CCTV Footage Enhancement Tool

Submitted By:

Danish Ali Cheema

Supervised By:

Syed Muhammad Naqi

Department of Computer Science

Quaid-i-Azam University

Islamabad

Session (2014-2018)

Abstract

CCTV footage enhancement is a desktop based tool to improve the quality of CCTV footages so that result is more suitable than original image for specific. Digital image enhancement techniques are used for the enhancement of footage which provides a multitude of choices for improving the visual quality of images. Appropriate choice of such techniques is greatly influenced by the imaging modality, task at hand and viewing conditions.

CCTV usage is quite common these days. CCTV cameras are found in everywhere (shopping malls, security centers, banks, etc). Also, security is primary concern of today's world; therefore CCTVs have become more a necessity. Image produced by CCTVs is usually blur, low quality and do not reveal the in-depth details (facial detection, pixels). This project assists in enhancement of footage with particular reference to spatial domain techniques, point processing methods and histogram processing.

In this tool, along with enhancement filters other functionalities like motion detection and automatic comparison among algorithms at each frame is provided to make the tool more sophisticated.

Table of Content

CHAPTER 1
INTRODUCTION

CCTV-Closed Circuit Televisions are used for monitoring purposes. Their signals are not distributed publically but are monitored by subjected authorities. CCTVs are basically used for surveillance and security purposes [1]. They are used in a community for the protection of people and their property. CCTV videos are widely used as a source of evidence in courts. They are a great source for maintaining public order, preventing any social nuisance thus help in providing economic and social well-being of a community. Thereby, the quality of videos produced by CCTV is of prime importance for maintaining a safe environment and identification of criminals. As we know, CCTV cameras are held at specific positions and angles, thus it is difficult to get complete overview of a specific region. Another factor influencing the production of low quality videos are the smoke, dust which makes the things harder to understand. This project focuses on video enhancement using algorithms for improving the quality of frames to produce better quality videos. Image enhancement includes accentuates or sharpens video features such as edges, boundaries, or contrast to make a graphic display more helpful for display and analysis. The enhancement doesn't increase the inherent information content of the data, but it increases the dynamic range of the chosen features so that they can be detected easily [2].

1.0 Project Management Plan

A project plan is the key to a successful project and is the most important document that needs to be created when starting any business project.

		Name	Duration
1		**Analysis Phase**	57 days?
2		Understanding Problem	1 day?
3		Making SPMP Document	2 days
4		**Analysis**	25 days?
5		Collect requirements	2 days
6		Refine requirements	1 day?
7		Make Document 1	2 days
8		**Specific requirements**	15 days?
9		User interfaces	2 days
10		software interfaces	1 day?
11		Making Document 2	2 days
12		**Finding Functional Requirments**	10 days
13		**Identify usecases**	5 days
14		Refine usecases	2 days
15		Define usecase Text	2 days
16		Draw Usecase Diagram	1 day
17		Identify System Functions	2 days
18		Refine Functional Requirements	1 day
19		Make Document 3	2 days

		Name	Duration
20		**Finding Non Functional Requirments**	1 day
21		Identify constraints	1 day
22		Making srs document	5 days
23		**Design**	25 days
24		Develop interface design	5 days
25		Refine interface design	2 days
26		Study algorithms	5 days
27		Develop algorithm	5 days
28		Define pseudo code	5 days
29		Review pseudo code	3 days
30		**Validate design**	2 days
31		Evaluate design	2 days
32		Make final Document	2 days
33		**Coding**	10 days
34		Explore matlab	10 days
35		Implement defined algorithms	5 days
36		Implement interfaces	3 days
37		refine implementation	2 days
38		**Testing**	4 days
39		Verification	2 days
40		Validation	2 days

Fig 1.1: Project plan

1.1 Project Overview

Image enhancement process consist to improve the appearance of an image or to convert the image to a form better suited for analysis by a human or a machine. Enhancement of image is very challenging issue in many research and application areas. Image enhancement techniques are used to improve certain features by modifying the colors or intensities [3]. There is no proper application or software for CCTV footage image enhancement. It requires complex algorithm development and implementation techniques for image enhancement. It is necessary to enhance the pixel quality as it is used widely as evidence in courts.

1.1.1 Scope

It is a desktop based project which will be deployed in security departments, helping them to understand the minute details of the picture produced thus assisting them in evidence production and crime scene analysis, forming a whole new video consisting of image enhancement techniques deployed.

1.1.2 Objectives

The aim of image enhancement is to improve the interpretability or perception of information in images for human viewers, or to provide `better' input for other automated image processing techniques. The main objective of the system is to save the time of the user (security and monitoring staff). This will not only save their time but will also assist them getting into the minor details of the CCTV footage.

Image enhancement techniques are divided into **two** broad categories:

1. Spatial domain methods, which operate directly on pixels, this is the reason that I implemented this enhancement technique in my project that the result results obtained are better.

2. Frequency domain methods, which operate on the Fourier transform of an image. Unfortunately, there is no general theory for determining what `good' image enhancement is, when it comes to human perception. If it looks good, it is good! However, when image enhancement techniques are used as pre-processing tools for other image processing techniques, then quantitative measures can determine which techniques are most appropriate [4].

1.1.3 Tools and Techniques

We will use **MATLAB** for development purpose and compare different algorithms for image enhancement and also for GUI purposes**.**

1.1.4 Techniques for Image Enhancement

Fourier transformations are nice, but the real fun begins when they can be used in the process of enhancing existing images. The results of these enhancements are dependent on the type of a specific application, which means that they are very much problem-oriented.

There are mainly two methods for image-enhancement: one deals with images in the spatial domain (say f(x; y)), the other one deals with images in the frequency domain (say F (u; v)). The first method is based on the processing of individual pixels in an image, the second is based on modifying the Fourier transform of an image.

1.1.5 System Designing Approach

Unified Modeling Language (UML).

1.1.6 Operating System

The system must use windows 7/8/8.1/10.

1.1.7 Definitions and Abbreviations

CCTV	Closed-Circuit Television.
UML	Unified Modeling Language
GUI	Graphical User Interface
MS Visio	Designing tool
UI	User Interface
DESP	Description
SAD	Sum Of Absolute Differences

SSD	System Sequence Diagram
RGB	Red, Green, Blue

1.2 Spatial Domain Methods:

Here, image processing functions can be expressed as:

$$\mathbf{G(x; y) = T(f(x; y));} \quad \ldots\ldots\ldots\ldots (1)$$

With f(x; y) the input image, g(x; y) the processed image (i.e. the result or output image) and T an operator on f, defined over some neighborhood N of (x; y). For N, we mostly use a rectangular sub image that is centered at (x; y) [5].

Image enhancement simply means, transforming an image f into image g using T. (Where T is the transformation). The values of pixels in images f and g are denoted by r and s, respectively. As said, the pixel values r and s are related by the expression;

$$s = T(r) \quad \ldots\ldots\ldots\ldots\ldots..(2)$$

Where T is a transformation that maps a pixel value r into a pixel value s. The results of this transformation are mapped into the grey scale range as we are dealing here only with grey scale digital images. So, the results are mapped back into the range [0, L-1], where L=2k, k being the number of bits in the image being considered. So, for instance, for an 8-bit image the range of pixel values will be [0, 255]. The same theory can be extended for the color images too. A digital gray image can have pixel values in the range of 0 to 255.

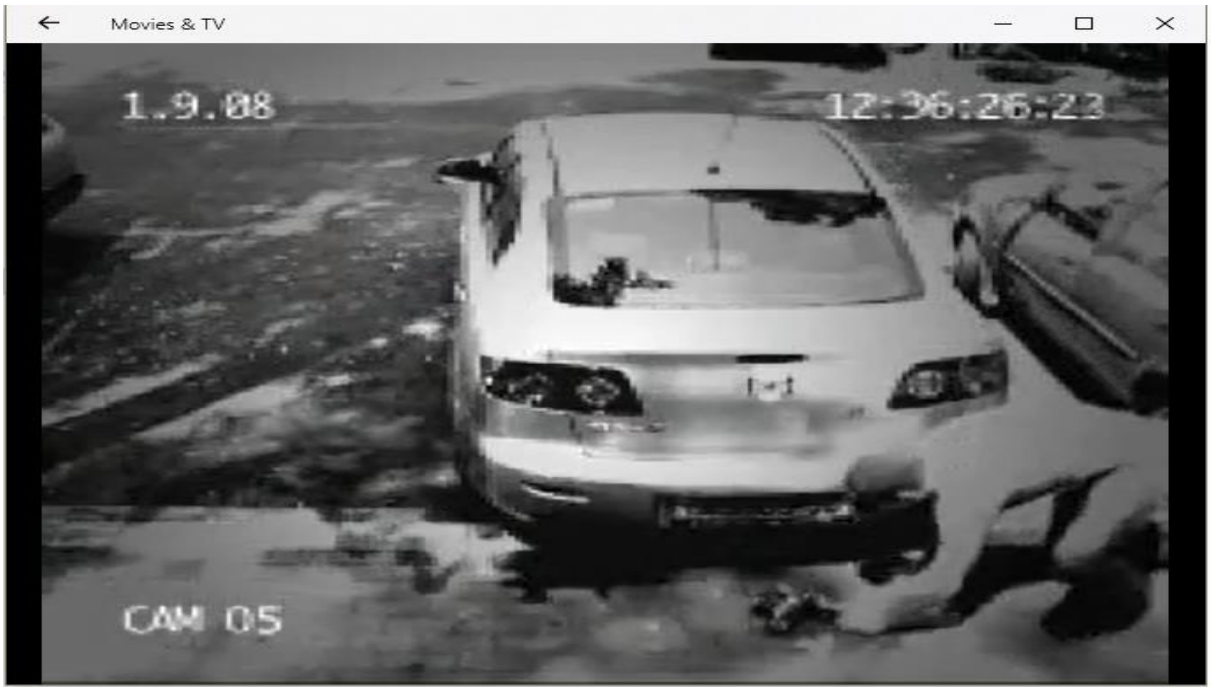

Fig 1.2 (a): Before

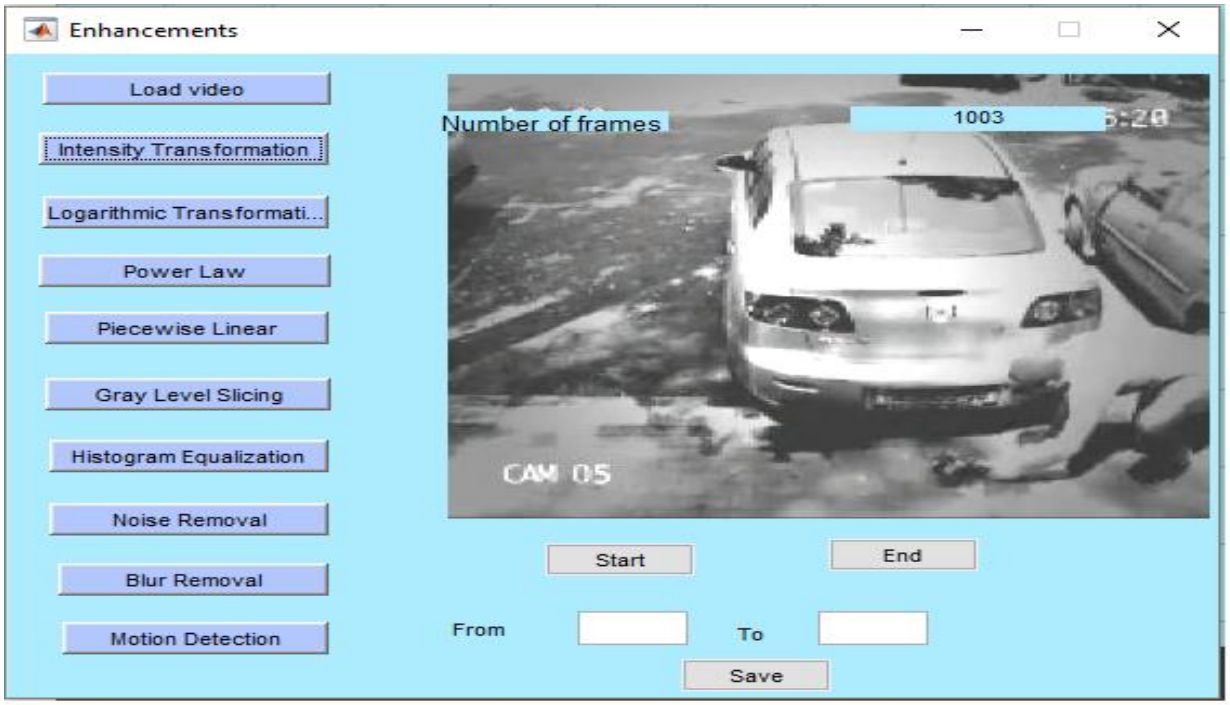

Fig 1.2 (b): After

1.2.1 Point Processing

The simplest spatial domain operations occur when the neighborhood is simply the pixel itself. In this case T is referred to as a grey level transformation function or a point processing operation.

Point processing operations take the form;

$$s = T(r) \quad \ldots\ldots\ldots\ldots\ldots\ldots (3)$$

Where, s refers to the processed image pixel value and r refers to the original image pixel value.

1.2.2 Point Processing -Intensity Transformation

Intensity transformation is the simplest of all image processing techniques. In this technique the image can be transformed into different scales i.e their contrast between black and white colors can be transformed by;

$$g(x, y) = T[f(x, y)] \quad \ldots\ldots\ldots (4)$$

where,

f(x, y): input image,

g(x, y): processed image, and T: operator.

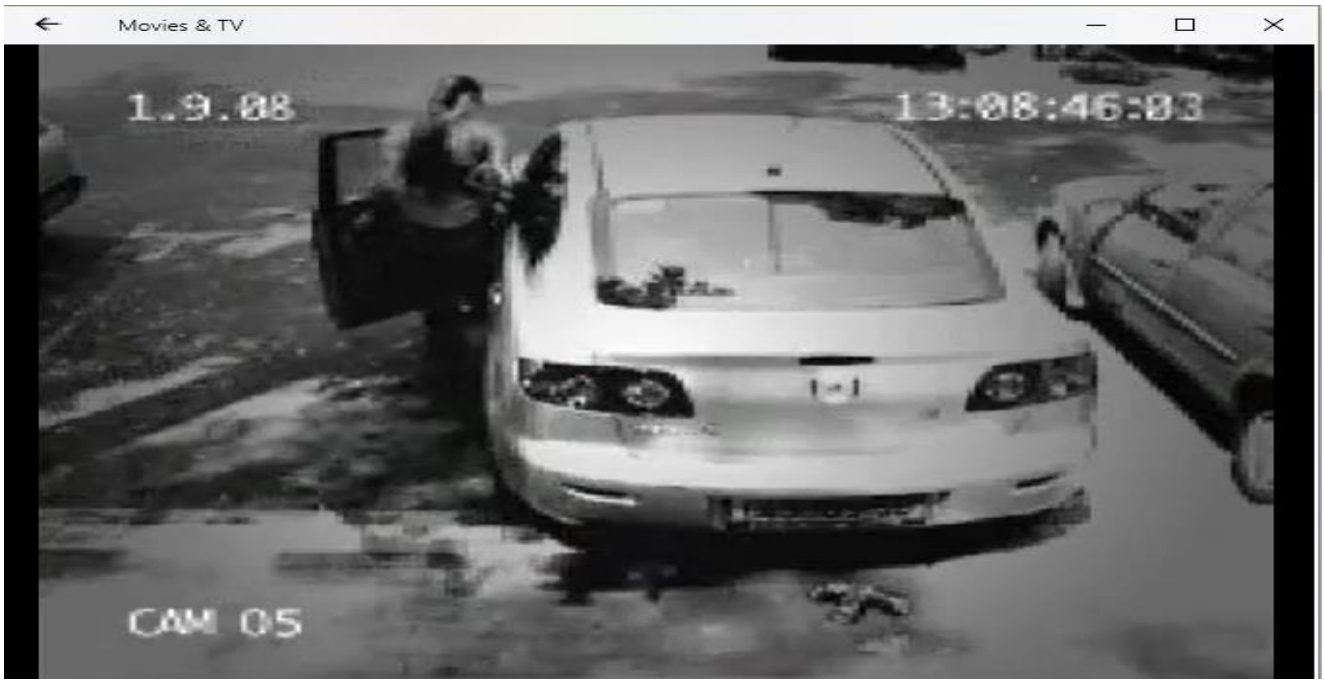

Fig 1.3(a): Before Intensity Transformation

Fig 1.3(b): After Intensity Transformation

1.2.3 Point Processing – Basic Grey Level Transformations

There are many different kinds of grey level transformations. Three of the most common are;

- Linear - Negative/Identity
- Logarithmic - Log/Inverse log
- Power law - nth power/nth root

It highlights a specific range of grey levels;

- Similar to thresholding.
- Other levels can be suppressed or maintained.
- Useful for highlighting features in an image.

