

INTELLIGENT SECURITY SYSTEM FOR CONFIDENTIAL AREAS

Dissertation submitted in fulfilment of the requirements for the Degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Praful Sharma (121051)

Vaibhav Jain (121065)

Gaurav Mehan (121098)

UNDER THE GUIDANCE OF

Ms Pragya Gupta



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT SOLAN
– 173 234, HIMACHAL PRADESH INDIA

June- 2016

TABLE OF CONTENTS

Chapter No.	Topics	Page No.
	INNER FIRST PAGE	ii-iii
	DECLARATION BY THE SCHOLAR	iv
	SUPERVISOR'S CERTIFICATE	v
	ACKNOWLEDGEMENT	vi
	LIST OF FIGURES	vii-viii
	ABSTRACT	ix
CHAPTER-1	INTRODUCTION	1-2
CHAPTER-2	HISTORY	3-4
CHAPTER-3	REVIEW/BACKGROUND MATERIAL	5
CHAPTER-4	WORK DESCRIPTION	6-7
	4.1 WORKING	6
	4.2 ALGORITHM	7
CHAPTER-5	IMAGE COMPARISON TECHNIQUES	8-18
	5.1 EDGE DETECTION TECHNIQUE	8
	5.2 BACKGROUND SUBTRACTION	10
	5.3 HISTOGRAM COMPARISON	14
	5.3.1 Comparing images using Histograms	17
	5.3.2 Applications of Histograms in Image Processing	17

CHAPTER-6	TECHNIQUE USED FOR IMAGE COMPARISON	19-20
CHAPTER-7	PROJECT WORK	21-24
7.1	CODE 1	21
7.2	CODE 2	21
7.3	CODE 3	23
CHAPTER-8	PROBLEMS FACED	25-27
8.1	PROBLEM 1	25
8.2	PROBLEM 2	27
CHAPTER-9	FUTURE SCOPE	28
CHAPTER-10	RESULTS	29-37
10.1	CONCLUSION	37
	REFERENCES	38

DECLARATION BY THE SCHOLAR


I hereby declare that the work reported in the B-Tech thesis entitled “**Intelligent Security System for Confidential Areas**” submitted at **Jaypee University of Information Technology, Waknaghat India**, is an authentic record of my work carried out under the supervision of **Ms Pragya Gupta**. I have not submitted this work elsewhere for any other degree or diploma.



Praful Sharma



Vaibhav Jain



Gaurav Mehan

Department of Electronics and Communication

Jaypee University of Information Technology, Waknaghat, India

Date:

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the B-Tech. thesis entitled “**Intelligent Security System for Confidential Areas**”, submitted by **Praful Sharma, Vaibhav Jain and Gaurav Mehan** at **Jaypee University of Information Technology, Wagnaghat, India** is a bonafide record of his / her original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.



Signature:

Name: Ms Pragya Gupta

Jaypee University of Information Technology Wagnaghat, solan (H.P)

Date: 27/05/2016

ACKNOWLEDGEMENT

It is our foremost duty to express our deep regards & gratitude to our Project Guide **Ms. Pragma Gupta**, under whose guidance & supervision we are able to work on the project *“Intelligent Security System for Confidential Areas”*. Any amount of gratitude towards our family and friends can never be enough for the constant support they have provided us with, we are totally indebted to them. At last it is him, to whom we owe everything, as without God’s blessings this endeavours of ours would never have been a success.

Thanking You.



Praful Sharma
(121051)



Vaibhav Jain
(121065)



Gaurav Mehan
(121098)

LIST OF FIGURES

Figure No.	Caption	Page No.
Figure 4.1	Flow Diagram	7
Figure 5.1	Original Image	10
Figure 5.2	Image after Edge detection	10
Figure 5.3	Original Image	12
Figure 5.4	Background Image	13
Figure 5.5	Image After Background Subtraction	13
Figure 5.6	Basic Image Types and their corresponding Histograms	16
Figure 5.7	Clicked Image	18
Figure 5.8	Histogram of Image	18
Figure 8.1	Image clicked using default Camera software of Windows 10	26
Figure 8.2	Image clicked using Camera but Interfaced with Matlab	26
Figure 8.3	Error regarding Camera use	27
Figure 10.1	Default Image	29
Figure 10.2	Histogram of default Image	29
Figure 10.3	Clicked Image1 (No intrusion)	30
Figure 10.4	Histogram of Clicked Image1 (No intrusion)	30
Figure 10.5	Clicked Image2 (Intrusion Present)	31
Figure 10.6	Histogram of Clicked Image2 (Intrusion Present)	31
Figure 10.7	Clicked Image3 (Intrusion Present)	32
Figure 10.8	Histogram of Clicked Image3 (Intrusion Present)	32

Figure 10.9	Clicked Image4 (No intrusion)	33
Figure 10.10	Histogram of Clicked Image4 (No intrusion)	33
Figure 10.11	Clicked Image5 (Intrusion Present)	34
Figure 10.12	Histogram of Clicked Image5 (Intrusion Present)	34
Figure 10.13	Clicked Image6 (Intrusion Present)	35
Figure 10.14	Histogram of Clicked Image6 (Intrusion Present)	35
Figure 10.15	Result of Image comparison	36

ABSTRACT

With increasing crime rates, relying solely on the protection of the law is not easy. With continued technological advancements, there is no reason not to look into additional security. A security system is a great way to provide extra protection for our homes or business. Although, the security systems which are being used these days help a lot in securing our homes and business, but still there are few drawbacks of this system.

One of the drawbacks is its unnecessary recording even when it is not required. Also this system is used as evidence after the incidence takes place but cannot help in avoiding the incidence. So, inspired by the importance of security system, we are working on this project and are trying to add some functionalities to this system.

To solve the problem of unnecessary recording, which ultimately leads to wastage of memory, we are using techniques of Digital Image Processing to intelligently record the video.

Also to avoid the incidence we are adding functionality of notifying a concerned authority regarding the presence of intrusion, which may help in avoiding some mishap. So our system would completely revolutionize the surveillance process by using much less space and would also eliminates the cumbersome process of keeping check on recorded data and may help in avoiding mishaps by notifications.

CHAPTER 1: INTRODUCTION

Every business owner strives to keep their employees, assets, and office space as safe as possible. Work is being done for years to build and maintain businesses, and when we leave, we want to make sure that everything is protected from harm. There are several ways to help increase the security at your business; one of the most effective is to install a security system.

Most buildings have several areas on the outside and inside that are great hiding places for intruders. Installing cameras inside can lower the amount of hiding places an intruder will have to hide in, as well as deter customers and/or employees from theft or other inappropriate work behaviour. This is the ultimate form of security, and with integrated alarm and camera system monitoring services, you can rest assured that any suspicious activities or alarm signals are reported and dealt with quickly.

[3] Observing or analyzing a particular site for safety and business purposes is known as video surveillance. Using a number of video cameras, a large amount of visual data is captured that is to be monitored and screened for intrusion detection. The current surveillance systems used requires constant human vigilance but we cannot always have someone sitting at a desk to monitor the cameras and watch what's going on. That is, the humans have limited abilities to perform in real-time which reduce the actual usability of such surveillance systems. Also such surveillance systems are not reliable for real time threat detection. From the perspective of forensic investigation, a large amount of video data obtained from surveillance video tapes need to be analyzed and this task is very tedious and error prone for a human investigator.

The project describes security system for monitoring confidential area like bank cash room. The idea behind developing this system is that, there are many security systems which continuously capture the video and leads to wastage of memory. So, we are going to develop this system which would intelligently capture video by detecting intrusion and thereby prevent wastage of memory.

