilon \geq \Theta_c$ then input image is not even a face image and not recognized.

If $\epsilon < \Theta_c$ and $\epsilon_k \geq \Theta$ for all k then input image is a face image but it is recognized as an unknown face.

If $\epsilon < \Theta_c$ and $\epsilon_k < \Theta$ for all the values of k then input images are the individual face images associated with the class vector Ω_k .

4.3.2.9 Result of Face Recognition

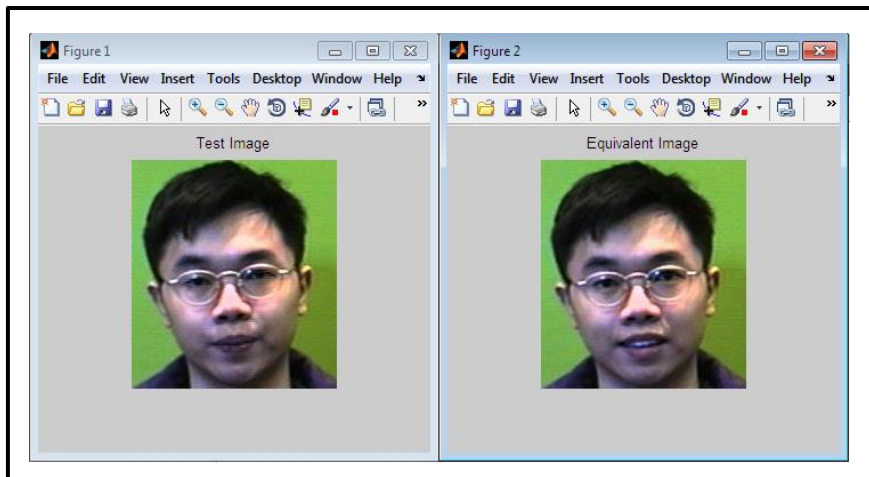


Figure 4.7: Result 1

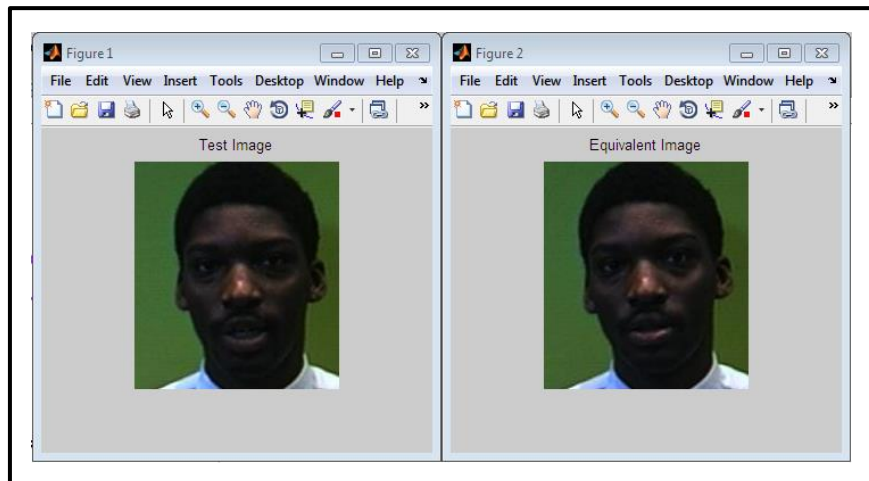


Figure 4.8: Result 2

Chapter 5

Simulations and Results

5.1 Block Diagram

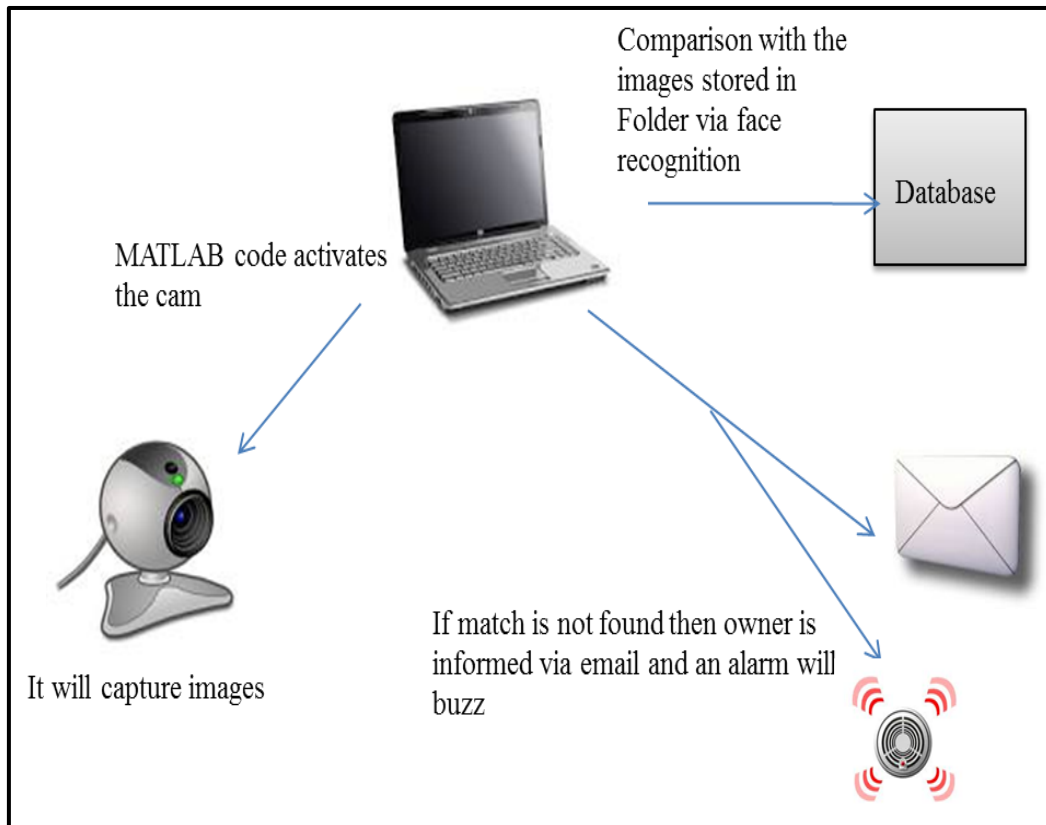


Figure 5.1: Block Diagram of Project

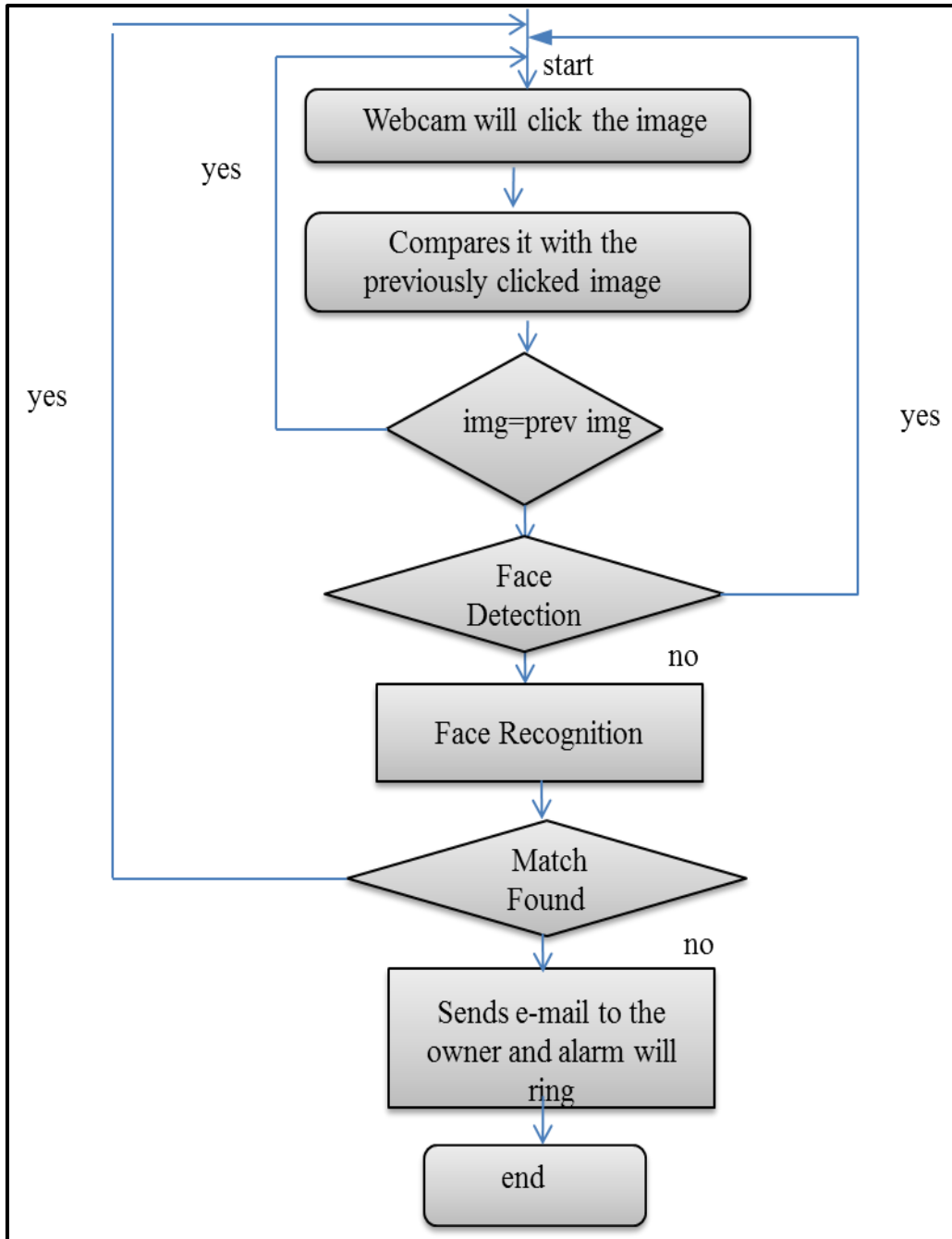


Figure 5.2: Flow Chart

5.2 Activation of webcam

The following commands in MATLAB are used to activate the webcam for capturing of the images.

```