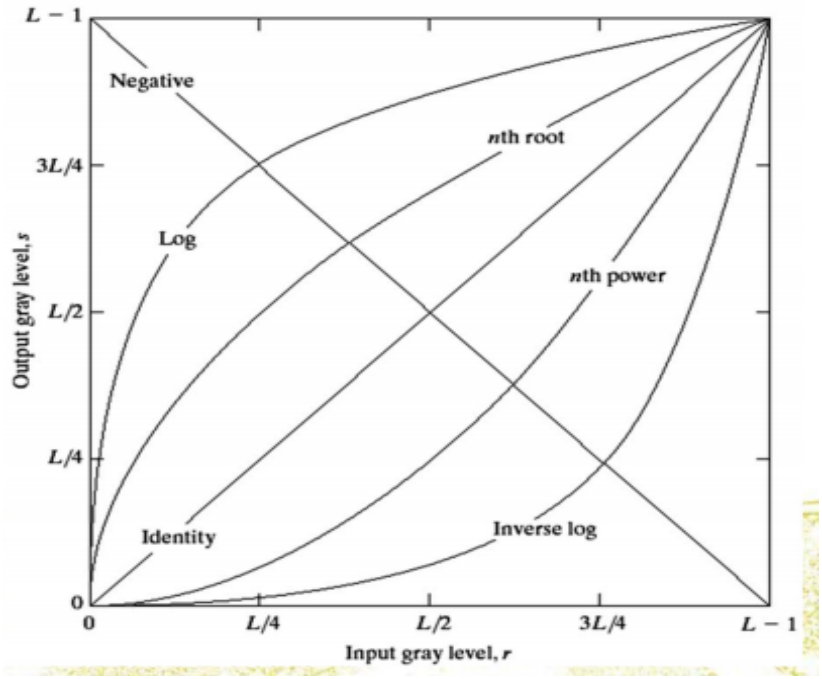

Fig 1.4: Grey Level Transformation [5]

1.2.4 Point Processing – Logarithmic Transformation

The general form of the log transformation is

$$\mathbf{s = c * log (1 + r)}\text{……..(5)}$$

Where s and r are the pixel values of the output and the input image and c is a constant. The value 1 is added to each of the pixel value of the input image because if there is a pixel intensity of 0 in the image, then log (0) is equal to infinity. So 1 is added, to make the minimum value at least 1.

During log transformation, the dark pixels in an image are expanded as compare to the higher pixel values. The higher pixel values are kind of compressed in log transformation. This results in following image enhancement. The value of c in the log transform adjusts the kind of enhancement the user is looking for [6].

The log transformation maps a narrow range of low input grey level values into a wider range of output values. The inverse log transformation performs the opposite transformation. Log functions are particularly useful when the input grey level values may have an extremely large range of values. We usually set c to 1 Grey levels must be in the range [0.0, 1.0].

Fig 1.5 (a) : Before Logarithmic Transformation

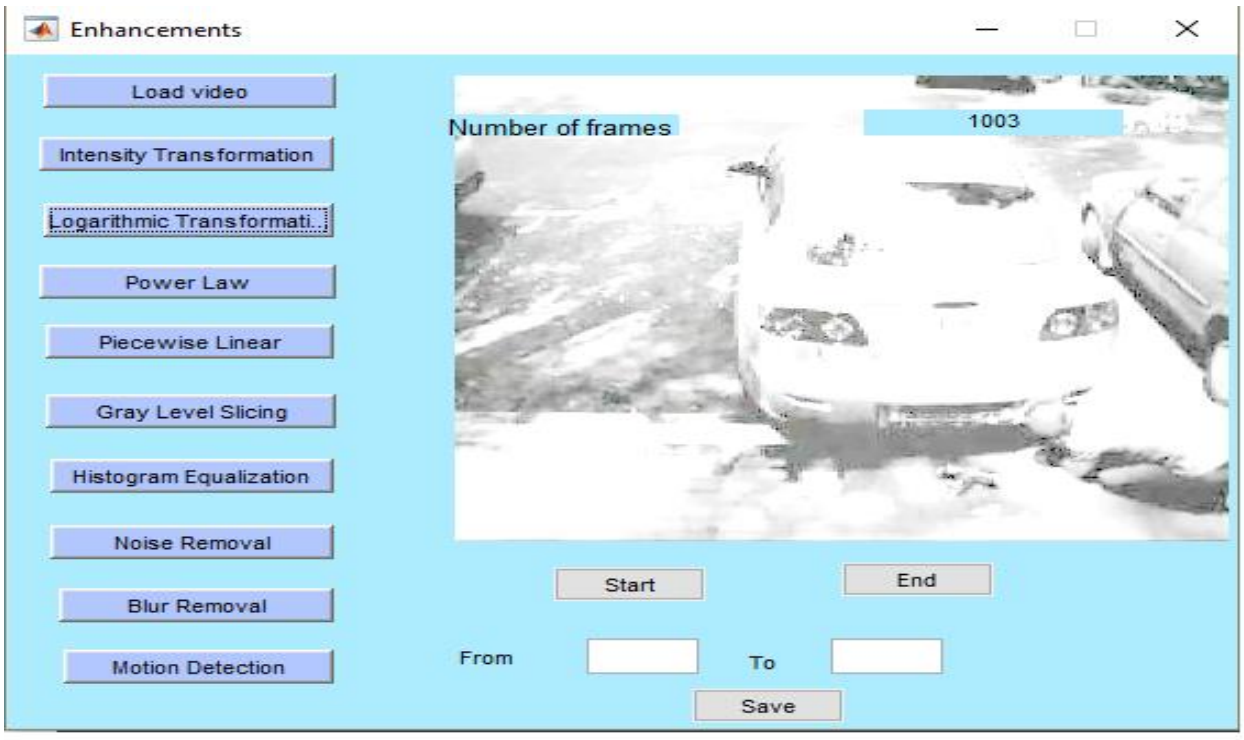

Fig 1.5 (b) : After Logarithmic Transformation

1.2.5 Point Processing – Power Law Transformation

Power law transformations have the following form;

$$\mathbf{s = c * r^{\gamma}} \text{.....(6)}$$

It maps a narrow range of dark input values into a wider range of output values or vice versa. Varying γ gives a whole family of curves. It is also known as Gamma Transformation. Variation in the value of γ varies the enhancement of the images. Different display devices / monitors have their own gamma correction, that's why they display their image at different intensity.

This type of transformation is used for enhancing images for different type of display devices. The gamma of different display devices is different. For example Gamma of CRT lies in between of 1.8 to 2.5, which means the image displayed on CRT is dark.

We usually set c to 1. Grey levels must be in the range [0.0, 1.0].

Fig 1.6 (a): Before Power Law Transformation

Fig 1.6 (b): After Power Law Transformation

1.2.6 Point Processing – Piecewise Linear Transformation

Rather than using a well-defined mathematical function we can use arbitrary user-defined transforms. Formula that is used in the code is;

$$\mathbf{i = m*a + c;}\text{……..(7)}$$

Where, i = output image, m = slope of line joining point (0,255), a= 1,2,3 (RGB) values.

c = intercept of the straight line with the axis.

m = 255/(rmax - rmin); (rmin and rmax are the min and max pixel in the image)

c = 255 - m*rmax;

The images below show a contrast stretching linear transform to add contrast to a poor quality image.

Fig 1.7(a): Before Piecewise Linear Transformation

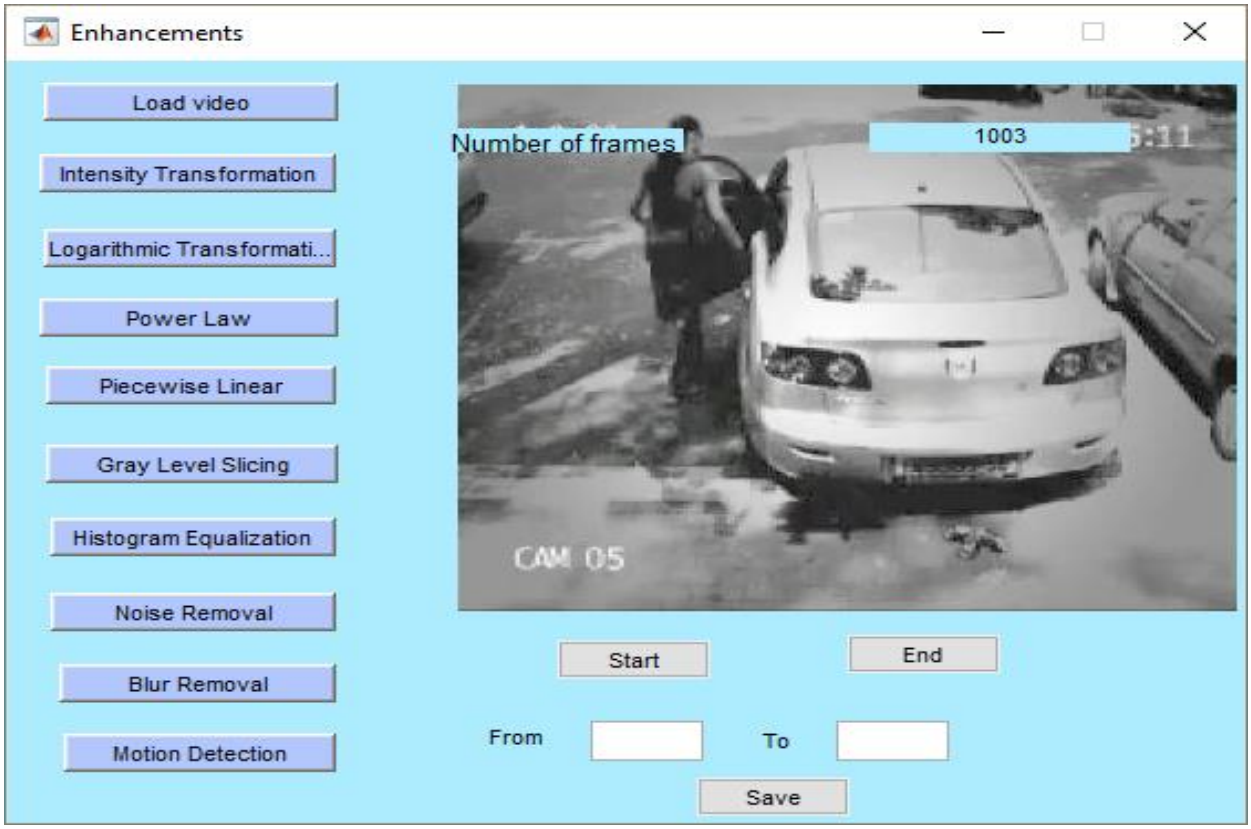

Fig 1.7(b): After Piecewise Linear Transformation

1.2.7 Histogram Equalization

Histogram equalization is a technique for adjusting image intensities to enhance contrast. Let f be a given image represented as a mr by mc matrix of integer pixel intensities ranging from 0 to L − 1. L is the number of possible intensity values, often 256. Let p denote the normalized histogram of f with a bin for each possible intensity.

p_n = number of pixels with intensity n / total number of pixels where,n = 0, 1, ..., L − 1.

The histogram equalized image g will be defined by [7];

$$g_{i,j} = \text{floor}((L-1)\ {}^{fi,j}\textstyle\sum_{n=0}\ p_n),\ldots\ldots\ldots\ldots\ldots\ldots(8)$$

where, floor() rounds down to the nearest integer. This is equivalent to transforming the pixel intensities, k, of f by the function. I have used built-in function of ‘histeq’ for histogram equalization.

Fig 1.8 (a): Before applying Histogram equalization

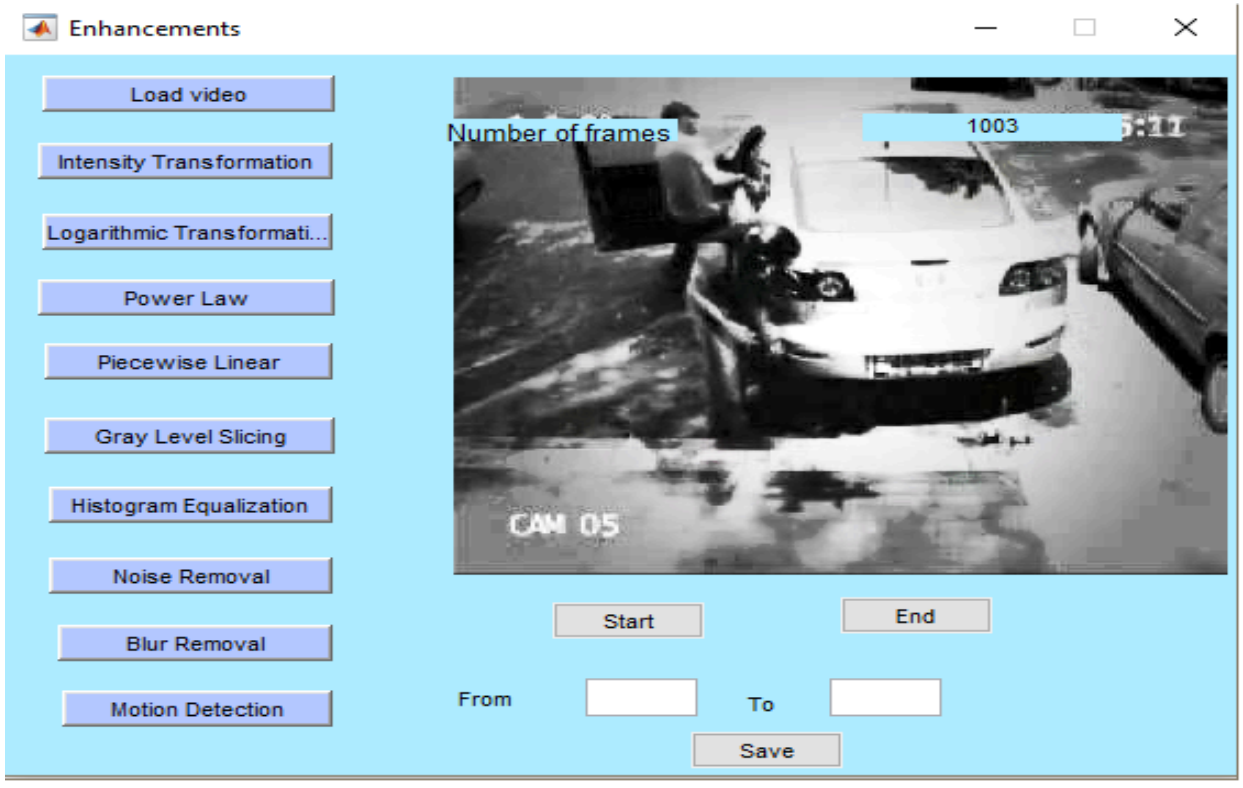

Fig 1.8 (b): Histogram Equalization applied

1.2.8 Motion Detection

SAD algorithm is used to detect the motion in the video. Sum of absolute differences (SAD) is a widely used as simple algorithm for measuring the similarity between image blocks. It works by taking the absolute difference between each pixel in the original block and the corresponding pixel in the block being used for comparison. These differences are summed to create a simple metric of block similarity. The sum of absolute differences may be used for a variety of purposes, such as object recognition.

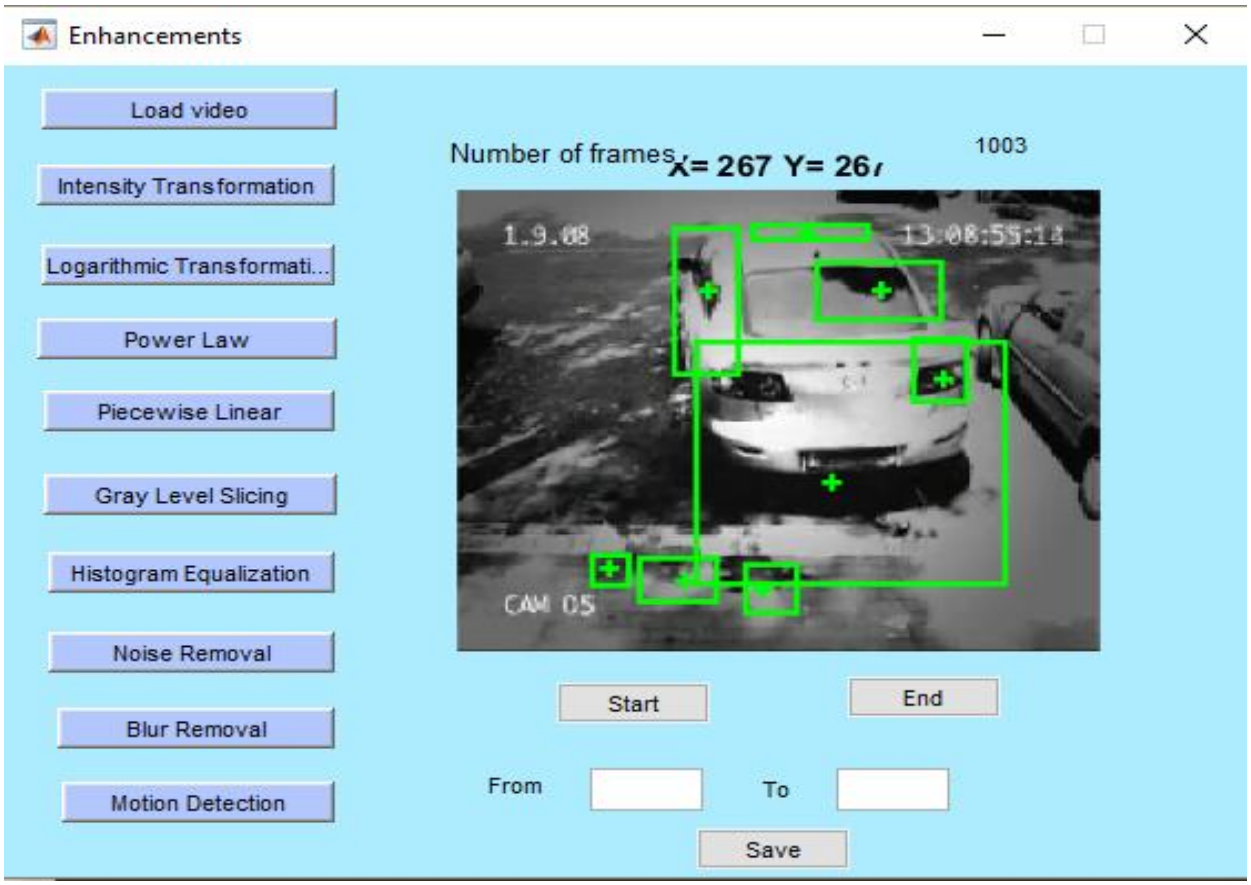

Fig 1.9: Motion Detection

CHAPTER 2
REQUIREMENT SPECIFICATION

This section of the document is specifying the general factors that affect the product and its requirements. It also describes the summary of the functions that the software will perform.