Also, the current surveillance systems are used only in the investigation process of the incidence but our system can help in prevention of such incidence by notifying the concerned authority as soon as video surveillance starts, thereby help in prevention of mishap.

By adding these functionalities, we are making security systems more efficient in terms of space consumption as well as in terms of surveillance process.

CHAPTER 2: HISTORY

Today's security systems aren't simply the product of technological developments of the past few years; the groundwork for smart security systems was laid generations ago. To understand how security evolved into what it is today, you have to take a look back at the past.

When World War I ended, an increase in crime followed. As a result, Americans became sensitive to security needs and were eager to find ways to protect themselves and their property. So in early stages, the security system evolved in the form of door shakers, a group of night watchmen who would shake subscribers doors each night to ensure they were locked. With evolution in the field of security systems, we now have security systems in the form of Video surveillance systems.

An early model of a video surveillance systems included a large motorized camera that moved down a track to view the exterior of the home through four peepholes mounted in the front door. The video camera transmitted grainy images of visitors to a stationary television monitor that also served as the control panel where the homeowner could remotely control the camera's movements. The panel, which was located in a separate room away from the camera, was equipped with security features such as an intercom to communicate with visitors, a door lock switch and an alarm button that could activate the alarm at the central station that monitors the residence.

Now, surveillance cameras are as small as one square inch, connected to the Internet and outfitted with powerful lenses that can capture and stream high definition video online that can be viewed from anywhere in the world. Additionally, with a connected smart home, homeowners can program their security systems to send a live video clip of an area if motion is detected when the home is unoccupied and the alarm is activated.

Video surveillance, more commonly called CCTV (closed-circuit television), is an industry that is more than 30 years old and one that has had its share of technology changes. As in any other industry, end users ever-increasing demands on the products and solutions are driving the changes, and evolving technologies are helping to support them. In the video surveillance market, some of the demands are:

- Better image quality
- Simplified installation and maintenance
- More secure and reliable technology
- Reduction in costs
- More built-in system intelligence

To meet these requirements, video surveillance has experienced a number of technology shifts.

CHAPTER 3: REVIEW/ BACKGROUND MATERIAL

Software Used

Matlab R2013a

Hardware Used

Web cam

Matlab R2013a

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python. Power that MATLAB brings to digital image processing is an extensive set of functions for processing multidimensional arrays of which images (two-dimensional numerical arrays) are a special case. There are various functions in MATLAB that have capability of making the process of image processing easy and convenient. These functions, and the expressiveness of the MATLAB language, make image-processing operations easy to write in a compact, clear manner, thus providing an ideal software prototyping environment for the solution of image processing problems.

Web Cam as Camera

We are using webcam as main camera in our system. Webcam serves both the purpose of default image capturing as well as clicking of images at regular interval to detect intrusion at the site of surveillance.

CHAPTER 4: WORK DESCRIPTION

4.1 Working

As soon as surveillance starts, the camera, starts clicking images at a constant interval of time say 5 seconds. This time can be adjusted by the user according to the level of security required.

There is an image which represents the default situation of the area under surveillance and this image is stored in the system. Real time comparison would be made between the clicked image and default image. Now on the basis of result of comparison, clicked image would either be stored or discarded.

On comparing images, if the clicked image is same as the default image, then it can be concluded that there is no intrusion and so the system would continue clicking images and would discard the clicked image. On the other hand, if the clicked image is not found to be same as the default image, then it can be concluded that there is some intrusion in the area under surveillance and so, the clicked image would be stored in the system automatically.

By storing the images where intrusion is present, exact information of timing is known, which can help in investigation process.

Also in our system, we have function to create video using image sequence so it becomes easier for the investigator at the time of investigation.

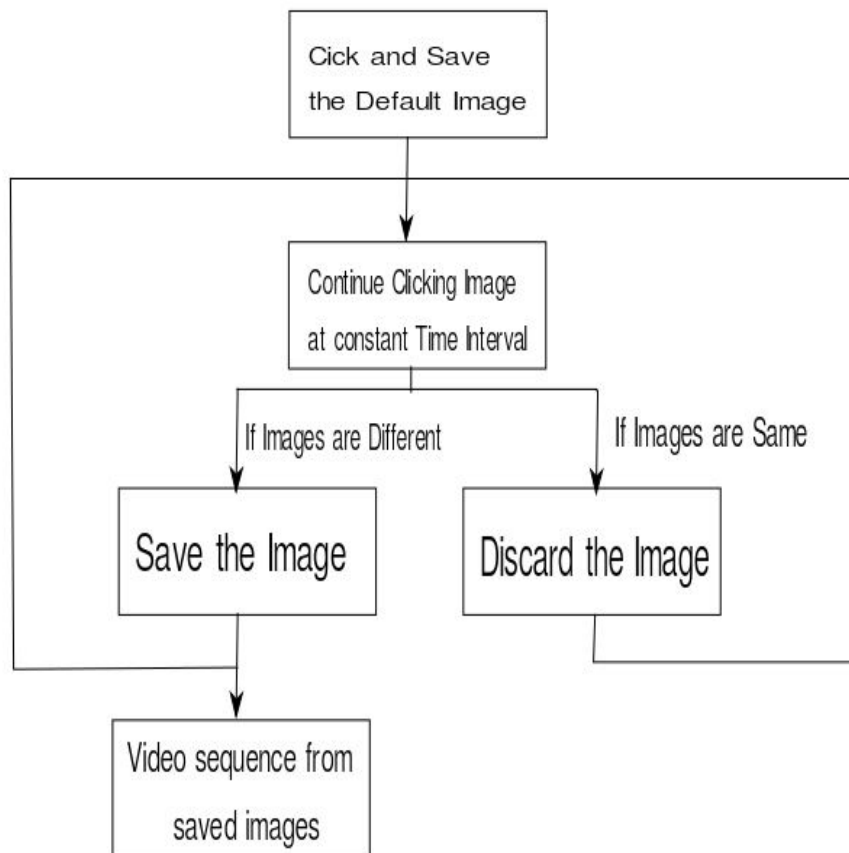


Figure 4.1: Flow Diagram

4.2 Algorithm

Clicking and saving of Default image(1 time process)

Continuous capturing of images at constant interval of time.(Continuous Process)

if: (Default image \neq Clicked Image)

Save the image;

else if: (Default image $=$ Clicked Image)

Stay idle;

Video from image sequence

CHAPTER 5: IMAGE COMPARISON TECHNIQUES

In our project, we are trying to make security systems work intelligently by adding function of Intelligent image capturing and intrusion detection. Intrusion detection in our project is facilitated by image comparison. Image comparison are the basics of Digital Image Processing and is involved in almost all the digital image processes. In our project using image comparing technique, we are able to detect the presence of intrusion in the area and thereby are able to make our system record data only at the time of intrusion.

Various types of image comparing techniques are-

5.1 Edge detection technique

[1]Edge detection is the name for a set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed *edges*. The same problem of finding discontinuities in 1D signals is known as step detection and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction.

The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to:

- Discontinuities in depth
- Discontinuities in surface orientation,
- Changes in material properties
- Variations in scene illumination.

In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation. Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image. If the edge detection step is successful, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified. However, it is not always possible to obtain such ideal edges from real life images of moderate complexity.

Edges extracted from non-trivial images are often hampered by fragmentation, meaning that the edge curves are not connected, missing edge segments as well as false edges not corresponding to interesting phenomena in the image – thus complicating the subsequent task of interpreting the image data.

Edge detection is one of the fundamental steps in image processing, image analysis, image pattern recognition, and computer vision techniques.