obj= videoinput(adaptorname,deviceid);  
videoinput('winvideo');
```

The winvideo Adaptor Works with Devices That Provide Input/Output Capabilities.

The device id is the code given to the camera which we want to activate.

The command for connecting the webcam to the computer/laptop through USB.

```
vid=videoinput('winvideo',1,'RGB24_320*240');
```

Specify the adaptor name and resolution.

5.3 Capturing Image:

GETSNAPSHOT -Immediately return a single image snapshot.

```
frame = getsnapshot(obj)
```

Immediately returns one single image frame.

This command will store the image of that instant into the variable data in a matrix of resolution as defined by the user.

obj must be a 1x1 video input object.

FRAME is returned as a H-by-W-by-B matrix, where

H is the Image height, as specified in the ROIPosition property.

W is the Image width, as specified in the ROIPosition property.

B is the Number of color bands, as specified in the NumberOfBands property

5.4 Conversion

The command for conversion is as follows

```
rgb =ycbcr2rgb(ycbcr)
```

It converts the YCbCr image to the equivalent true color image RGB.

The image captured is in YCbCr mode. This color space has been defined to meet the increasing demand of digital algorithms in handling video information and has become the

widely used color space in digital videos. It has three components, two of them is of chrominance and one is of luminance.

YCbCr color model was developed in order to allow transmitting color information onto the television sets keeping in mind that existing black and white television sets continue to display the images in tones of grey. And get the characteristic of isolating luminance and color information, thus used in many applications such as compression.