2.1 Stakeholders

Any person who has a stake or interest in any module of the developed system is a stakeholder. Our developed system's stakeholder is the user who is interacting with the system for CCTV footage enhancement.

2.2 Functional Requirements

Functional requirements are primarily the use cases that describe that what the system do. The functional requirement of the system is as follow:

Load Video:

System loads video from any drive on which system implements the algorithm .Video in MP4 and AVI are supported. For this process user will click on load video button.

Save Video:

Users can save the enhance video after implementation of algorithms. The saved video is stored in .MP4 format. The user can start and pause the video at any time after saving it.

Extraction of Frames:

System extracts frames of the video.

DESP: The application extracts frames from a video being load from drive and analyze those frames, also tells that how many frames are in the video. The frames are stored in .JPG format.

Intensity Enhancement:

System enhances intensity level of images/frames. They are especially useful for bringing out detail.

Logarithm Transformation:

System applies logarithm transformation on Frame/images.

DESP: Logarithmic Transformations can be used to brighten the intensities of an image. More often, it is used to increase the detail (or contrast) of lower intensity values.

Power law Transformation:

System also provides option for power law transformation on images/Frames. It is used for images enhancement.

Piecewise Linear Transformations:

System calculates the piecewise transformation of images and frames. Highlight a specific range of gray values.

Gray Level Slicing:

System provides gray scale level slicing of images/frames.

DESP: Two approaches (Gray Level Slicing)

1. Display high value for range of interest, low value else (‘discard background’**)**
2. Display high value for range of interest, original value else (‘preserve background’)

Histogram-Equalization:

System also provides histogram equalization process.

DESP: This can be implementing on extract frames from video or images to display more details in images. In some images, pixels values are not uniformly equalized. Histogram uniform these pixels values.

Comparison Algorithm:

System applies the best algorithms automatically which give good result on video.

DESP: It selects the algorithm that gives the best enhanced results on the basis of their Histogram values. The function compares all the histograms values of above mentioned algorithms and apply the best one.

Noise removal:

Users can also remove noise from images /extract frames from video. For noise removal user click on noise remove button to get result.

DESP: In this process system will remove noise from images /frames of video. Noise removal algorithm is the process of removing or reducing the noise from the image. The noise removal algorithm reduces visibility of noise by smoothing the entire image by leaving areas near contrast boundaries.

Blur removal:

System removes blur pixels from images/frames.

DESP: To remove the blurriness from the image/frame, I have implemented deconvlucy method, the deconvlucy algorithm is based on maximizing the likelihood of the resulting image with original image.

Motion Detection:

System detects the motion of object in the video.

DESP: Motion detection is a basic operation in the selection of significant segments of the video. Once motion has been detected, other features can be considered to decide what a video has to present to the users.

2.2.1 User Requirement

Following are the functionalities provided by the system to the user.

- Start the application.
- Upload footage.
- Extract the frame.
- Selects the enhancement algorithm.
- Close the application.

2.3 Non-Functional Requirements

Following are the non-functional requirements provided by the systems.

2.3.1 Performance

The system is efficient and able to classify the person without a noticeable delay.

2.3.2 Reliability

The application reliably classifies and recognizes the objects in the selected frame.

2.3.3 Security

The system can be allowed to certified users to interact with it.

2.3.4 Availability

The system does not require any internet connectivity and is be available once the application is running on a device.

2.3.5 Maintenance

The maintenance of the application will be carried out by the developers, if required.

2.4 Use Case Diagram

Use case is a powerful tool for describing scope and interaction of the user with the system. The use case in my application is presented in the following.

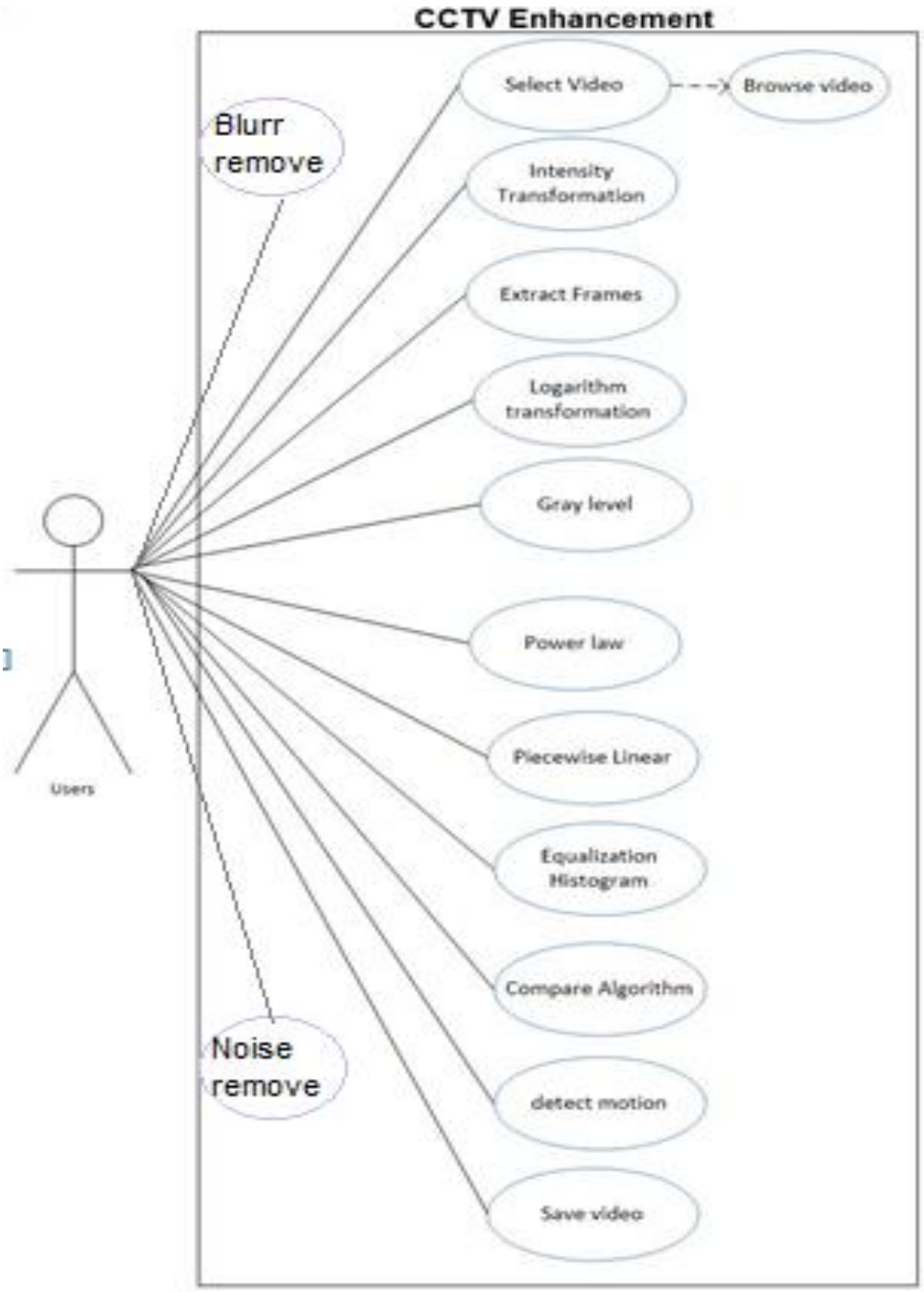

Fig:2.1: footage enhancement use case

2.4.1 UC 1: Select Video

Title	Footage Enhancement
Level	User goal
Use Case Description:	This use case describes how to select video.
Pre-condition:	System must be in running state.
Main Success Scenario	1. User enter the file name or browse. 2. User selects a video. 3. System opens selected video.
Extensions	1a. User enters a wrong file name. 1. User again writes the file name. 2. User selects the file.
Post-condition	Video selected.

2.4.2 UC 2: Save Video

Title	Footage Enhancement
Use Case Description:	This use case describes that how to save enhance video.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User selects the location to save the video. 2. Video stored in .MP4 format.
Extensions	1a. User enters wrong location to save the file.
Post-condition	Video saved successfully.

2.4.3 UC 3: Intensity Transformation

Title	Intensity Transformation
Use Case Description:	This use case describes that how to implement Intensity Transformation algorithm on upload video
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User enter the file name or browse. 2. User selects a video. 3. System opens selected video. 4. System uploads the video. 5.System will apply the intensity transformation algorithm on video. 6.System will show the enhance result to users.
Extensions	1a. Intensity transformation not implement on video. 2.Video not found.
Post-condition	System will successfully implement intensity algorithm on video.

2.4.4 UC 4: Extract Frames

Title	Extract Frames
Use Case Description:	This use case describes that how to extract frames from video.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User enter the file name or browse. 2. User selects a video. 3. System opens selected video. 4. System uploads the video. 5.System will extract the frames from video. 6.Frames will be extracted.
Extensions	1a. User extract frames from video. Frames will not be extracted User again writes the file name. 1. User selects the file. 2. System will extract the frames
Post-condition	Frames will be successfully extracted from video.

2.4.5 UC 5: Logarithm transformation

Title	Logarithm transformation
Use Case Description:	This use case describes that how Logarithm transformation implements on video.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User selects a video. 2. System opens selected video. 3. System uploads the video. 4. System will apply the Logarithm transformation algorithm on video. 5.system will show the enhance result to users
Extensions	1a. Logarithm transformation not implement on video 2. Video not found.
Post-condition	System will successfully implement Logarithm algorithm on video.

2.4.6 UC 6: Gray- Level Slicing

Title	Gray level
Use Case Description:	This use case describes that how to implement Gray level on video.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User selects a video. 2. System opens selected video. 3. System uploads the video. 4. System apply the Gray level slicing
Extensions	1a. Gray level not implemented on video 2. Video not found..
Post-condition	System successfully implemented Gray level slicing on video.

2.4.7 UC 7: Power law

Title	Power law
Use Case Description:	This use case describes that how to implement power law on video.
Pre-condition:	System must be in running state.

Level	User goal
Main Success Scenario	1. User selects a video. 2. System opens selected video. 3. System uploads the video. 4. System will apply the Power law transformation algorithm on video. 5.system will show the enhance result to users
Extensions	1a. Power law transformation not implement on video 2. Video not found.
Post-condition	System will successfully implement power law algorithm on video.

2.4.8 UC 8: Piecewise Linear

Title	Piecewise Linear
Use Case Description:	This use case describes that how to implement Piecewise Linear on video.
Pre-condition:	System must be in running state.
Level	User goal

Main Success Scenario	1. User selects a video. 2. System opens selected video. 3. System uploads the video. 4. System will apply the Piecewise Linear algorithm on video. 5.system will show the enhance result to users
Extensions	1a. Piecewise Linear not implement on video 2. Video not found.
Post-condition	System will successfully implement Piecewise on video.

2.4.9 UC 9: Histogram Equalization

Title	Histogram Equalization
Use Case Description:	This use case describes that how apply Equalize histogram on video/Frames and compare the result.
Pre-condition:	System must be in running state.
Level	User goal

Main Success Scenario	1. User selects a video. 2. System will start selected video. 3. System will apply the equalize histogram on video/frames of video. 4.system will show the enhance result to users
Extensions	1a. Equalize histogram not implement on video/Frames 2. Video not found.
Post-condition	System will successfully implement histogram equalization on video/frames of video.

2.4.10 UC 10: Compare Algorithm

Title	Compare Algorithm
Use Case Description:	This use case describes that system will select Compare algorithm apply on video/Frames and compare the result.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User selects a video. 2. System will start selected video.

	3. System will apply the best algorithm automatically which gives good results on video/frames of video. 4.system will show the enhance result to user
Extensions	1a. System will not apply the be algorithm automatically on video/Frames 2. Users implement it again frames of video/video.
Post-condition	System will successfully implement best algorithm automatically video/frames of video.

2.4.11 UC 11: Noise and blur Removal

Title	Noise and blur Removal
Use Case Description:	This use case describes that how remove noise and blur pixel's from video/frames of video.
Pre-condition:	System must be in running state.
Level	User goal
Main Success Scenario	1. User selects a video. 2. System will start selected video. 3. Users select noise removal algorithm to

	implement on video/frames of video. 3. System will apply noise removal algorithm on video/frames of video. 4. Users select blur removal algorithm to implement on video/frames of video. 5. System will apply blur removal algorithm on video/frames of video 4.System will show the enhance result to users
Extensions	1a. Noise removal not implement on video/Frames 2a. Blur removal not implement on video/Frames 3. Video not found.
Post-condition	System will successfully remove noise and remove blur on video/frames of video.

2.4.12 UC 12: Motion detection

Title	Motion detection
Use Case Description:	This use case describes that how detect motion of object in the video.
Pre-condition:	System must be in running state.

Level	User goal
Main Success Scenario	1. User selects a video. 2. System will start selected video. 3.User will select motion detection algorithm. 4. System will apply the motion detection on video/frames of video. 5.system will show the moving objects to user.
Extensions	1a. Motion detection not implement on video/Frames 2. Video not found.
Post-condition	System will successfully implement motion detection on video/frames of video.

CHAPTER 3
SOFTWARE DESIGN AND DESCRIPTION

This chapter specifies the general factors that can affect the product and its requirements, providing a background for the requirements of the software. To describe the key activities that will be performed by the system, the following diagrams have been used system sequence diagram, activity diagram and state transition diagram.

Software Design Description (SDD) is a representation of a system that how the users can interact with the system. This can be shown with the help of diagrams. The SSD shows that the system completely fulfills the requirements.

3.1 Design Overview

In the design overview, we describe the system at architecture level and the complete structure of the system. It includes the explanation that how the user interacts with the system.

3.2 Architecture Diagram

Architecture Diagram is used to represent the components of system and interaction between them. System under discussion is based on “Three Tier” Architectural pattern. Three-tier architecture is a software architecture pattern in which the user interface (presentation), functional process logic (business rules), computer data storage and data access are developed and maintained as independent modules Interacting between components of system is shown in diagram. Double arrows represent the interaction from both sides. Similarly single arrow represents one way interaction. The singular quality of three-tier architecture is the separation of the application logic into a distinct logical middle tier of software. The interface tier is relatively free of application processing. The middle tier communicates with the back-end storage layer. It is possible to make changes on the presentation level without affecting the other two (business or data access layer).

As each tier is independent it is possible to use different sets of developers since the user doesn't have direct access to the database business logic, making it are more secure.

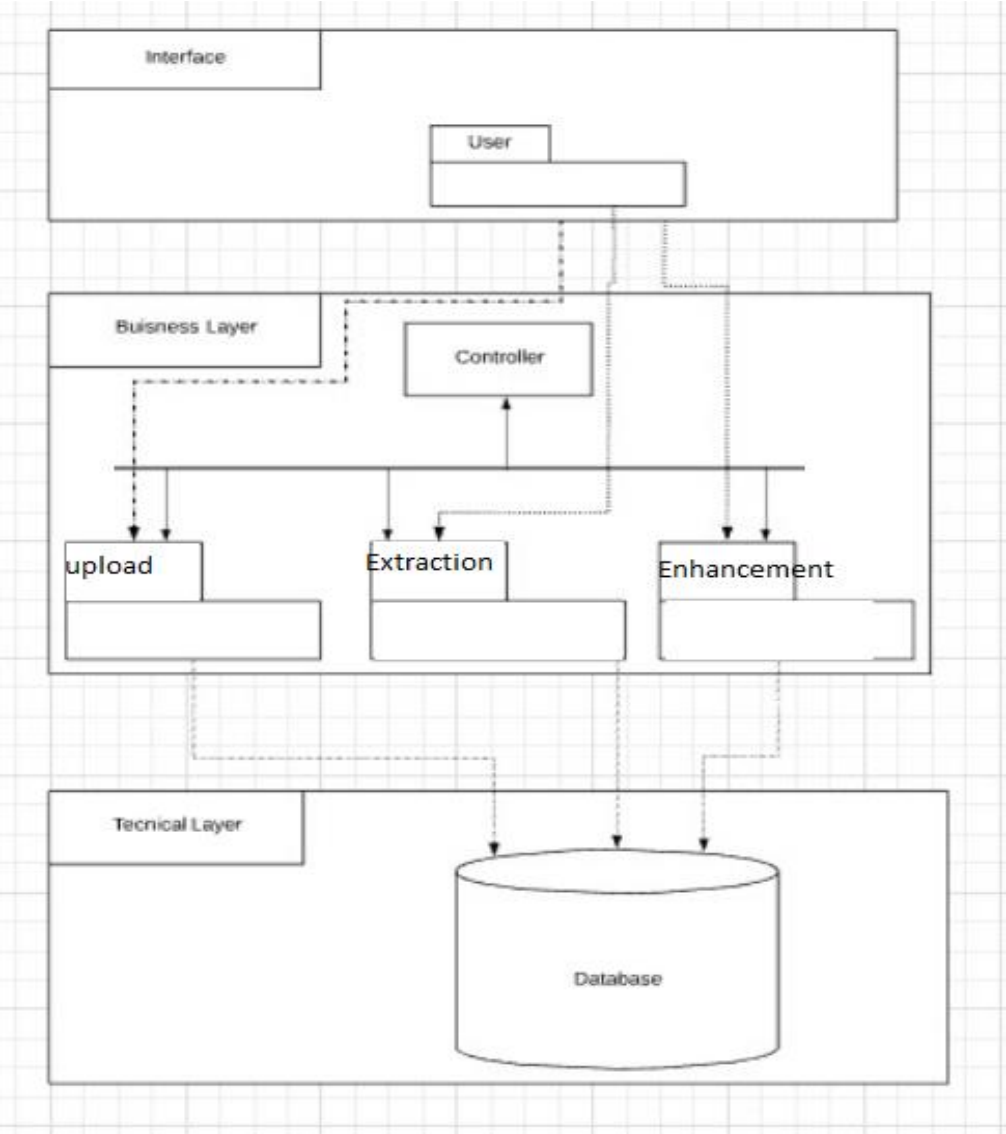

Fig 3.1: Architecture Diagram

3.3 Interface design

User Interface Design is the process of crafting a visual language and hierarchy that allows someone to use and engage an application. It focuses on the user's experience and interaction.