The edges extracted from a two-dimensional image of a three-dimensional scene can be classified as either viewpoint dependent or viewpoint independent. A *viewpoint independent edge* typically reflects inherent properties of the three-dimensional objects, such as surface markings and surface shape. A *viewpoint dependent edge* may change as the viewpoint changes, and typically reflects the geometry of the scene, such as objects occluding one another.

A typical edge might for instance be the border between a block of red color and a block of yellow. In contrast a line (as can be extracted by a ridge detector) can be a small number of pixels of a different color on an otherwise unchanging background. For a line, there may therefore usually be one edge on each side of the line.



Figure 5.1: Original image



Figure 5.2: Image after edge detection

5.2 Background Subtraction

[1]Background subtraction, also known as Foreground Detection, is a technique in the fields of image processing and computer vision wherein an image's foreground is extracted

for further processing (object recognition etc.). Generally an image's regions of interest are objects (humans, cars, text etc.) in its foreground. After the stage of image preprocessing (which may include image de-noising, post processing like morphology etc.) object localization is required which may make use of this technique. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called “background image”, or “background model”. Background subtraction is mostly done if the image in question is a part of a video stream. Background subtraction provides important cues for numerous applications in computer vision, for example surveillance tracking or human poses estimation. However, background subtraction is generally based on a static background hypothesis which is often not applicable in real environments. With indoor scenes, reflections or animated images on screens lead to background changes. In a same way, due to wind, rain or illumination changes brought by weather, static backgrounds methods have difficulties with outdoor scenes.

A robust background subtraction algorithm should be able to handle lighting changes, repetitive motions from clutter and long-term scene changes. The following analyses make use of the function of $V(x, y, t)$ as a video sequence where t is the time dimension, x and y are the pixel location variables. e.g. $V(1,2,3)$ is the pixel intensity at (1,2) pixel location of the image at $t = 3$ in the video sequence.

A motion detection algorithm begins with the segmentation part where foreground or moving objects are segmented from the background. The simplest way to implement this is to take an image as background and take the frames obtained at the time t , denoted by $I(t)$ to compare with the background image denoted by b . Here using simple arithmetic calculations, we can segment out the objects simply by using image subtraction technique of computer vision meaning for each pixels in $I(t)$, take the pixel value denoted by $P[I(t)]$ and subtract it with the corresponding pixels at the same position on the background image denoted as $P[b]$.

The background is assumed to be the frame at time t . This difference image would only show some intensity for the pixel locations which have changed in the two frames. Though

we have seemingly removed the background, this approach will only work for cases where all foreground pixels are moving and all background pixels are static. A threshold "Threshold" is put on this difference image to improve the subtraction (see Image thresholding).

This means that the difference image's pixels' intensities are 'thresholded' or filtered on the basis of value of Threshold. The accuracy of this approach is dependent on speed of movement in the scene. Faster movements may require higher thresholds.



Figure 5.3: Original Image



Figure 5.4: Background Image



Figure 5.5: Image After Background Subtraction

5.3 Histogram comparison

[4]A histogram is a graph. It is a graphical representation of the distribution of numerical data. It is an estimate of the probability distribution of a continuous variable (quantitative variable) and was first introduced by Karl Pearson. To construct a histogram, the first step is to "bin" the range of values—that is, divide the entire range of values into a series of intervals—and then count how many values fall into each interval. The bins are usually specified as consecutive, non-overlapping intervals of a variable. The bins (intervals) must be adjacent, and are usually equal size.

If the bins are of equal size, a rectangle is erected over the bin with height proportional to the frequency, the number of cases in each bin. In general, however, bins need not be of equal width; in that case, the erected rectangle has *area* proportional to the frequency of cases in the bin. The vertical axis is not frequency but *density*: the number of cases per unit of the variable on the horizontal axis. A histogram may also be normalized displaying relative frequencies. It then shows the proportion of cases that fall into each of several categories, with the sum of the heights equaling 1.

As the adjacent bins leave no gaps, the rectangles of a histogram touch each other to indicate that the original variable is continuous.

Histograms give a rough sense of the density of the underlying distribution of the data, and often for density estimation: estimating the probability density function of the underlying variable. The total area of a histogram used for probability density is always normalized to 1. If the length of the intervals on the x -axis are all 1, then a histogram is identical to a relative frequency plot.

[2]**Image** – An image may be defined as a two-dimensional function, $f(x, y)$, where x and y are spatial (plane) coordinates, and the amplitude of f at any pair of coordinates (x, y) is called the intensity or 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. The field of digital image processing refers to processing digital images by means of a digital computer.

An **image histogram** is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each 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 grey 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. Thus, the histogram for a very dark image will have the majority of its data points on the left side and center of the graph. Conversely, the histogram for a very bright image with few dark areas and/or shadows will have most of its data points on the right side and center of the graph.

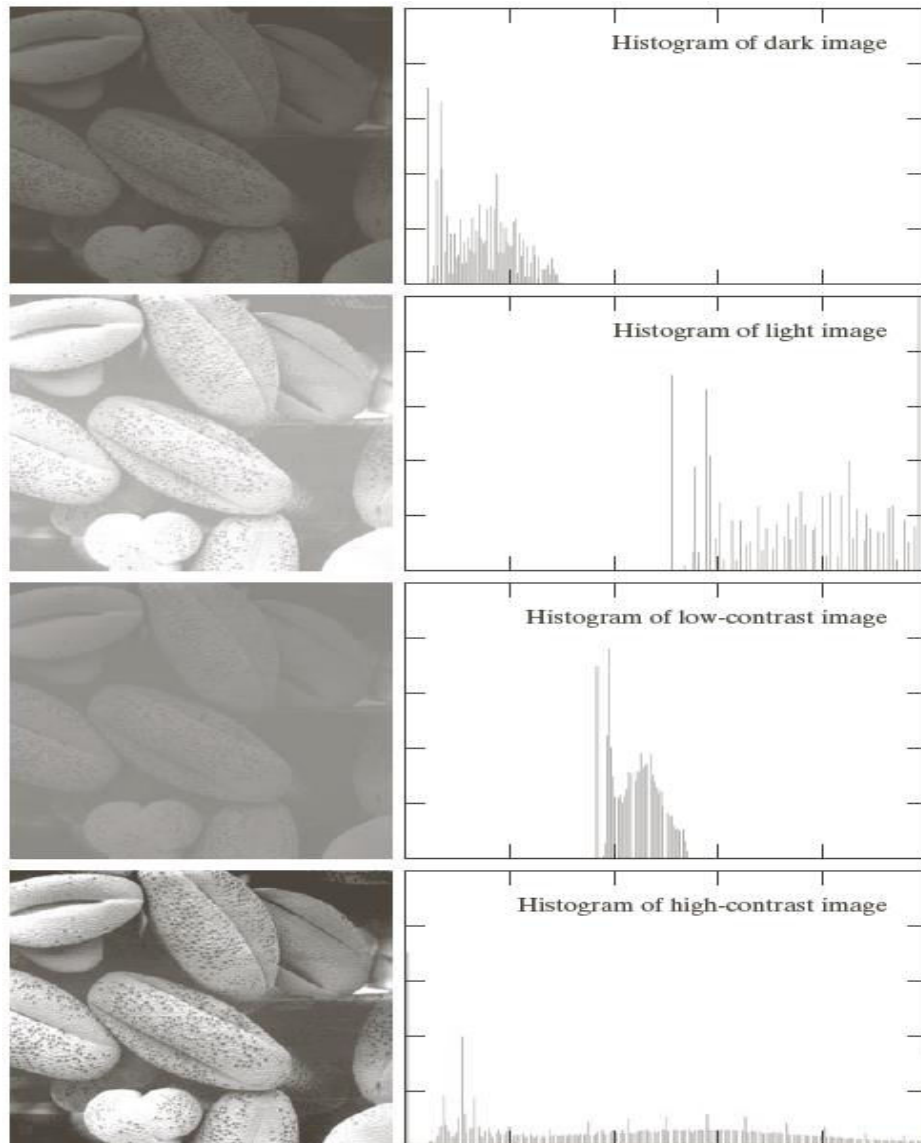


Figure 5.6: Basic Image types and their corresponding histograms.