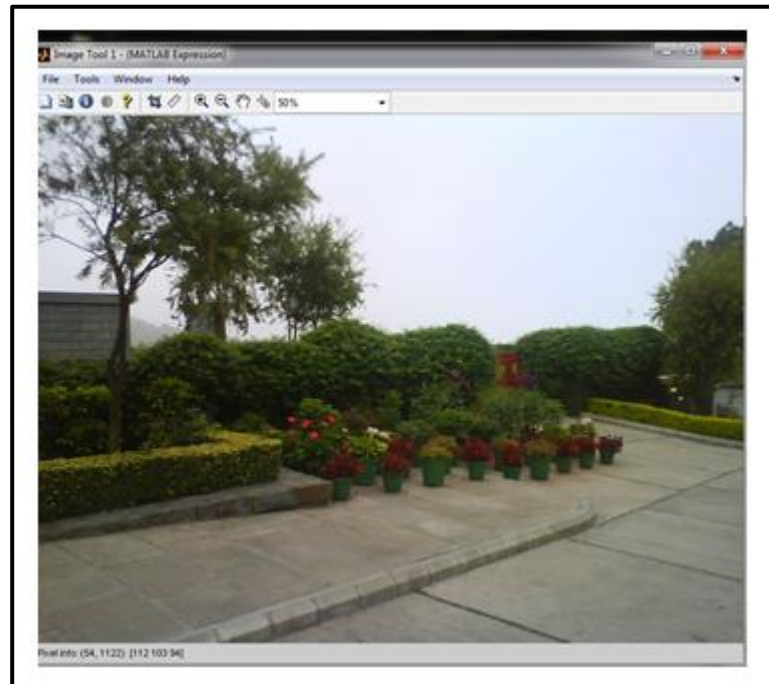


Figure 5.3: RGB Mode

An RGB image, sometimes referred to as a "truecolor" image, is stored as an m-by-n-by-3 data array that defines red, green, and blue color components for each individual pixel. The color of each pixel is determined by the combination of the red, green, and blue intensities stored in each color plane at the pixel's location. Graphics file formats store RGB images as 24-bit images, where the red, green, and blue components are 8 bits each.

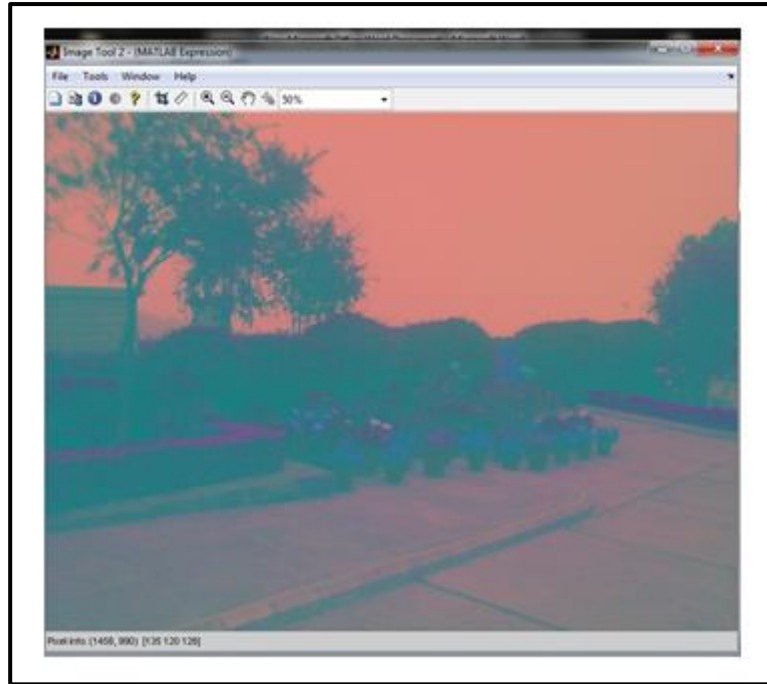


Figure 5.4: YCbCr Mode

After converting to RGB we convert the image into black and white image using the following command.

$$bw = im2bw(i,level)$$

It converts the intensity image I to black and white. The output image BW replaces all pixels in the input image with luminance greater than level with the value 1 (white) and replaces all other pixels with the value 0 (black). You specify level in the range [0,1], regardless of the class of the input image.