3.3.1 Simple and Appealing

The interface is simple to use, a native user can also use it very easily and efficiently to get their required results. Different buttons are used that makes application appealing.

3.3.2 Responsive

The interface is responsive which means that the size of display is fixed according to the screen size. (Small screens, Large screens etc).

3.4 Interfaces

A user interface, consisting of the set of dials, knobs, operating system commands, graphical display formats, and other devices provided by a computer or a program to allow the user to communicate and use the computer or program. A graphical user interface (GUI) provides its user a more or less "picture-oriented" way to interact with technology. A GUI is usually a more satisfying or user-friendly interface to a computer system.

3.4.1 Interface upload video

Following picture will show the interface of uploading a video, which is being used in the project. By clicking on load video button which is at the top of GUI another window will appear to select the video or browse it.

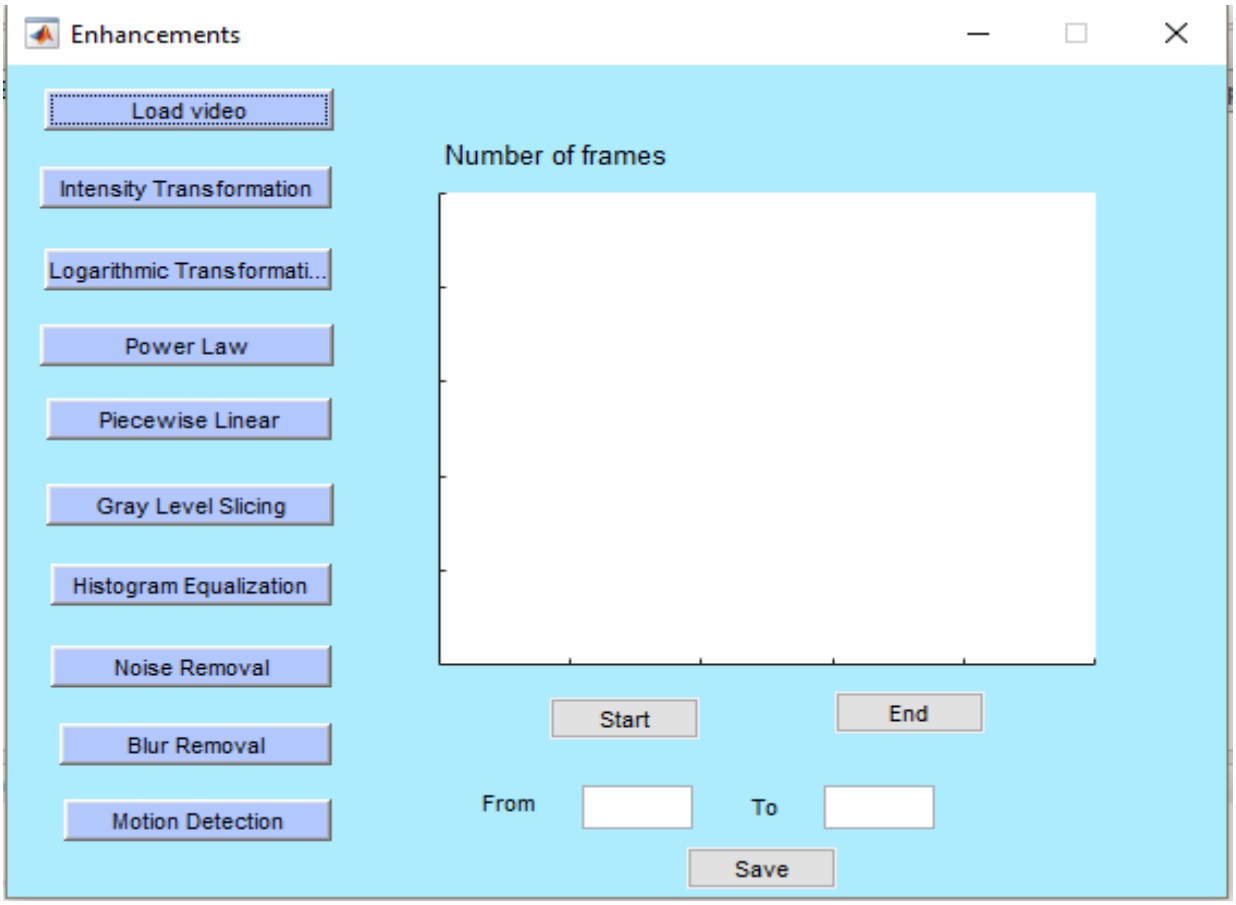

Figure 3.2: Interface upload video

3.4.2 Select Video interface

Following picture will show the interface of selecting a video, which is being used in the project. User can either enter the name of the file or can browse if from different drives. After the selecting the video click on OPEN button to display the video.

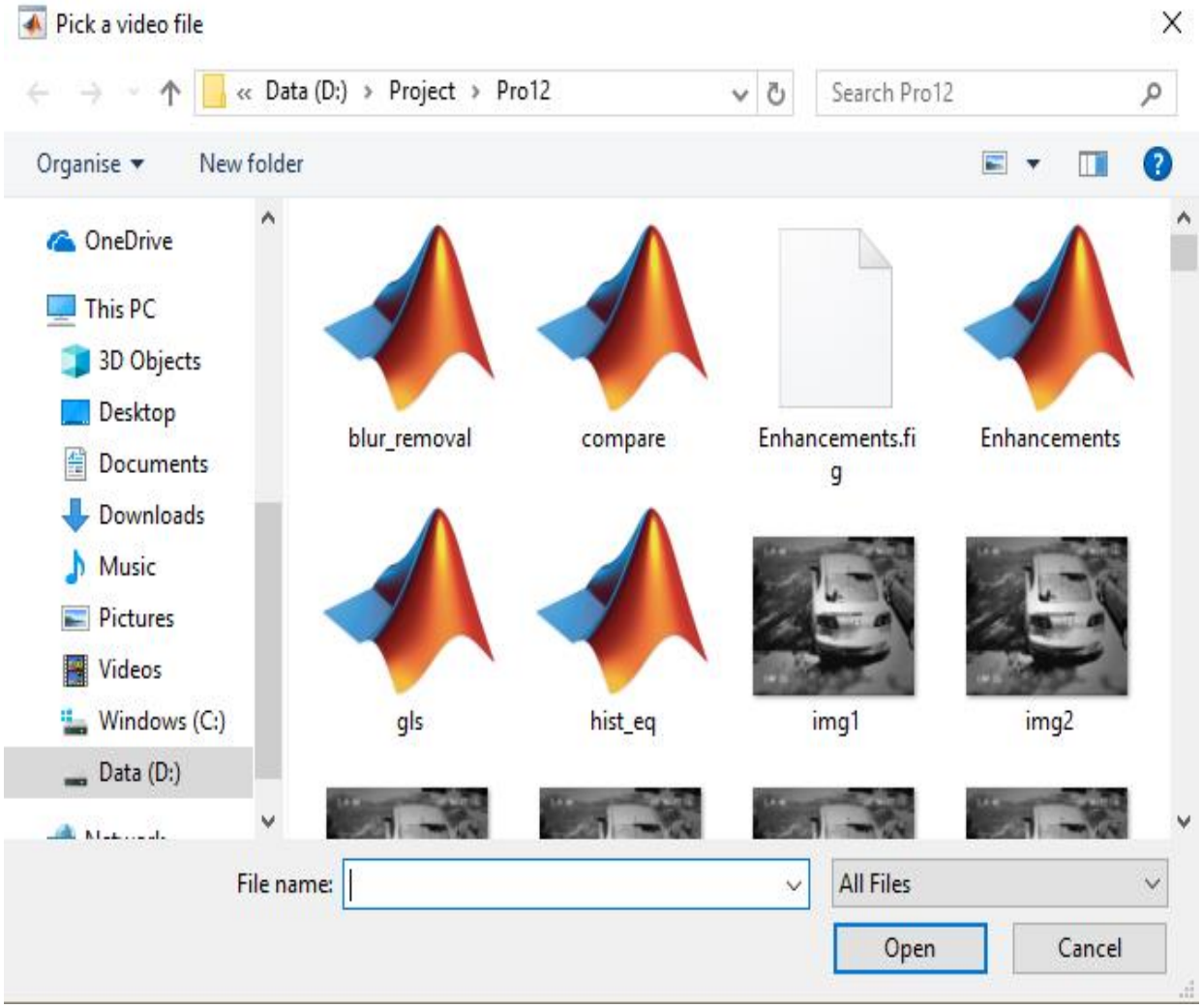

Fig 3.3: Interface Select video

3.4.3 Select Enhancement Algorithm

Following picture will show the interface of selecting the enhancement algorithm, which is being used in the project. It contains different image enhancement techniques in the left mot column. User will simply click on the required algorithm technique and it will be executed and start showing results.

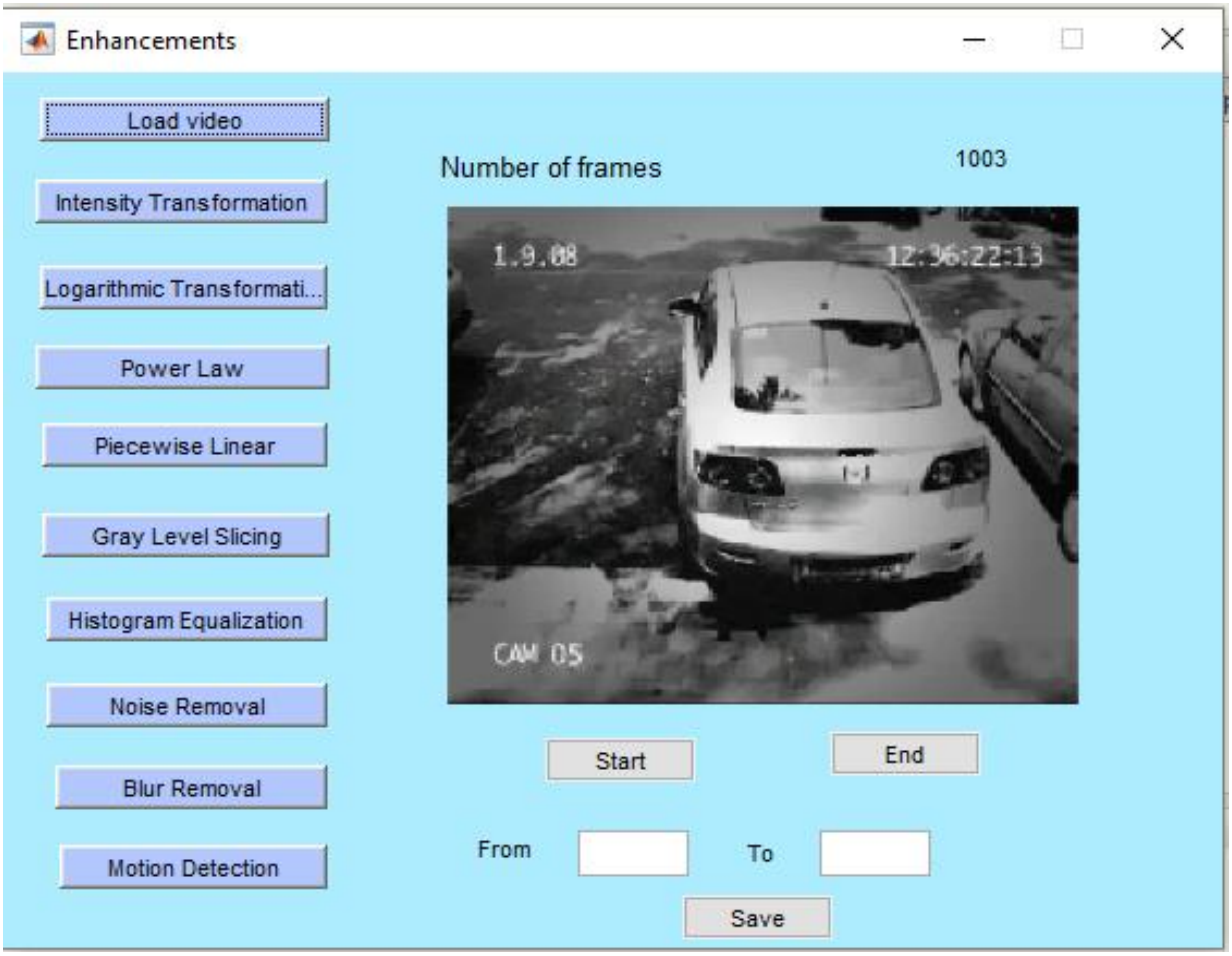

Figure 3.4: Interface Selection for Enhancement Technique

3.5 Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. They are used to describe the steps in a use case diagram. Activities modeled can be sequential and concurrent. In both cases an activity diagram will have a beginning and end. Major activity diagrams related to my project includes;

3.5.1 Select a video

This diagram will show the activities that are being performed in selecting a video.

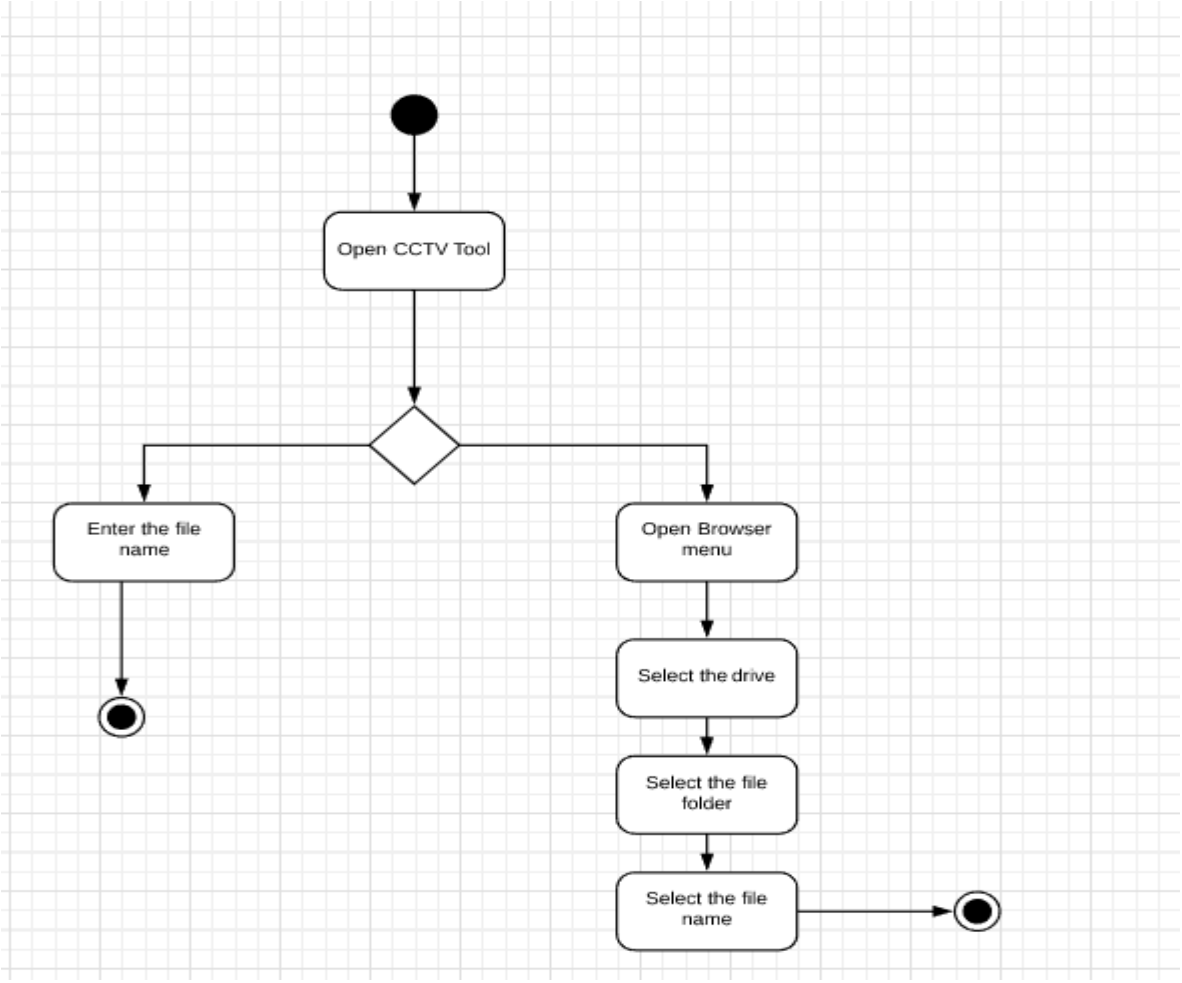

Fig 3.5: Activity diagram of selecting a video

3.5.2 Upload a file

This diagram will show the activities that are being performed in uploading a video.