Normalization- The normalized count is the count in a class divided by the total number of observations. In this case the relative counts are normalized to sum to one (or 100 if a percentage scale is used). This is the intuitive case where the height of the histogram bar represents the proportion of the data in each class.

5.3.1 Comparing images using Histograms

The histogram based method was historically the first algorithm provided . It was designed to compare two images of equal size (dimensions). The method is based on comparison of histograms (color counters) and merely tests whether two images of the same size contain the same number of pixels for each colour. It basically returns a percentage reflecting the rate of pixels of matching colors.

Advantages of this method are **simplicity and fast performance**. This method can be successfully used in quick test of image comparison.

5.3.2 Applications of Histograms in Image processing

Histograms has many uses in image processing. The first use as it has also been discussed above is the analysis of the image. We can predict about an image by just looking at its histogram. Its like looking an x ray of a bone of a body.

The second use of histogram is for brightness purposes. The histograms has wide application in image brightness. Not only in brightness, but histograms are also used in adjusting contrast of an image.

Another important use of histogram is to equalize an image.

And last but not the least, histogram has wide use in thresholding. This is mostly used in computer vision.



Figure 5.7: Clicked image

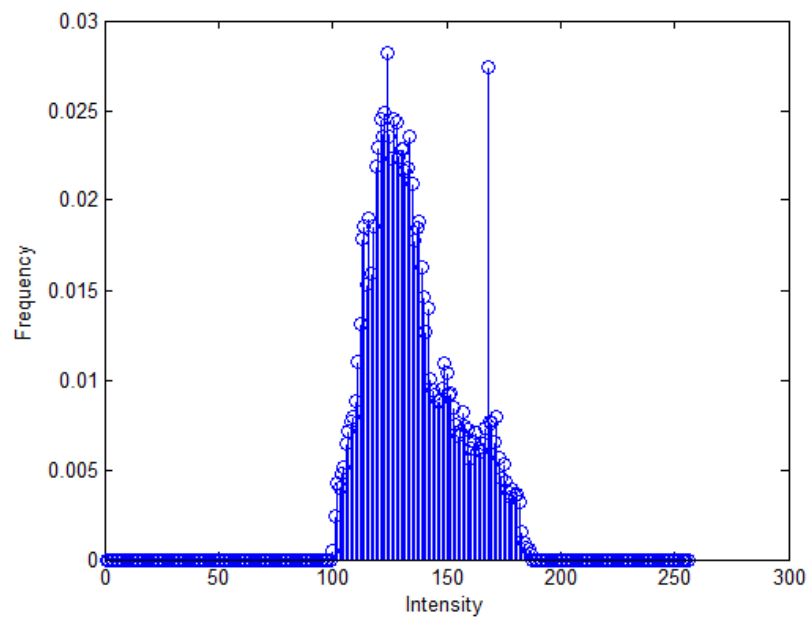


Figure 5.8: Histogram of the Image

CHAPTER 6: TECHNIQUE USED FOR IMAGE COMPARISON

Out of many image comparison techniques, the comparison technique which we are using is, Image histogram comparison. This is because, the system which we are working on is a real time system and it requires the image comparison results to be generated as quickly as possible, for its proper functioning. Although, other image comparison techniques may be more accurate than histogram comparison technique but here quick results are required. Histogram comparison being simple and fast performer can be used for the quick comparison of images to sense intrusion.

Histogram is a graph that shows frequency of anything. It represents the information in the form of bars that represent frequency of occurring of data in the whole data set. It consists of two axis the x axis and the y axis which contains event whose frequency is to be counted and frequency of event respectively.

The histogram based method was historically the first algorithm provided. It was designed to compare two images of equal size (dimensions). The method is based on comparison of histograms (colour counters) and merely tests whether two images of the same size contain the same number of pixels for each colour. It basically returns a percentage reflecting the rate of pixels of matching colours.

Formula Used for calculating difference between histograms-

$$P = \sum (hn_1 - hn_2)^2;$$

where hn_1 is the normalized histogram of default Image and hn_2 is the normalized histogram of the clicked image for which intrusion is to be checked.

P represents the difference between the image histograms, on the basis of which intrusion is detected.

If $P < Th$, where Th represents threshold which is .0025 for lighting conditions during the experiment, then we can conclude that no intrusion is present.

Where as if $P > Th$, then we can conclude that intrusion is present.

Note-[5] Threshold may be different for different lighting conditions.

CHAPTER 7: PROJECT WORK

7.1 Code 1

Webcam interfacing with MATLAB software and, clicking and saving of default image

In this part of our project, we are able to interface the internal web cam of our laptop with MATLAB Software. This internal web cam is acting as a source to capture still images. Here, we are able to click a default image of a area under surveillance and are able to save this image in our system. This default image represents that situation of the area when there is no intrusion present.

CODE:

```
clc;
vid = videoinput('winvideo', 1);%imaqhwinfo
start(vid);
Im = getsnapshot(vid);
imshow(Im);
img=fullfile('C:\Users\Vaibhav Jain\Desktop\project 8\def\comp.jpg');
imwrite(Im, img);
```

7.2 Code 2

Real Time comparison between clicked image and default image

In this part of project, we are able to make real time comparison between a default image and image which is being clicked at a fixed interval of time (say 5 seconds) without actually saving these clicked images. On the basis of this real time comparison, we are able to detect the presence of any type of intrusion present in the area under surveillance.

CODE:

```
clc
Im1 = imread('C:\Users\Vaibhav Jain\Desktop\project 8\def\comp.jpg');
```

```

imshow(Im1);
Im1 = rgb2gray(Im1);
Im1 = im2double(Im1);
hn1 = imhist(Im1)./numel(Im1);
    figure;
    stem(hn1);
    xlabel('Intensity');
    ylabel('Frequency');

cam = videoinput('winvideo', 1);
start(cam)
k=input('Enter the time taken between two images');
i=0;
j=1;
while i<18
    pause(k);
    i=i+k;
    Im2=getsnapshot(cam);
    figure;
    imshow(Im2);
    Im2 = rgb2gray(Im2);
    Im2 = im2double(Im2);

    hn2 = imhist(Im2)./numel(Im2);

    figure
    stem(hn2);
    xlabel('Intensity');
    ylabel('Frequency');
    %x=size(Im1);
    %y=size(hn2);
    %disp(hn1);
    %disp(hn2);
    p=sum((hn1-hn2).^2);

```

```

disp(p);

if p<25e-004
    disp('No Intrusion')
else
    disp('Intrusion');
%     vid = videoinput('winvideo', 1);%imaqhwininfo
% start(vid);
% Im = getsnapshot(vid);
% imshow(Im);
img=fullfile('C:\Users\VaibhavJain\Desktop\project8',sprintf('%d.jpg',j));
imwrite(Im2,img,'jpg');
j=j+1;

end

end

Untitled13();

```

7.3 Code 3

Constructing video from image sequences.

In this part of project, we are using the clicked images where intrusion is detected, to construct a video. Using this method we are able to save memory to a large extent.