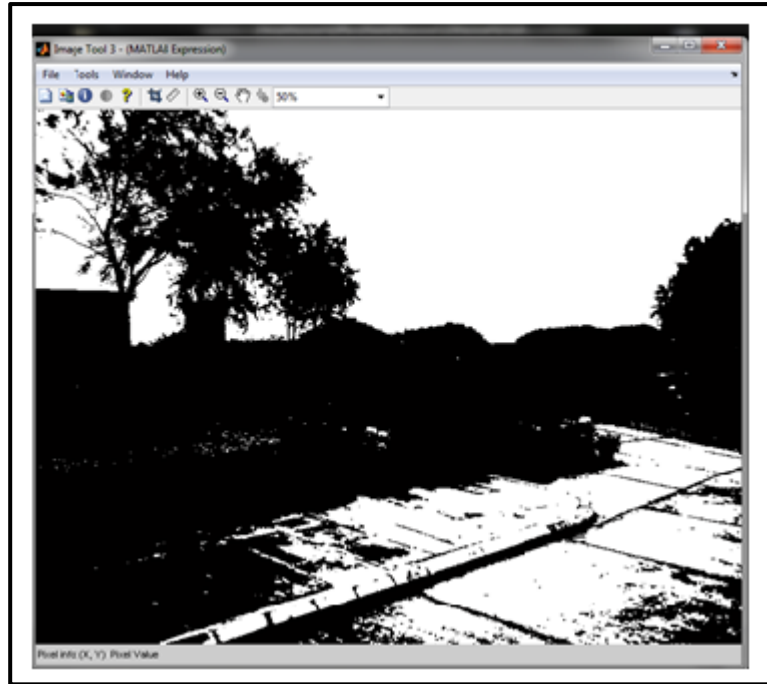


Figure 5.5: Black and White Mode

5.5 Comparison

Comparison is based on a simple technique where after converting the image to black and white we count the high value pixels and low value pixels separately.

These values of both the images are counted and if the images are same then these values will be same or will have negligible difference caused by illumination which can be ignored.

If the values differ then the images are not same i.e. some change is detected in the surrounding.

5.6 Result of Comparison

If changes are detected in the surroundings then the image is further analyzed to detect face in it.

5.7 Face Detection

Viola Jones algorithm is used to detect any face in the image.

FDetect = vision.CascadeObjectDetector

The cascade object detector uses the Viola-Jones algorithm to detect people's faces, noses, eyes, mouth, or upper body.

5.8 Face Recognition

The given function

[m, A, Eigenfaces] = EigenfaceCore(T)

It uses Principle Component Analysis (PCA) to determine the most discriminating features between images of faces.

This function gets a 2D matrix, containing all training image vectors and returns 3 outputs which are extracted from training database

5.9 Result

If face recognition gives a positive result then it will keep on clicking the images, otherwise it will send an e-mail to the user and an alarm will set off.

Chapter 6

Applications

6.1 Security Issues

The main goal of the project is to act as a security device. It can successfully detect an intrusion and can inform the user about the intrusion immediately.

- User Friendly

It is user friendly. It is easy to handle as one have to just run the MATLAB code. Various changes can be made by the user according to his need.

6.2 Face Detection and Recognition

With the use of Face Detection and Recognition technology, it can be used for

- Biometric Identification

As a biometric it derives a number of advantages from being the primary biometric that humans use to recognise one another. It has become more beneficial than fingerprint recognition systems as fingerprint can be forged.