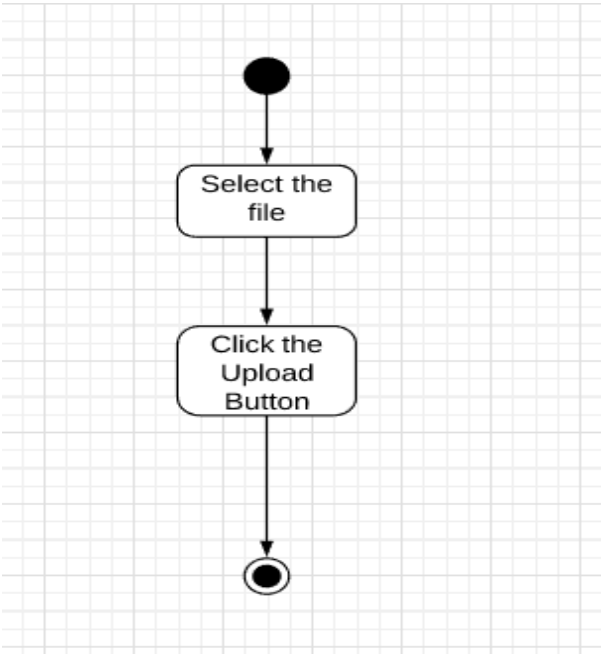

Fig 3.6: Activity diagram of uploading a video

3.5.3 Extraction of frame

This diagram will show the activities that are being performed in extracting frame(s) from a video.

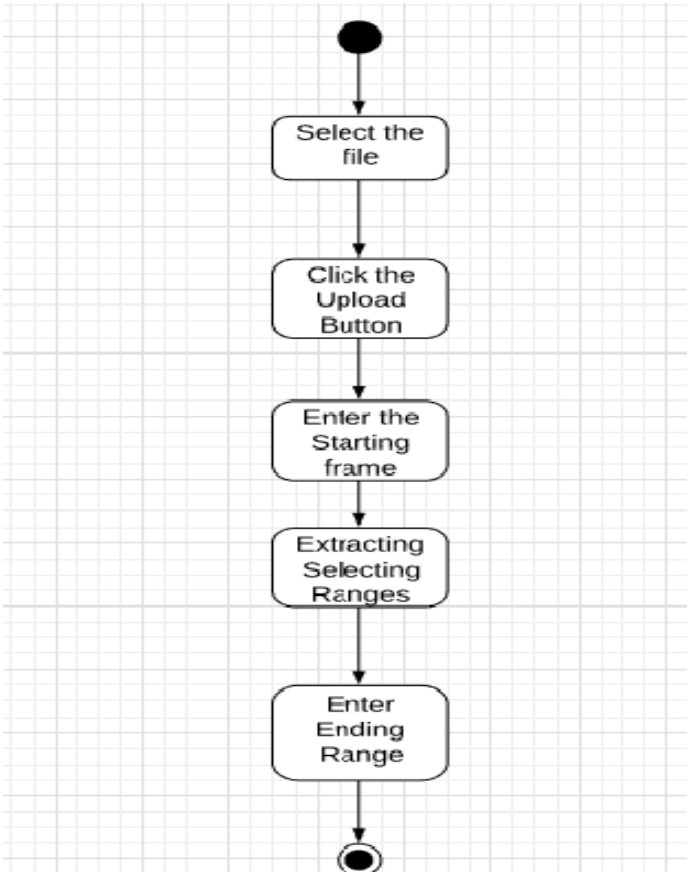

Fig 3.7: Activity diagram of extracting the frame

3.5.4 Enhancement of frame

This diagram will show the activities that are being performed in enhancing the frame(s) of video.

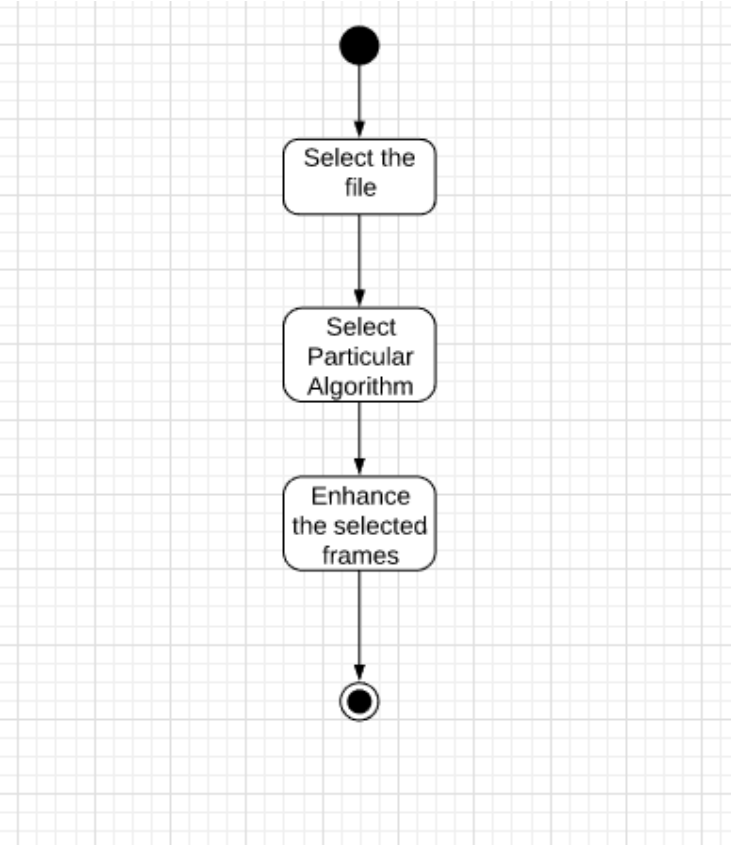

Fig 3.8: Activity diagram of enhancing frame

3.6 Data Flow Diagram (DFD)

A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented.

Different type of DFD`s includes;

- Context-level DFD.
- 1st level DFD.
- 2nd level DFD.

3.6.1 Context level DFD

A context level data flow diagram (DFD) provides an at-a-glance look at an information system and the ways it exchanges data with outside entities. They are often used for high-level planning.

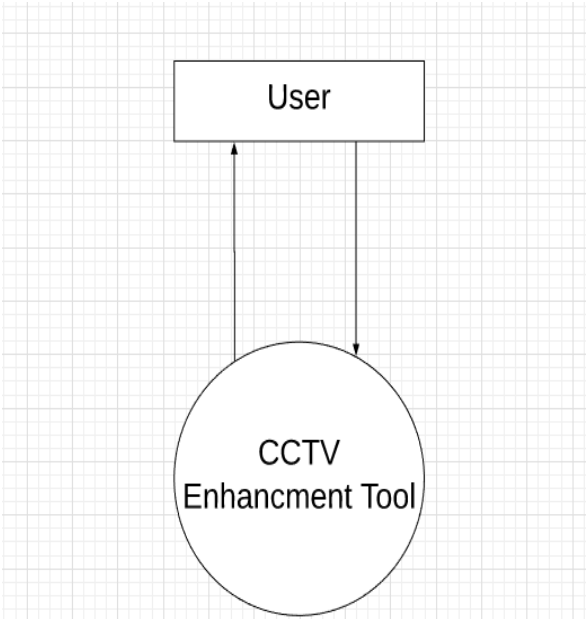

Fig 3.9: Context level dfd

3.6.2 1st level DFD

A level 1 data flow diagram (DFD) is more detailed than a level 0 DFD but not as detailed as a level 2 DFD. It breaks down the main processes into sub processes that can then be analyzed and improved on a more intimate level.

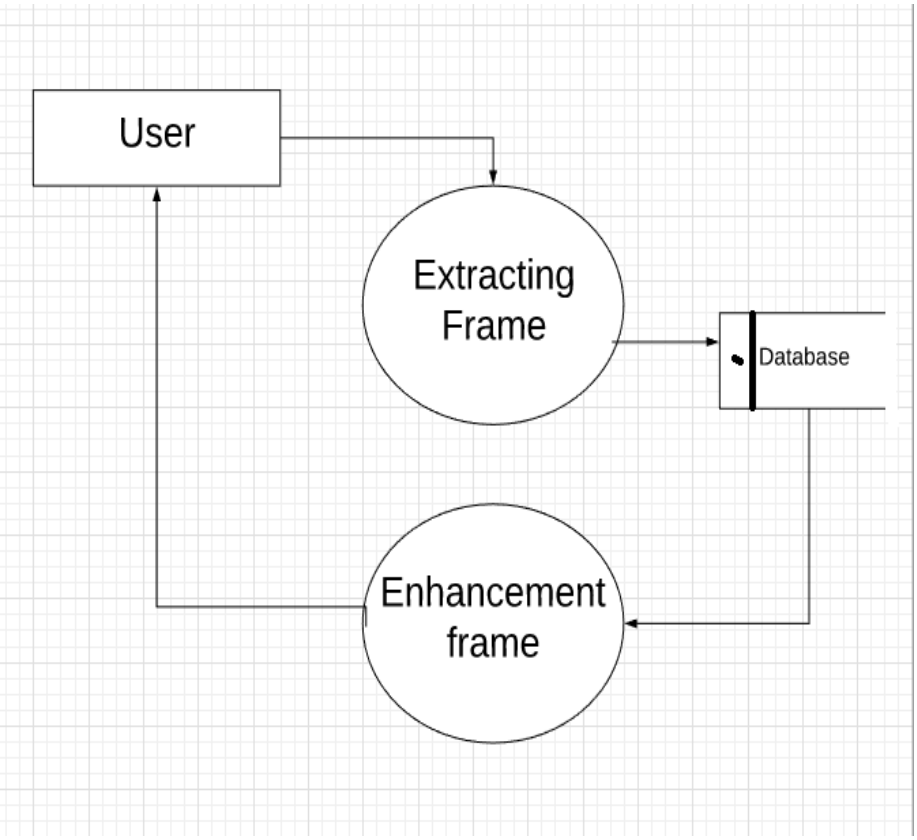

Fig 3.10: 1st level dfd

3.6.3 2nd level DFD

A level 2 data flow diagram (DFD) offers a more detailed look at the processes that make up an information system than a level 1 DFD does. It can be used to plan or record the specific makeup of a system.

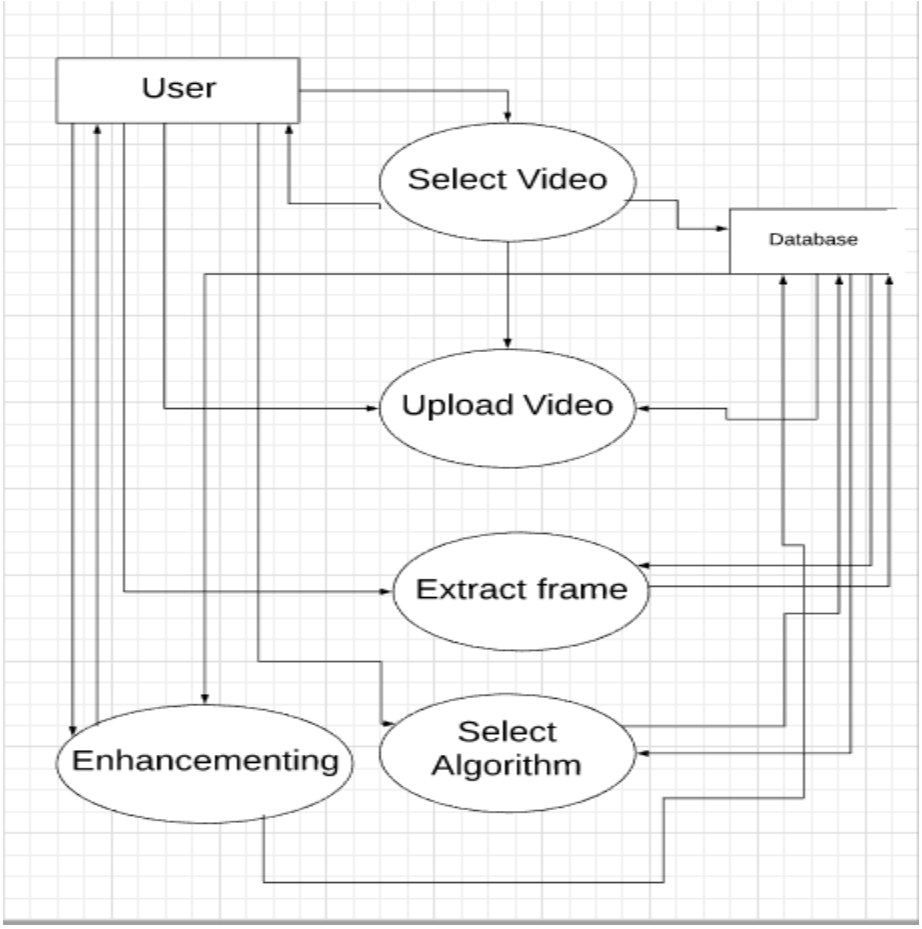

Fig 3.11: 2nd level dfd

3.7 System Sequence Diagram

A system sequence diagram illustrates input and output events related to our system. System is treated as a black box and the emphasis of the diagram is events that are generated by system for a particular scenario of a use-case.

3.7.1 SSD for Select video

SSD for selecting the video is given below.

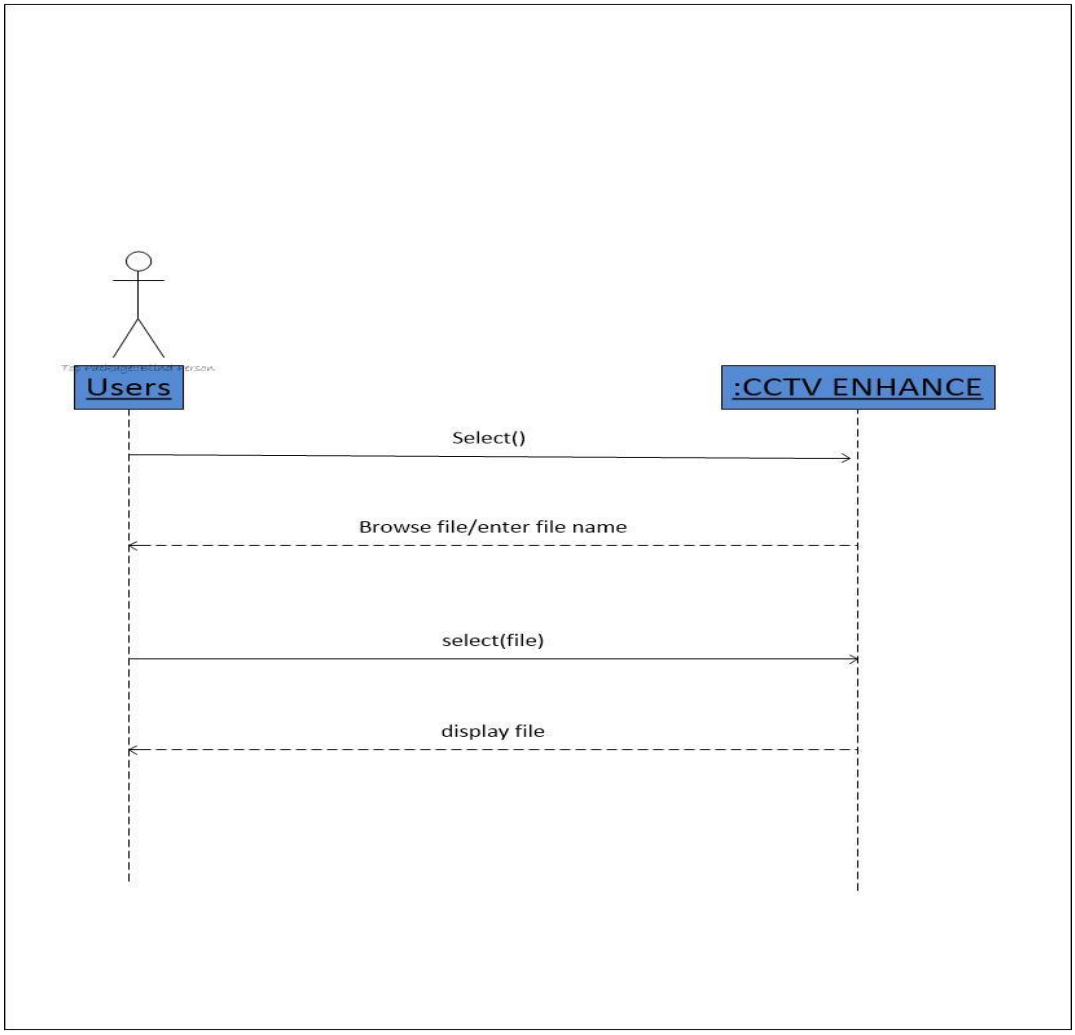

Fig 3.12: SSD for selecting video

3.7.2 SSD for Enhancing frame

SSD for enhancing the selected frames is given below.