CODE:

```

function Untitled13()
cd('C:\Users\Vaibhav Jain\Desktop\project 8');
F = dir('*.jpg');
NF= size(F,1);
Im = uint8(zeros([120 160 1 NF*5]));
VideoObj = VideoWriter('video.avi');
VideoObj.FrameRate = 5;
%VideoObj.Quality = 80;
count=1;
for i = 1 : NF

```



```
I = imread(F(i).name);
RI = imresize(I,[120 160]);
%Each Image is copied 5 times so that in a second 1 image can be
viewed
for j = 1 : 5
    Im(:,:,j,count)=RI;
    count = count + 1;
end
end

open(VideoObj);
writeVideo(VideoObj, Im);
%close(VideoObj);
end
```

CHAPTER 8: PROBLEMS FACED

8.1 Problem 1

Difference in dimensions of images:

This was a problem which we faced during clicking of default image. Initially we were using the default camera software of Windows 10 for clicking and saving of default image then we were making real time comparison between this default image and the image which was clicked by MATLAB interfaced camera. Now the problem that we faced was the huge resolution difference between the picture quality of the default and the clicked image due to which we are getting wrong comparison results.

Solution:

To solve this problem, the first approach which we used, was to increase the resolution of the images clicked by MATLAB interfaced camera but still we were unable to get the proper results.

The second approach which we used, was to decrease the resolution of default image which was clicked using default camera software of Windows 10, but still we were not able to match the dimensions and picture quality of default and clicked image.

Finally, we came to the Third approach which proved to be the solution of our problem. In this approach we decided to click as well as save the default image using MATLAB interfaced camera. After coding for this, we are able to match the dimensions of default and clicked image. Now we were able to get accurate results of the image comparison.



Figure 8.1: Image clicked using default camera software of windows 10

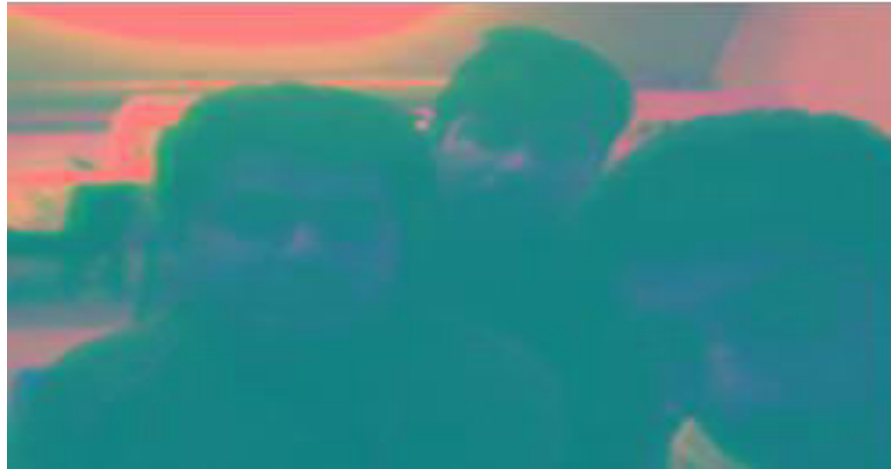


Figure 8.2: Image clicked using same camera but interfaced with MATLAB

NOTE: Although both the above shown images are clicked using same camera but still we can see the difference in the resolution of image after the camera is interfaced with MATLAB.

8.2 Problem 2

Use of Single Camera for image capturing and video recording:

Initially, we tried to use a single camera for image capturing as well as video recording but we were getting error during execution of the code. This was because MATLAB does not support use of same camera for two different functionalities simultaneously.

Solution:

As our project is completely MATLAB software based, so the only solution to this problem is to use two different cameras for capturing still images and recording videos simultaneously.

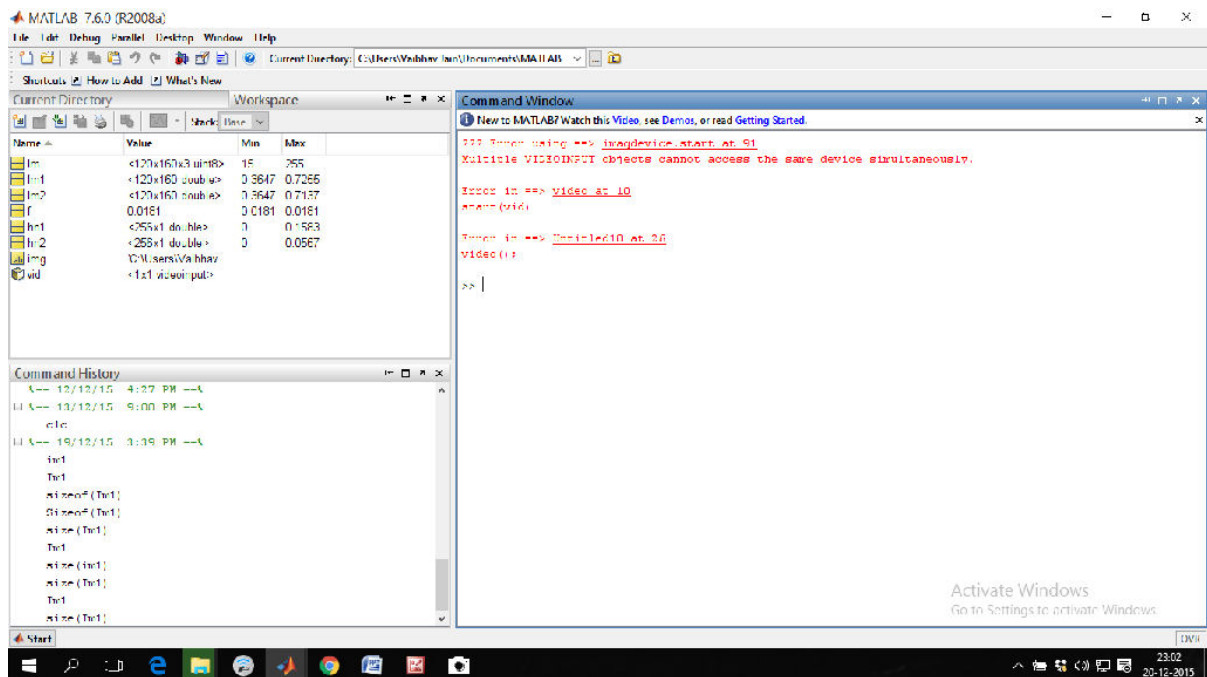


Figure 8.3: Error regarding camera use

ERROR: Multiple VIDEOINPUT objects cannot access the same device simultaneously.

CHAPTER 9: FUTURE SCOPE

Additional SMS Alert Functionality

The current security cameras are used as evidence after the incidence has already taken place. We will try to make our system more real-time efficient by adding a notification functionality to it. In this, a notification function would be interfaced to the system which would notify the concerned authority as soon as the video recording starts which may help in avoiding some mishap. This notification can be in the form of E-mail, Buzzer or a SMS.

Face recognition Functionality

We may add a functionality of face recognition to our system. This would make the process of image capturing even more intelligent. Using this functionality, our security system would be able to detect the faces of people entering the area under surveillance, if the face matches the face of people who has access to the area, then it would not save the clicked images, but in case the result of face recognition finds the person to be somebody other than the authorised person then it would start image capturing and saving them in the system.