- Surveillance Systems

The Surveillance systems rely on passive acquisition by capturing the face image without the cooperation or knowledge of the person being imaged.

- Access Control

Access control for various private data, especially which is stored in electronic devices like computers or laptops and smart phones. It has increased the ease of use over password protection.

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

On implementing the system for intruder detection, we can assert with confidence that such system can be used to detect an intrusion efficiently. This software can be used to detect an intrusion at various places like in a shop or at home (when no one is at home). A part of this project that is Face Detection and Recognition, they have numerous applications. The algorithms used in this project are uniformly accepted and recognised and are used for practical implementations of such systems. As mis-happenings like thefts are increasing day by day so demand of such a system is also increasing. This software can be used by the common people without any difficulties because it is user friendly.

There is always a need to modify and add features according to the growing technology. We know there is profitability for modifying the technology to compete the market. Even though there are certain disadvantages and limitations to these systems due to lack of funds and limited scope and time available to us, they can be overcome by making certain changes and enhancements to our current system.

7.2 Future Scope

There are certain enhancements which would increase the efficiency of these intruder detection systems. Firstly, the accuracy of these systems can be increased by using high resolution cameras. This would make the system more reliable and can be used in any conditions. Secondly, we can use various other Face Recognition/Detection algorithms to improve its accuracy. We can expand our database to use it on a bigger platform. These additions would make the system more complete and more capable of being able to work in any kind of environment.

References

1. Meftah Ur Rahman, Dept. of Computer Science, George Mason University; A comparative study on face recognition techniques and neural network, Available at: mrahma12@masonlive.gmu.edu.,(2011).
2. Sanjay Kr. Singh, D. S. Chauhan, MayankVatsa, Richa Singh; A Robust Skin Color based Face Detection Algorithm, Tamkang Journal of Science and Engineering, Vol. 6, No. 4, pp. 227-234 (2003).
3. Devendra Singh Raghuvanshi, Dheeraj Agrawal; Human Face Detection by using Skin Color Segmentation, Face Features and Regions Properties, International Journal of Computer Applications (0975 – 8887) Volume 38– No.9, January 2012..
4. DaggiVenkateshwar Rao, ShrutiPatil, Naveen Anne Babu and V Muthukumar; Implementation and Evaluation of Image Processing Algorithms on Reconfigurable Architecture using C-based Hardware Descriptive Languages, International Journal of Theoretical and Applied Computer Sciences Volume 1 Number 1 , pp. 9–34, (2006) .
5. Ravi Kumar A.V, Nataraj K.R; Result Analysis of LabVIEW and MatLab in Application of Image Edge Detection, International Journal of Computer Applications (0975 – 888) Volume 48– No.9, June 2012.
6. Paul Viola ,Michael J. Jones; Robust Real-Time Face Detection, International Journal of Computer Vision 57(2), pp-137–154, 2004.
7. Soriano, Huovinen S; Using the Skin locus to cope with changing illumination conditions in color- based facetracking, IEEENordicSignalProcessing Symposium, Kolmarden, Sweden. pp. 383-386, 2000.
8. Q. Xiaofang, W. Xiaojun, W. Shitong, Yang Jing-Yu; An improved algorithm of face detection, Computer Applications, vol. 28, pp. 986-989, Apr.2008.
9. Nicolas Morizet, FrédéricAmiel, InsafDrisHamed, Thomas Ea, A Comparative Implementation of PCA Face Recognition Algorithm Department of Electronics InstitutSupérieur d'Electronique de Paris (I.S.E.P.) Paris, France).
10. Robin Strand, Centre for Image Analysis Swedish University of Agricultural Sciences Uppsala University; Image coding and compression.
11. HarisPapasaika-Hanusch Institute of Geodesy and Photogrammetry; Digital Image Processing Using Matlab, ETH Zurich, Available at haris@geod.baug.ethz.ch
12. The MathWorks, Inc.. (2013). MATLAB - The Language Of Technical Computing.