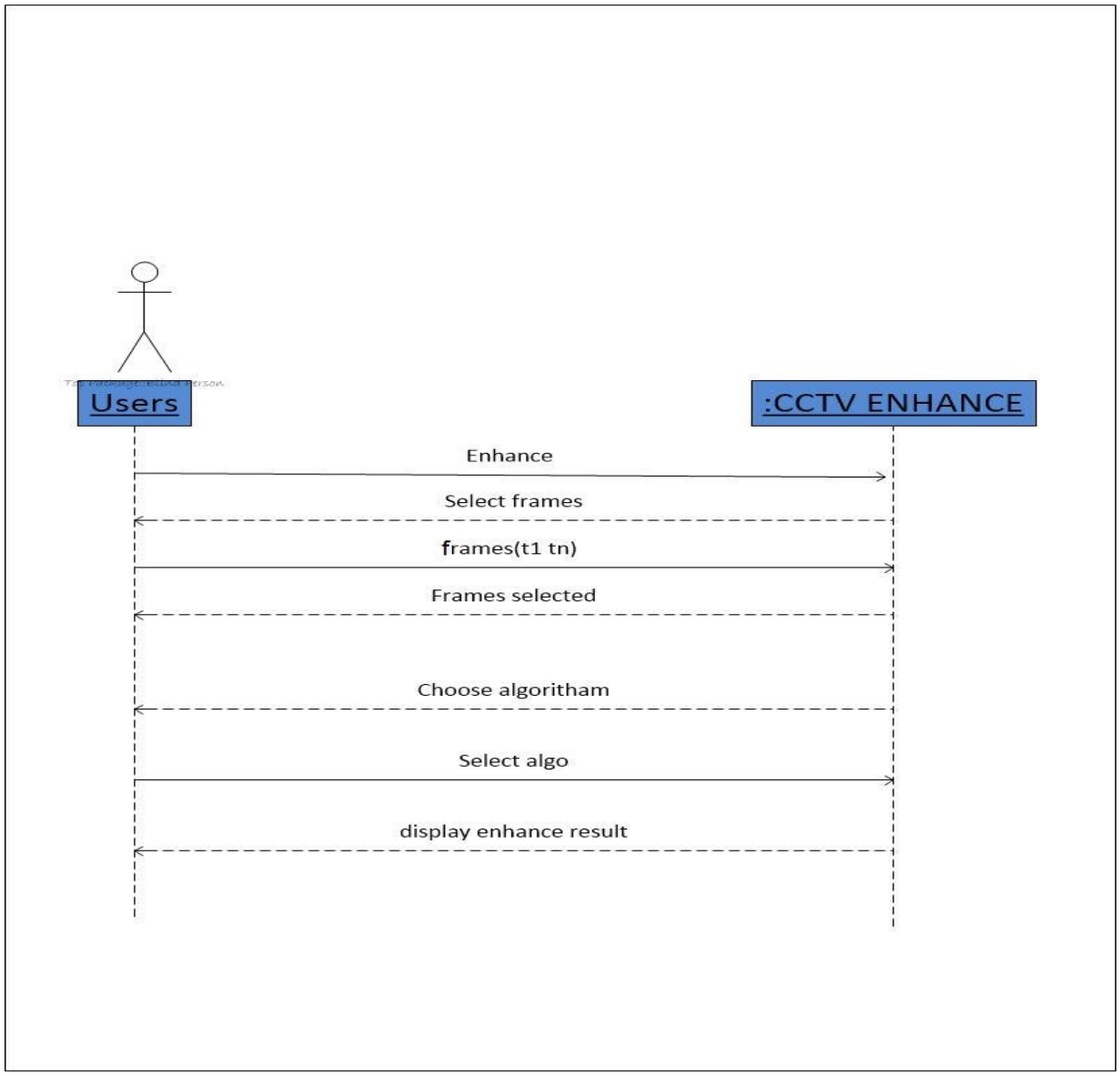

Fig 3.13: SSD for enhancing frame

3.7.3 SSD for Intensity Transformation

SSD for intensity transformation is given below.

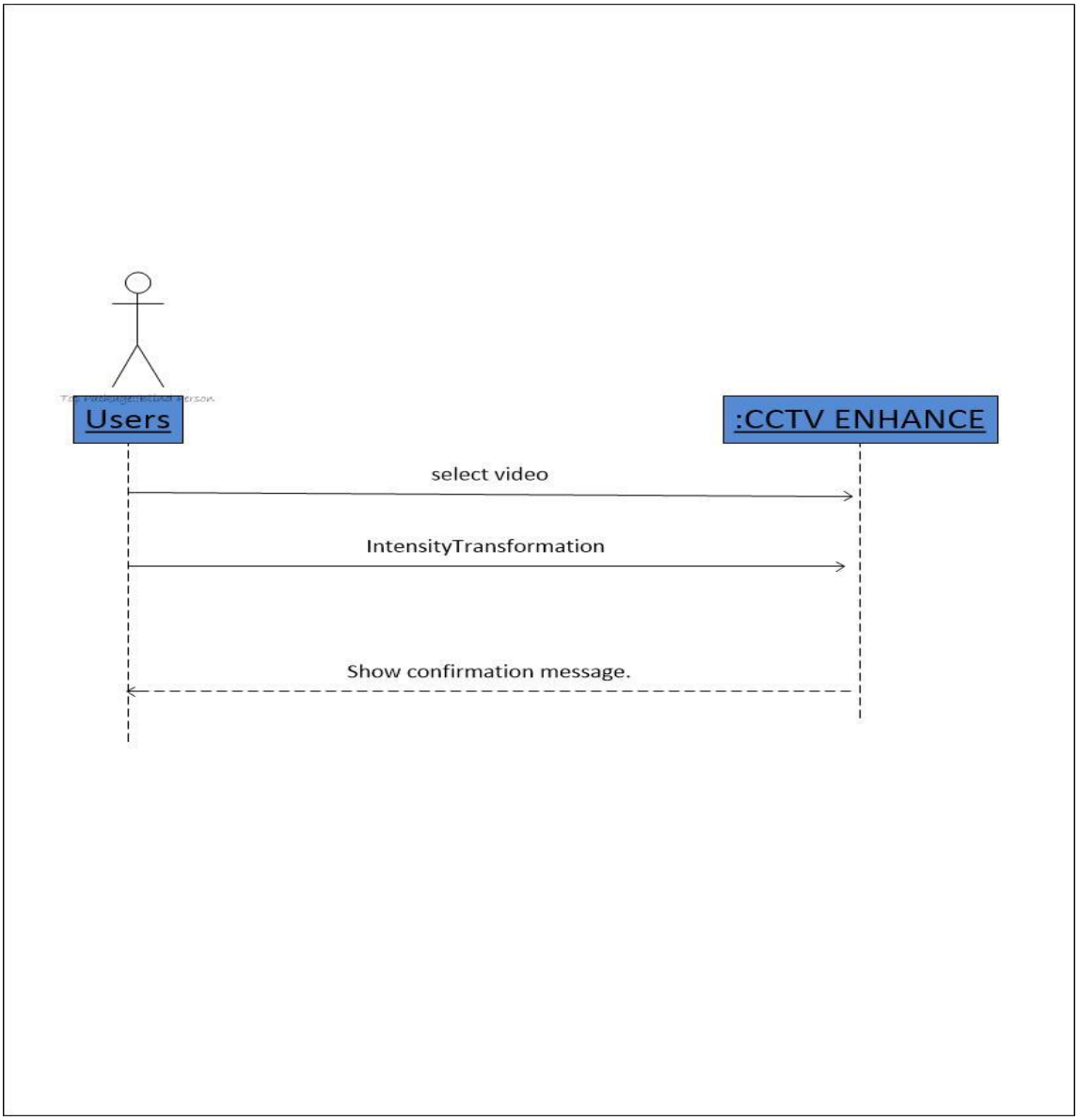

Fig 3.14: SSD for intensity transformation

3.7.4 SSD for Logarithmic Transformation

SSD for logarithmic transformation is given below.

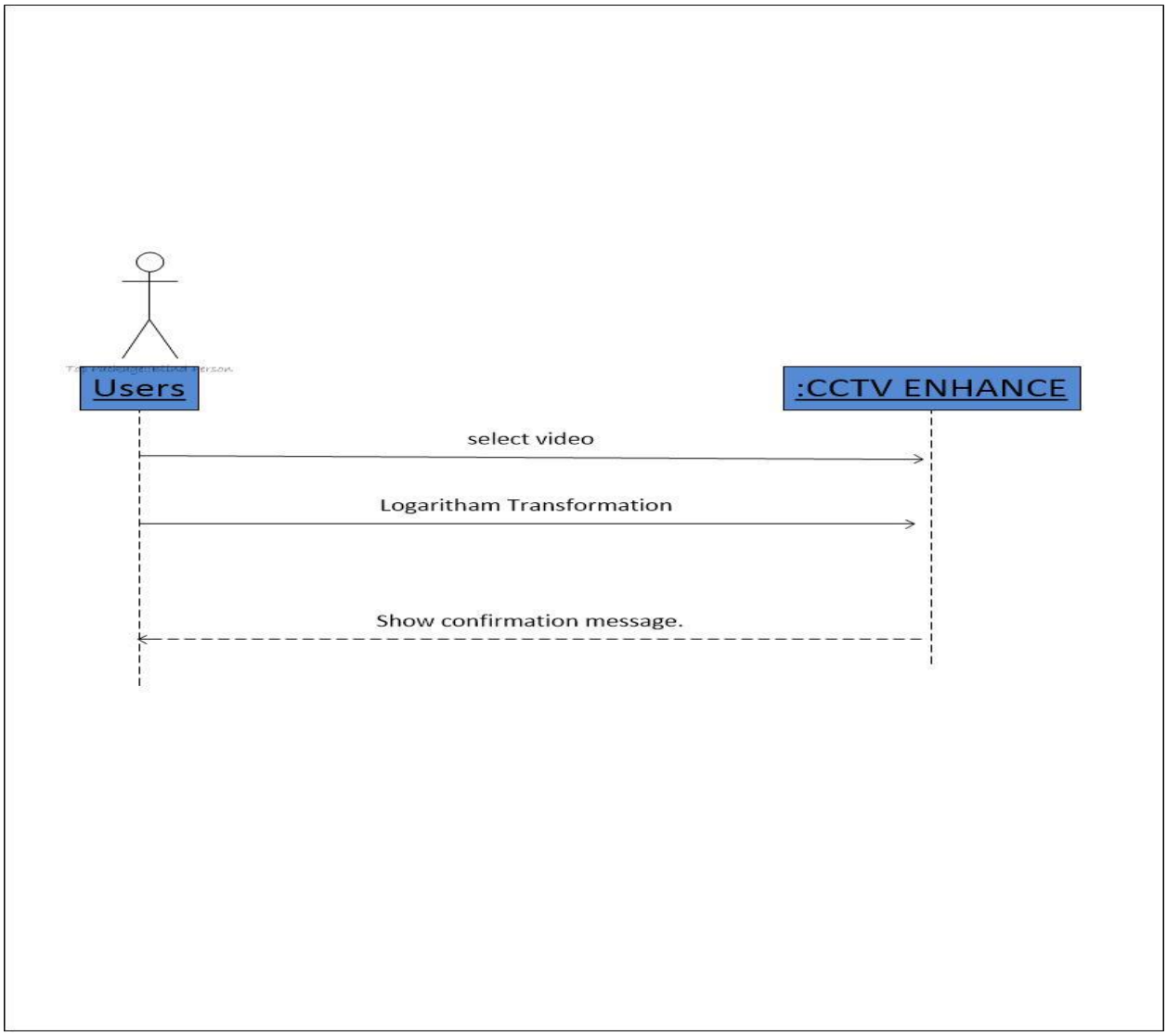

Fig 3.14: SSD for Logarithmic transformation

3.7.5 SSD for Power Law

SSD for power law is given below.

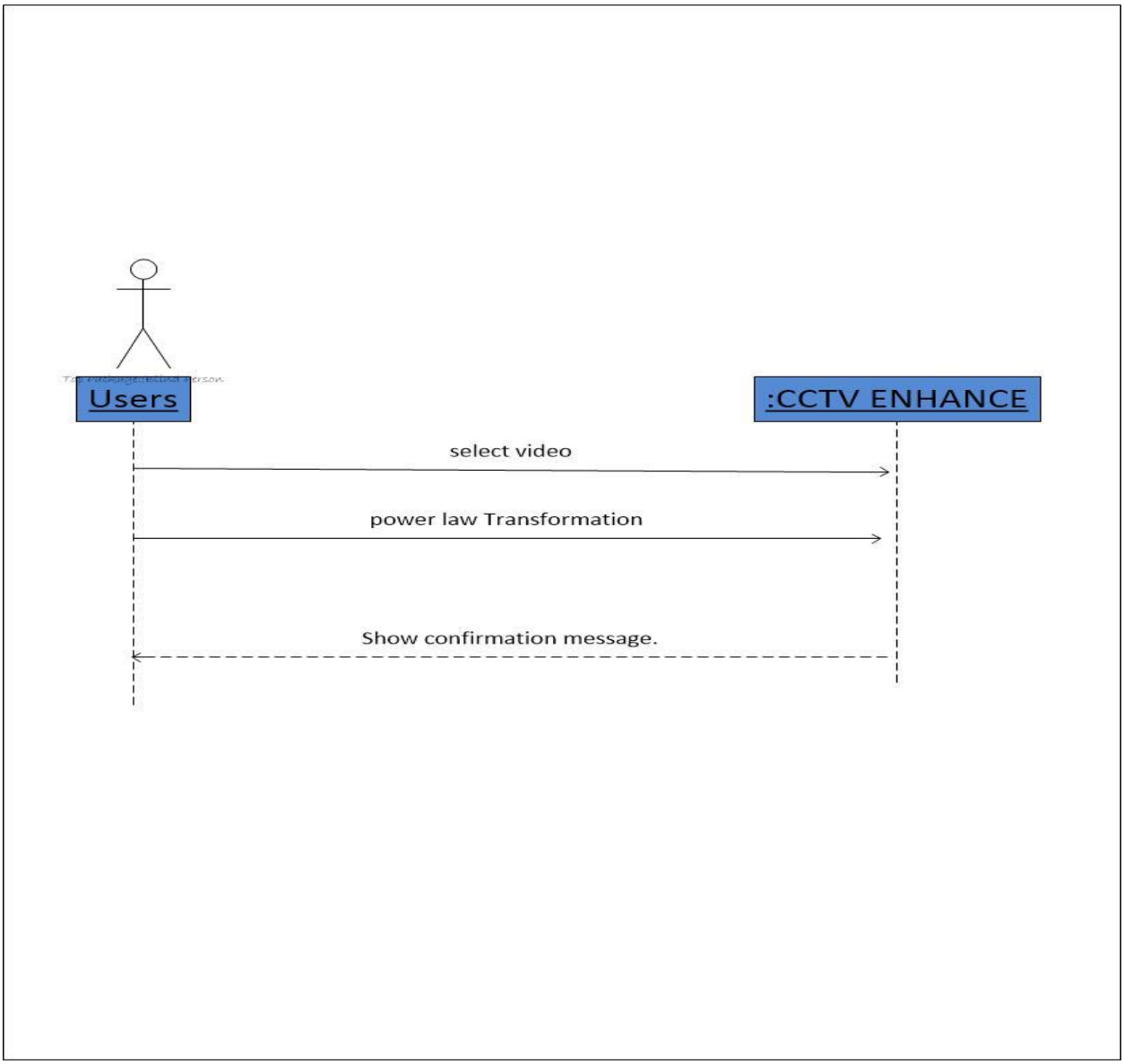

Fig 3.15: SSD for power law transformation

3.7.6 SSD for Piece-wise Linear

SSD for piece-wise linear transformation is given below.

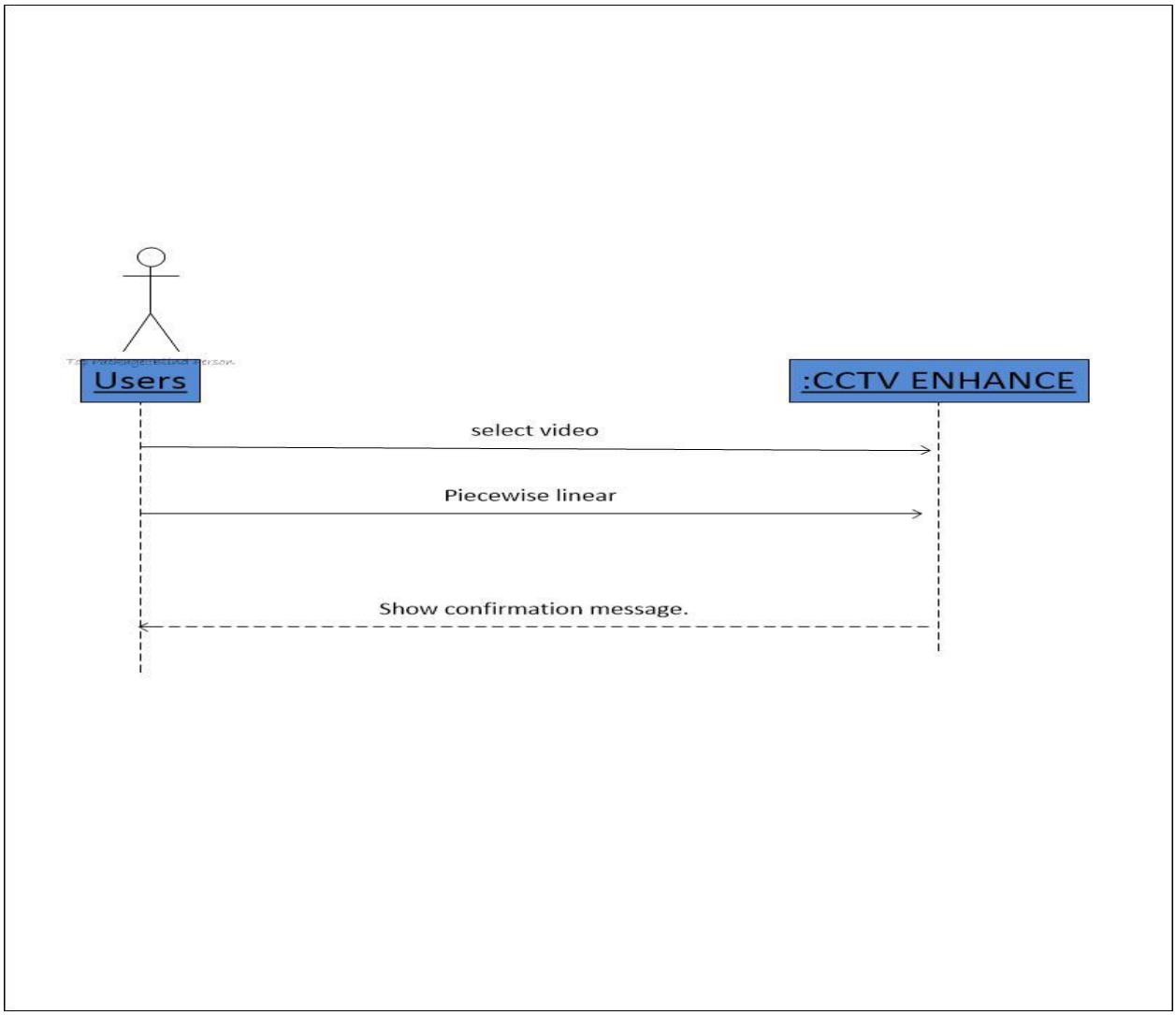

Fig 3.16: SSD for piecewise transformation

3.7.7 SSD for Histogram Equalization

SSD for histogram equalization is given below.