CHAPTER 10: RESULTS

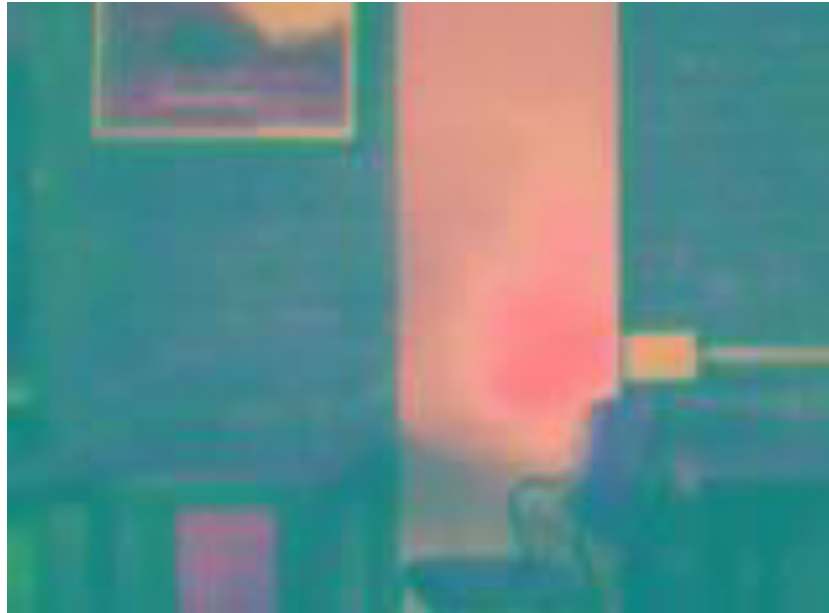


Figure 10.1: Default Image

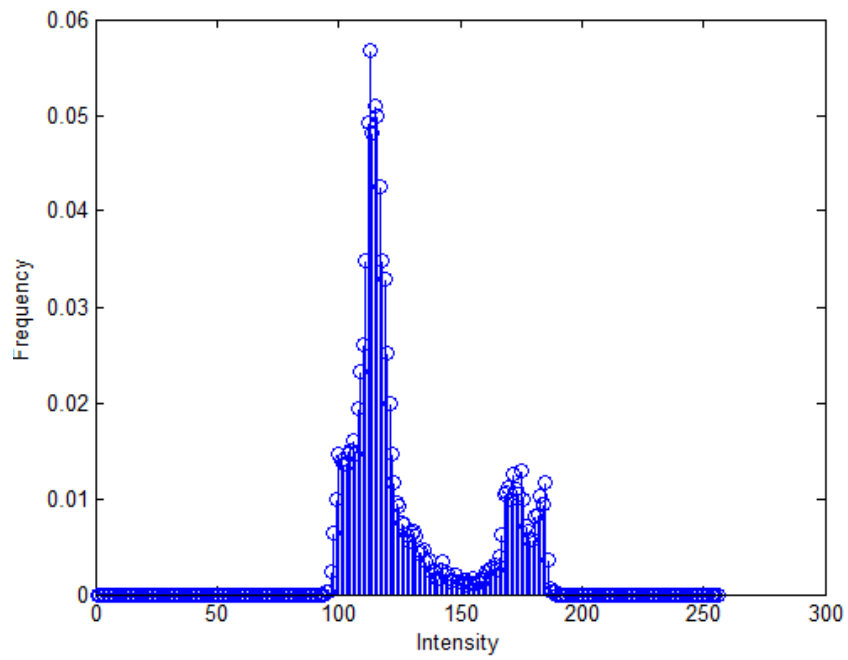


Figure 10.2 : Histogram of Default Image



Figure 10.3: Clicked Image1 (No intrusion)

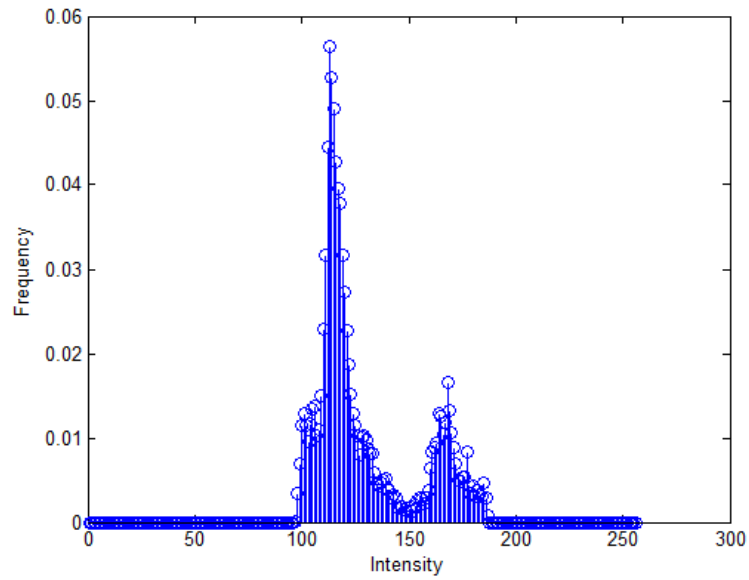


Figure 10.4: Histogram of Clicked Image1 (No intrusion)

Note: $P=0.0014 < Th$, thereby showing that no intrusion is present and hence the image is not saved in the system.



Figure 10.5: Clicked Image2 (Intrusion Present)

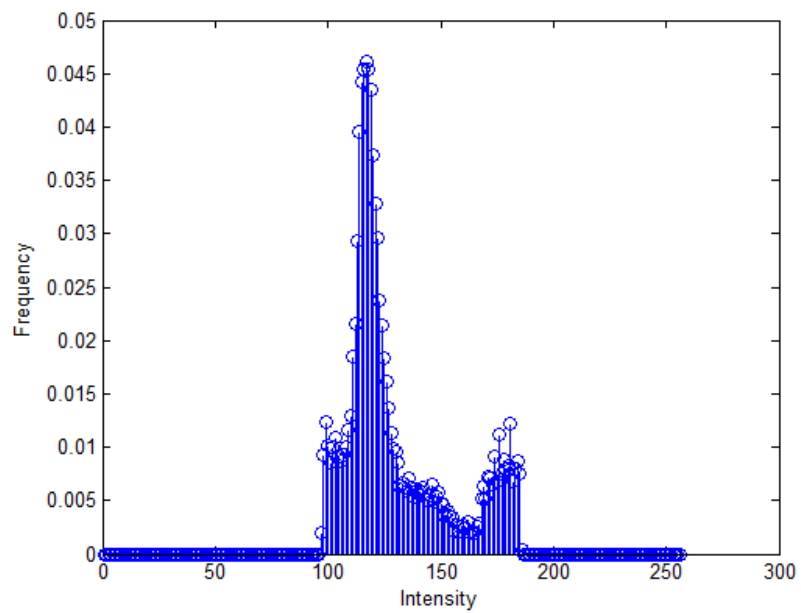


Figure 10.6: Histogram of Clicked Image2 (Intrusion Present)

Note: $P=0.0044 > Th$, thereby showing that intrusion is present and hence the image is saved in the system.



Figure 10.7: Clicked Image3 (Intrusion Present)

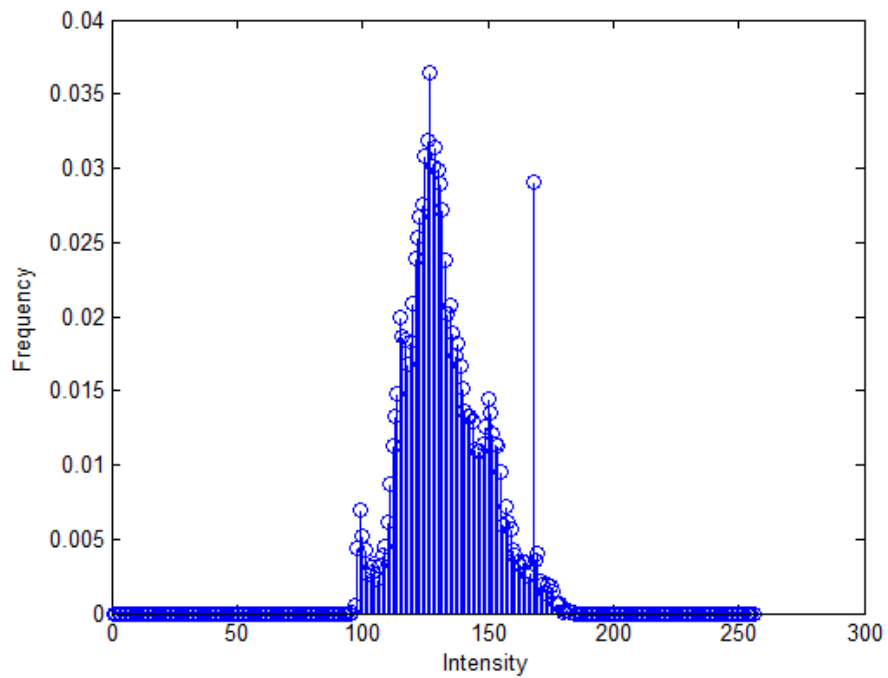


Figure 10.8: Histogram of Clicked Image3 (Intrusion Present)

Note: $P=0.0208 > Th$, thereby showing that intrusion is present and hence the image is saved in the system.

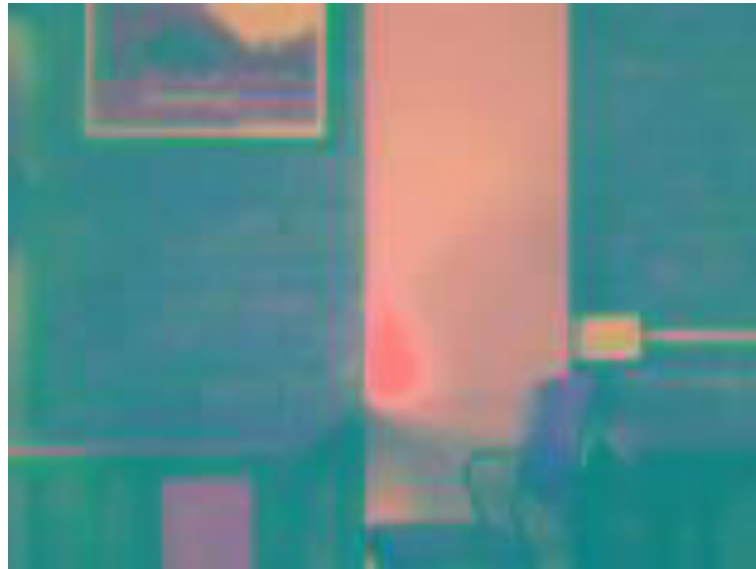


Figure 10.9: Clicked Image4 (No intrusion)

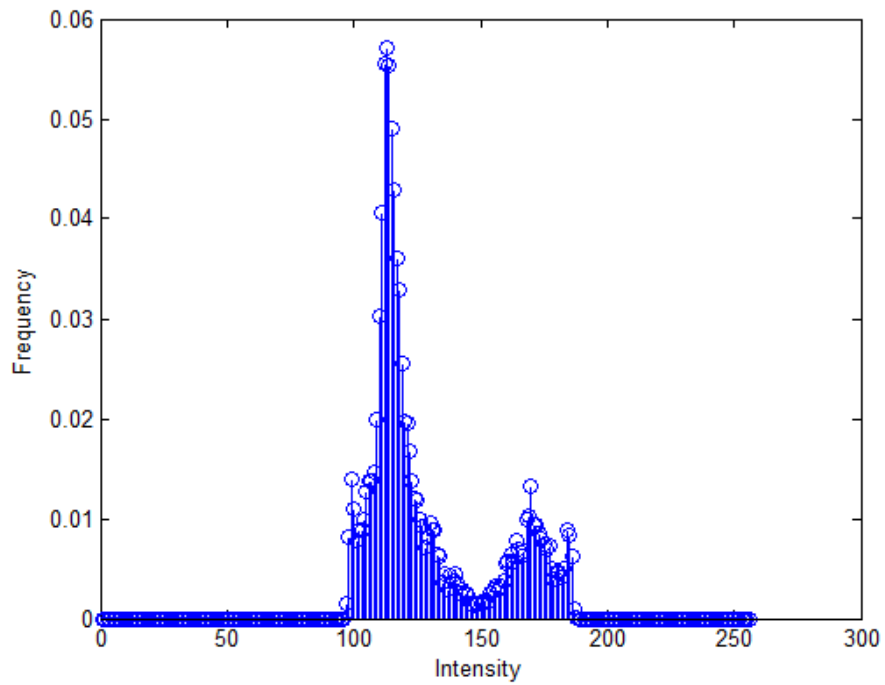


Figure 10.10: Histogram of Clicked Image4 (No intrusion)

Note: $P = 8.7084e-04 < Th$, thereby showing that no intrusion is present and hence the image is not saved in the system.



Figure 10.11: Clicked Image5 (Intrusion Present)

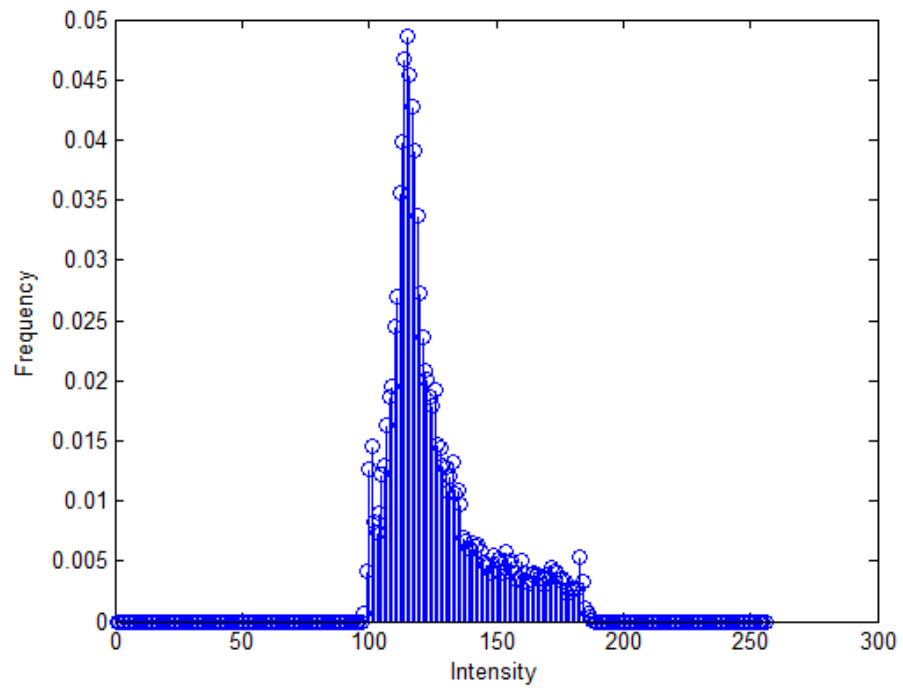


Figure 10.12: Histogram of Clicked Image5 (Intrusion Present)

Note: $P=0.0028 > Th$, thereby showing that intrusion is present and hence the image is saved in the system.