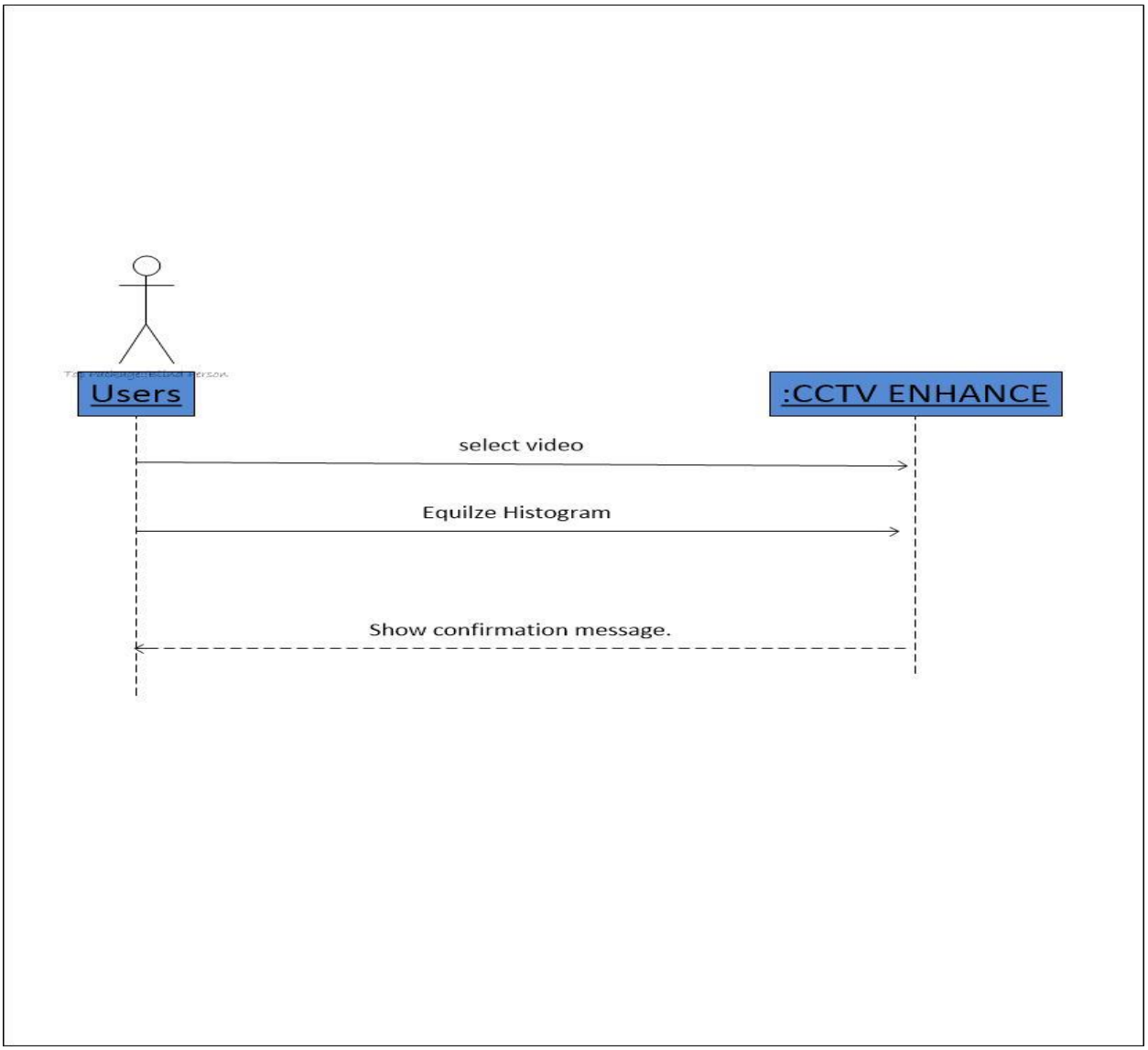

Fig 3.17: SSD for histogram equalization

3.7.8 SSD for Comparison

SSD for comparison functionality is given below.

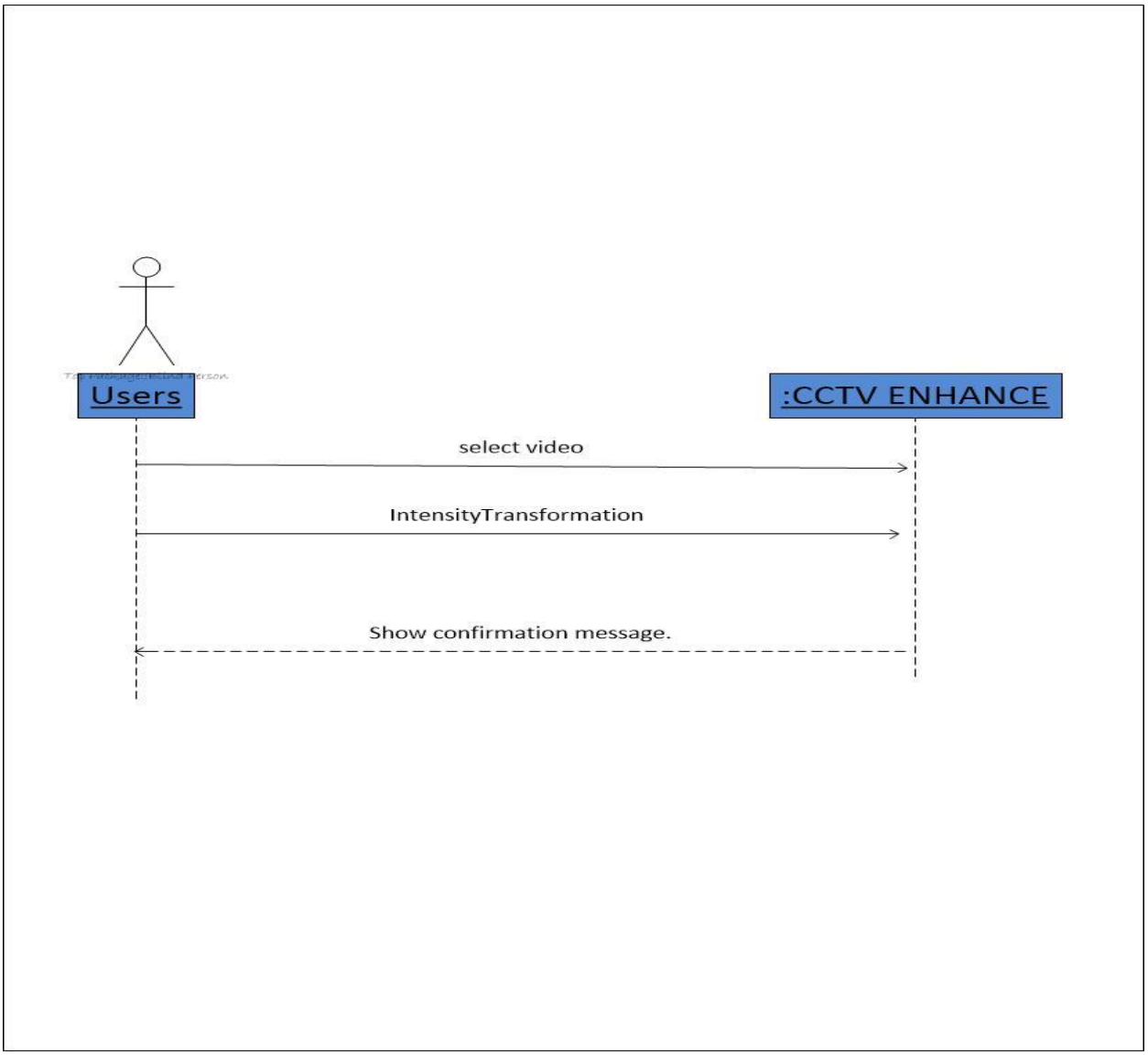

Fig 3.18: SSD for comparison of intensities

3.7.9 SSD for Motion Detection

SSD for motion detection is given below.

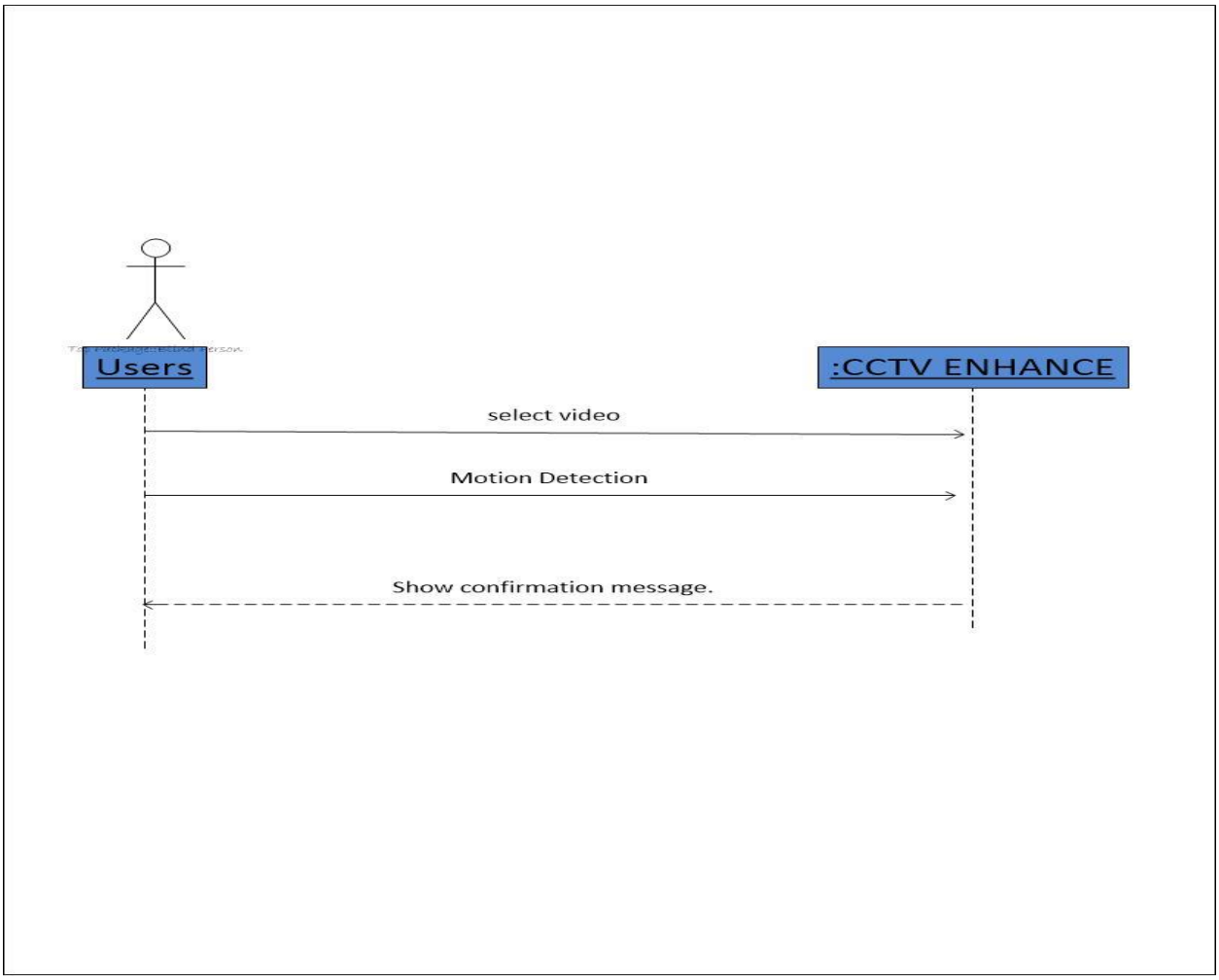

Fig 3.19: SSD for motion detection

3.7.10 SSD for Saving Video

SSD saving enhanced is given below.

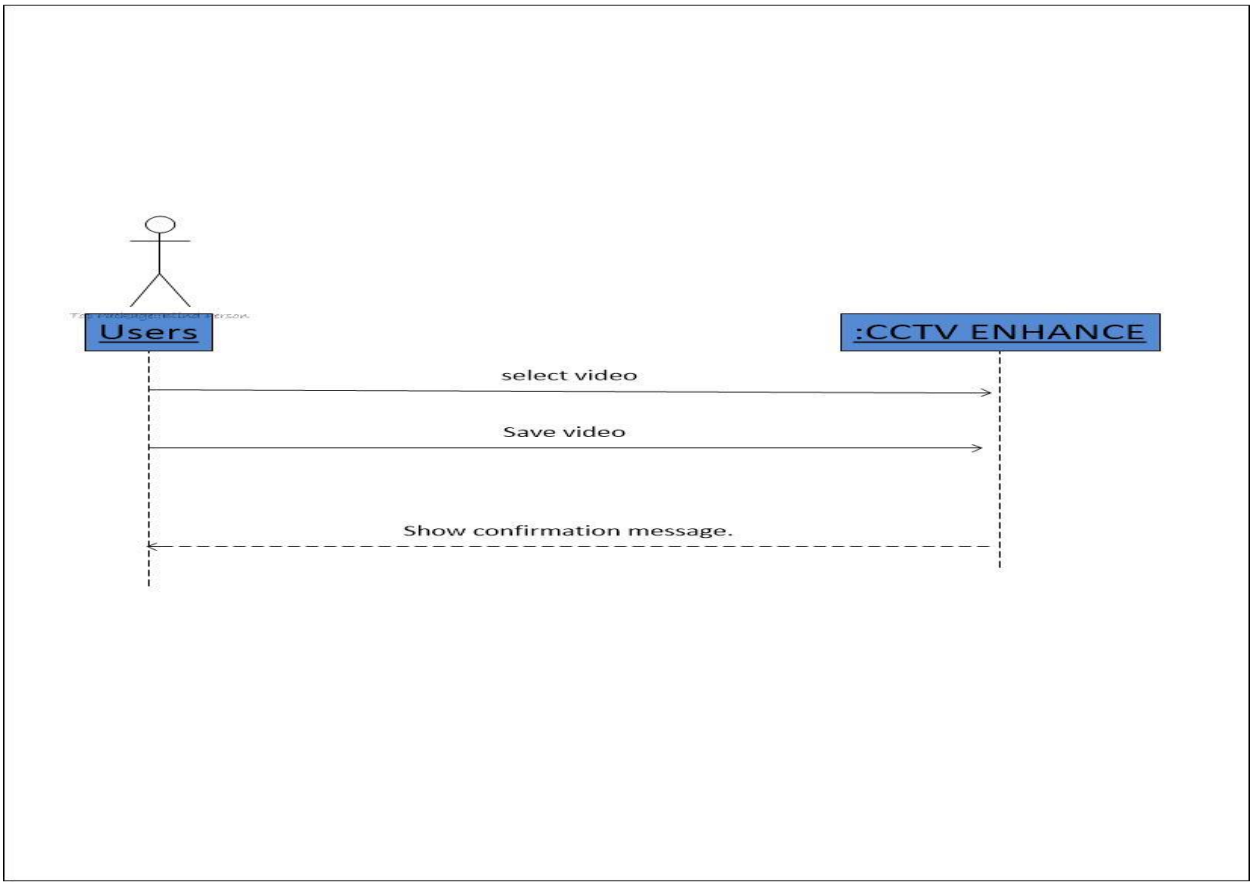

Fig 3.20: SSD for saving enhanced video

3.7.11 SSD for Gray level slicing

SSD for Gray level slicing is given below.