Figure 10.13: Clicked Image6 (Intrusion Present)

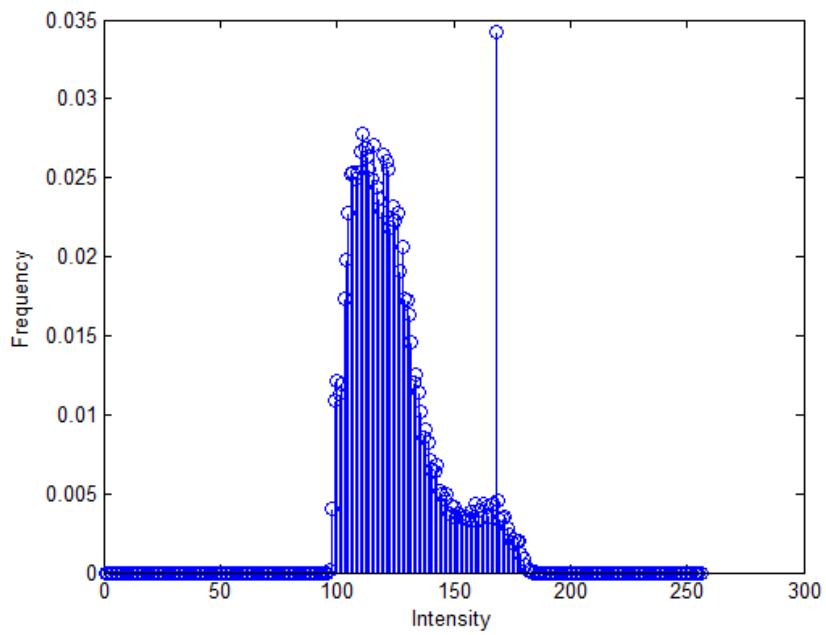


Figure 10.14: Histogram of Clicked Image6 (Intrusion Present)

Note: $P=0.0079 > Th$, thereby showing that intrusion is present and hence the image is saved in the system.

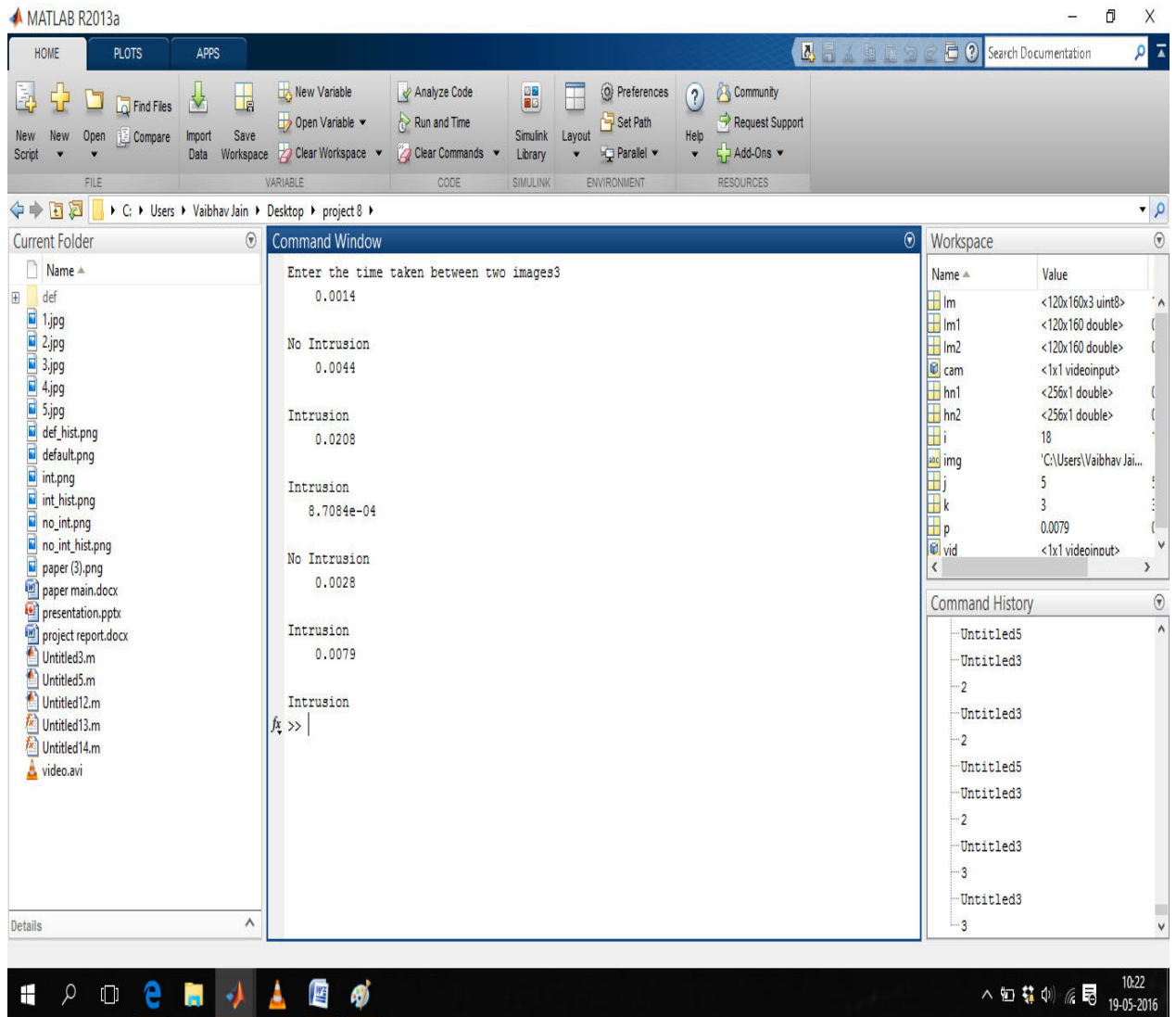


Figure 10.15: Result of image comparison.

10.1 Conclusion

By working on this project, we are able to add functionalities to security systems which help in countering few of its drawbacks.

We are able to solve the problem of memory wastage without using any extra hardware.

But still, there are few disadvantages of this system as well. One such disadvantage is as this system is histogram based, so its working requires constant lighting conditions limiting its uses to places like bank cash rooms.

Also, there are many other functionalities which could be added to security systems. One such facility is notification function which may help in prevention of ill practices like robbery and theft. This notification can be in the form of buzzer which can get activated when intrusion in the surveillance area is detected.

Also, memory saving functionality can be further enhanced by adding face recognition for those who have access to the surveillance area, so that recording is done only when people who do not have access to the area gets recorded.

REFERENCES

- [1] Arun Hampapur, Lisa Brown, Jonathan Connell, Ahmet Ekin, Norman Haas, Max Lu, Hans Merkl, Sharath Pankanti, Andrew Senior, Chiao-Fe Shu, and Ying Li Tian “Smart Video Surveillance” IEEE SIGNAL PROCESSING MAGAZINE MARCH 2005.
- [2] R. Gonzalez, R. Woods “Digital Image Processing” 3rd edition.
- [3] Abhishek Kumar Pandey ,Aditya Ashok Kulkarni ,Shruti Jitendra Shah “Automation in Video Security Surveillance using Mobile Remote Control” Abhishek Kumar Pandey et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 1814-1817.
- [4] Aleksandra Mojsilovic, Jianying Hu “A Method for Color Content matching of images”.
- [5] Dinesh Sonker, M. P. Parsai “Comparison of Histogram Equalization Techniques for Image Enhancement of Grayscale images of Dawn and Dusk” International Journal of Modern Engineering Research (IJMER) www.ijmer.com Vol. 3, Issue. 4, Jul - Aug. 2013 pp-2476-2480.