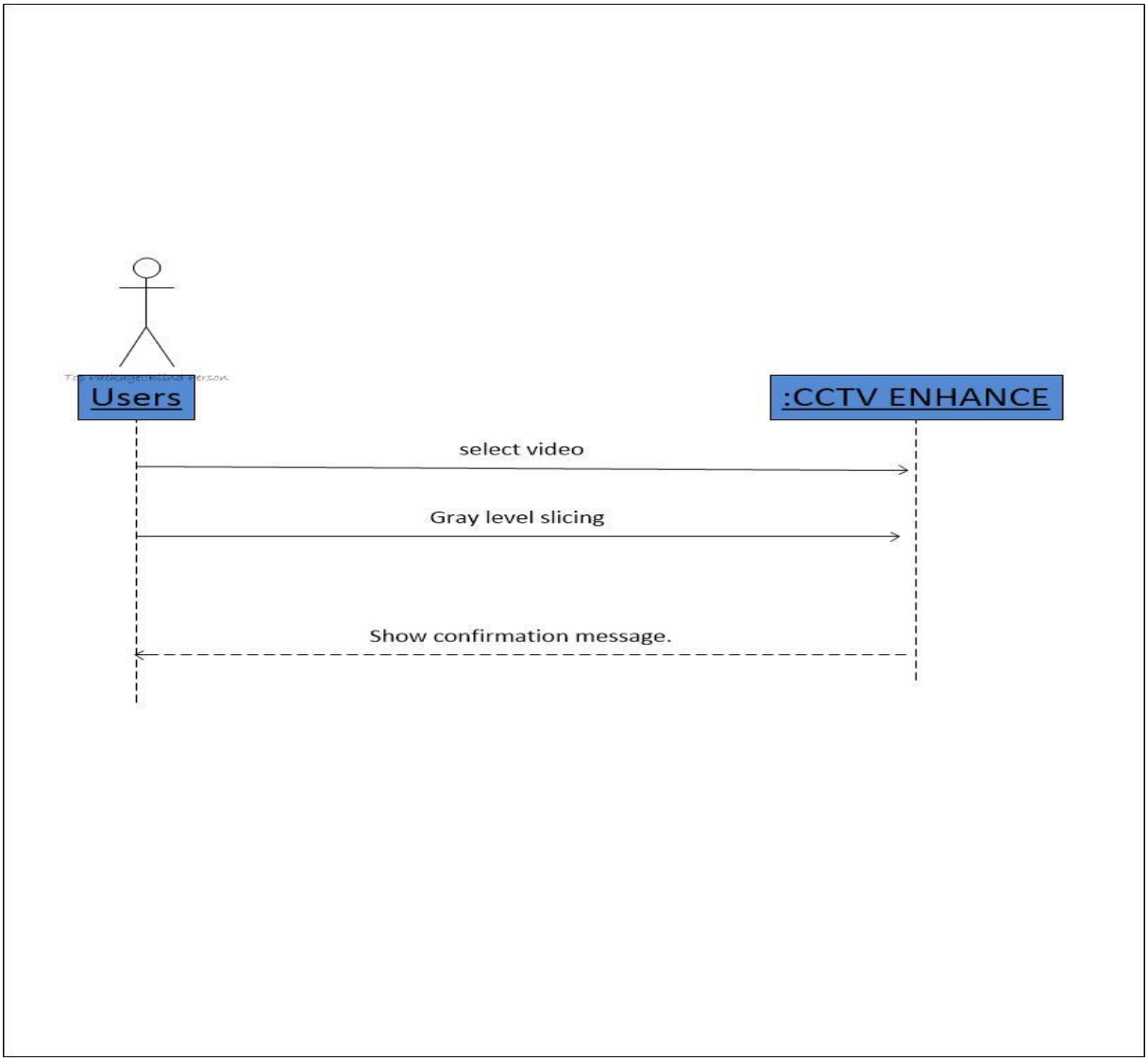

Fig 3.21: SSD for gray level

CHAPTER 4
IMPLEMENTATION

After the design phase, there is implementation phase. This chapter is related to system implementation. The chapter mentions the tools, framework, platforms and database used to develop the application. In this phase we decide how to implement our design and with which technique.

4.1. System Definition

System will mainly be composed of two parts;

- File Storage.
- Desktop Application.

4.1.1. Database Server

System requires of a file storage which is used to store selected video, to store extracted frames and to store videos of different image enhancement techniques.

4.1.2. Desktop Application

System consists of a desktop based Application which will be used as UI (User Interface). User will be able to select and upload video from system using this desktop application, and users will also be able to apply different image enhancement techniques.

4.2 Development Tool

In development tool it includes a software that assists in the creation of new software. Example: Compilers, debuggers etc.

4.2.1 MATLAB

MATLAB (MATrix LABoratory) is a mathematical software that offers integrated development environment (IDE) with a programming language itself (language M).

4.2.1.1 Platform

MATLAB is available for Unix, Windows and MacOS X platforms. Among its high potential, basic functions worthy of mention are matrix manipulation, data and functions representation, implementation of algorithms, creation of user interfaces (GUI) and communication with other programming languages and other hardware devices.

4.2.1.2 Benefits

The software package MATLAB has two additional tools that expand their benefits:

- Simulink (multidomain simulation platform).
- GUIDE (editor user interfaces - GUI).

Additionally, you can extend the capabilities of MATLAB with toolboxes. Toolboxes currently cover most major areas of the world of engineering and simulation, such as digital processing toolbox Images (IPT hereinafter), the signal processing toolbox, the communications toolbox, etc.

4.2.1.3 Interactive

MATLAB provides an interactive work environment whose basic working element are the matrices, allowing the numerical solution of problems in a much shorter time than in traditional programming languages such as Fortran, Basic or C languages, with the advantage that its own programming language is similar to traditional languages. When working with matrices numerous variables can be described mathematically in a highly flexible and efficient way. For

example, an image can be written as a matrix of pixels, a sound like a matrix jitter, and generally can be described with any linear matrix relationship between the components of a mathematical model.

CHAPTER 5
SYSTEM TESTING

This chapter illustrates the test approach that is used in this project, testing tools and environment, and the test cases.

5.1 Test Approach

A test approach is the test strategy that includes the implementation techniques used in a project and defines how testing would be carried out. Test approach has two techniques:

- **Proactive:** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- **Reactive:** An approach in which the testing is not started until design and coding are completed.

The reactive approach will be used in this project because by this approach, we will analyze the field or tool expert's knowledge extremity. We will be able to better analyze various risks linked to the project.

5.2 Test Plan

A document describing the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, the features to be tested, the testing tasks, the test environment, the test design techniques and entry and exit criteria to be used, and the rationale for their choice, and any risks requiring contingency planning. It is a record of the test planning process. It is the basis for formally testing any software/product in a project.

5.2.1 Testing Tools and Environment

The testing environment which I use is black box testing in which the internal structure design implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box inside which one cannot see. This method attempts to find errors in the following categories incorrect or missing functions interface errors.

5.3 Test Cases

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.

The process of developing test cases can also help find problems in the requirements or design of an application.

5.3.1 Select video

This test case will describe the success scenario of Select Video .The user selects the video. The system checks that the selected video exists in the database. If it is exists the user selects successfully.

ID	T001
Description	User select a video.
Tester	User
Setup	Select the file with name “abc.jpg”
Instructions:	1. Enter the file name “abc.jpg” in the search box. 2. Select the file. 3. Press OK Button.
Expected Results	File with name “abc.jpg” should be selected successfully.
Oracle	**Pass**

5.3.2 Select Video (Alternative Scenario)

This test case will describe the alternate scenario of Select Video .The user selects the video. The system checks that the selected video exists in the database. If it does not exists the user cannot select successfully.

ID	T002
Description	User select a video.
Tester	User
Setup	Select the file with name “abc.jpg”
Instructions:	1. Enter the file name “abc.jpg” in the search box. 2. Select the file. 3. Press OK Button.
Expected Results	1. File with name “abc.jpg” should not be selected successfully. 2. There is error in selecting the video.
Oracle	**Fail**

5.3.3 Upload Video

This test case will describe the success scenario of Upload Video .The user uploads the video. The system checks that the uploaded video exists in the database. If it is exists the user upload successfully.

ID	T003
Description	User upload a video.
Tester	User
Setup	Upload the file with name “xyz.jpg”

Instructions:	1. Enter the file name “xyz.jpg” in the search box. 2. Select the file. 3. Press OK Button.
Expected Results	File with name “xyz.jpg” should be uploaded successfully.
Oracle	**Pass**

5.3.4 Upload Video (Alternate Scenario)

This test case will describe the alternate scenario of Upload Video. The user uploads the video. The system checks that the uploaded video exists in the database. If it does not exists the user cannot upload successfully.

ID	T004
Description	User upload a video.
Tester	User
Setup	Select the file with name “xyz.jpg”
Instructions:	1. Enter the file name “xyz.jpg” in the search box. 2. Upload the file. 3. Press OK Button.
Expected Results	1. File with name “xyz.jpg” should not be uploaded successfully. 2. There is error in uploading the video.
Oracle	**Fail**

5.3.5 Extract Frame

This test case will describe the success scenario of Extraction of Frame. The user extracts the required frame. The system checks that the extracted frame exists in the video. If it is exists the user will extract successfully.

ID	T005
Description	User extract the frame(s).
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. Extract the frame(s) from file “xyz.mpg”.
Instructions:	1. Enter the file name “xyz.mpg” in the search box. 2. Select the file. 3. Press OK Button. 4. Enter the time of staring frame and ending frame. (e.g 1:04 – 1:09). 5. Press enhance button for further process.
Expected Results	File with name “xyz.mpg” should be uploaded successfully and the required frames should be successfully extracted.
Oracle	**Pass**

5.3.6 Extract Frame (Alternate Scenario)

This test case will describe the alternate scenario of Extraction of Frame. The user extracts the required frame. The system checks that the extracted frame exists in the video. If it is not existed the user will not extract successfully.

ID	T006
Description	User extract the frame(s).
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. Extract the frame(s) from file “xyz.mpg”.
Instructions:	1. Enter the file name “xyz.mpg” in the search box. 2. Select the file. 3. Press OK Button. 4. Enter the time of staring frame and ending frame. (e.g 1:04 – 1:09). 5. Press enhance button for further process.
Expected Results	1. File with name “xyz.mpg” should not be uploaded successfully 2. The required frames are not successfully extracted.
Oracle	**Fail**

5.3.7 Logarithm Transformation

This test case will describe the success scenario of logarithm transformation. The system applies logarithm transformation on the video successfully.

ID	T007
Description	User apply the logarithm transformation.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement logarithm transformation on “xyz.mpg”.

Instructions:	1. Select the file. 2. Press OK Button. 3. Press logarithm transformation or implementation.
Expected Results	File with name “xyz.mpg” should be uploaded successfully and logarithm transformation should be successfully apply.
Oracle	**Pass**

5.3.8 Logarithm Transformation (Alternate Scenario)

This test case will describe the alternate scenario of logarithm transformation. The system will not implement logarithm transformation successfully.

ID	T008
Description	User apply the logarithm transformation.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement logarithm transformation on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press logarithm transformation button.
Expected Results	File with name “xyz.mpg” will not exist. Logarithm transformation will not successfully apply.
Oracle	**Fail**

5.3.9 Power law

This test case will describe the success scenario of power law. The system will implement power law successfully.

ID	T009
Description	User apply the power law transformation.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement power law transformation on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press power law button to implement power law on video.
Expected Results	File with name “xyz.mpg” should be uploaded successfully and power law should be successfully apply.
Oracle	**Pass**

5.3.10 Power law (Alternate Scenario)

This test case will describe the alternate scenario of power law. The will not implement power law successfully.

ID	T010
Description	Users apply the power law transformation.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement power law transformation on “xyz.mpg”.

Instructions:	1. Select the file. 2. Press OK Button. 3. Press power law implementation on video.
Expected Results	File with name “xyz.mpg” not exits and power law will not be successfully apply.
Oracle	**Fail**

5.3.11 Piecewise Linear Transformation

This test case will describe the success scenario of piecewise linear. The will implement piecewise linear successfully.

ID	T011
Description	User apply the piecewise linear.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement piecewise linear on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press piecewise linear on video.
Expected Results	File with name “xyz.mpg” should be uploaded successfully and piecewise linear should be successfully apply.
Oracle	**Pass**

This test case will describe the alternative scenario of piecewise linear. The will implement piecewise linear successfully.

ID	T012
Description	User apply the piecewise linear.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement piecewise linear on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press piecewise linear on video.
Expected Results	File with name “xyz.mpg” not uploaded successfully and piecewise linear not implement successfully.
Oracle	**Fail**

5.3.13 Histogram Equalization

This test case will describe the success scenario of histogram equalization. The system will implement histogram equalization successfully.

ID	T013
Description	User apply the Equalize histogram.
Tester	User

Setup	1. Upload the video “xyz.mpg”. 2. User implement Equalize histogram on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press Equalize histogram on video.
Expected Results	File with name “xyz.mpg” uploaded successfully and histogram equalization implements successfully.
Oracle	**Pass**

5.3.14 Equalize histogram (alternative scenario)

This test case will describe the alternative scenario of Equalize histogram. The system will not implement histogram equalization successfully.

ID	T014
Description	User apply the Equalize histogram.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement Equalize histogram on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press histogram equalization on video.
Expected Results	File with name “xyz.mpg” not uploaded successfully and Histogram equalization not implemented successfully.
Oracle	**Fail**

5.3.15 Detect motion

This test case will describe the success scenario of motion detection on objects moving in video. The system implements motion detection successfully.

ID	T016
Description	User apply the motion detection.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implements Detect motion on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press Detect motion on video.
Expected Results	File with name “xyz.mpg” uploaded successfully and motion is detected successfully.
Oracle	**Pass**

5.3.16 Detect motion (alternative scenario)

This test case describes the alternative scenario of motion detection of object in video. The system not implement not motion detection successfully.

ID	T016
Description	User apply the motion detection.
Tester	User
Setup	1. Upload the video “xyz.mpg”. 2. User implement Detect motion on “xyz.mpg”.

Instructions:	1. Select the file. 2. Press OK Button. 3. Press Detect motion on video.
Expected Results	File with name "xyz.mpg" not uploaded successfully and motion detection is not implemented.
Oracle	**Fail**

5.3.17 Compare Result

This test case describes the successful scenario of comparing results of different algorithms that automatically enhances video frame by frame. The system implements comparison functionality successfully.

ID	T018
Description	Users apply the compare function.
Tester	User
Setup	1. Upload the video "xyz.mpg". 2. User implement compare result on "xyz.mpg".
Instructions:	1. Select the file. 2. Press OK Button. 3. Press compare result on enhance video.
Expected Results	File with name "xyz.mpg" uploaded successfully and implemented the best algorithm at each frame successfully.
Oracle	**Pass**

5.3.18 Compare Result (alternative scenario)

This test case describes the alternative scenario of comparing result of different algorithms on video. . The system implements comparison functionality unsuccessfully.

ID	T018
Description	User apply the compare result.
Tester	User
Setup	3. Upload the video “xyz.mpg”. 4. User implement compare result on “xyz.mpg”.
Instructions:	1. Select the file. 2. Press OK Button. 3. Press compare result on enhance video.
Expected Results	File with name “xyz.mpg” not uploaded successfully and comparison functionality is not implemented.
Oracle	**Fail**

CHAPTER 6
CONCLUSION AND FUTURE ENHANCEMENT

I have presented an approach for digital image processing for the enhancement of RGB type CCTV footages frame by frame. For this purpose, I have implemented different point processing techniques. To enhance the usability and to give the tool a proper look, Motion detection algorithm is also implemented. After applying the different algorithms, users can save their enhanced videos, hence providing a facility to check their results in future. The tool is capable of automatically implementing the best algorithm, using histogram values at each frame produced by different algorithms. However, the results obtained from it may not be considered as best results since the best results are totally dependent on the requirement of user that what type of enhancement is required by the user, example either noise removal or contrast enhancement, etc. I conclude that for multiple scenarios user have to use different enhancement algorithms. In future, application can be enhanced by;

- By automatic detection of persons in the footage.
- By using more advanced enhancement algorithms.
- By providing the built-in enhancement techniques in the CCTV cameras.

7 References

[1] Margaret Rouse, “CCTV”, [Online]. Available: https://whatis.techtarget.com/definition/CCTV-closed-circuit-television [Accessed 15 11 2017]

[2] “Image Enhancement”, Enychan [Online]. Available: http://www.eie.polyu.edu.hk/~enyhchan/imagee.pdf [Accessed 2 12 2017]

[3] Professor G R Sinha**, “**Design and Implementation of Image Enhancement Techniques”, [Online] Available: https://www.researchgate.net/publication/296675494_Design_and_Implementation_of_Image_Enhancement_Techniques_in_Frequency_Domain [Accessed28 12 2017]

[4] Ms.Seema Rajput, “Comparative Study of Image Enhancement Techniques”, [Online], Available: https://pdfs.semanticscholar.org/541e/8f1f0cc141f1b939f84b941b2f53c516c026.pdf [Accessed 15 12 2017]

[5] “Image Enhancement in the Spatial Image Enhancement in the Spatial Domain”, [Online], Available: http://www.ece.mcmaster.ca/~xwu/lecture3_enhancement_spatial.pdf [Accessed 20 12 2017]

[6] Raman Maini and Himanshu Aggarwal, “A Comprehensive Review of Image Enhancement Techniques”, [Online], Available: https://arxiv.org/ftp/arxiv/papers/1003/1003.4053.pdf [Accessed 15 01 2018]

[7] “Histogram Equalization”, [Online], Available:https://www.math.uci.edu/icamp/courses/math77c/demos/hist_eq.pdf [Accessed 18 02 2018]