

1. Introduction

1.1 Problem definition:

To develop a system to control computer using hand gesture recognition.

1.2 Aim and objective of the project:

- a) To develop an intelligent system this can recognize human gesture by which humans can interact with computer.
- b) To control all operations of mouse such as right click left click and movement of cursor over the desktop by hand gesture recognition.

To control certain functions of a computer like play/pause a video, move left/right in a photo slide show, scroll up/down in a web page and other without use of keyboard, mouse or joystick.

1.3 Scope and limitation of the project:

a) Scope -

Scope is, user can easily use this application, to control all operations of mouse such as right click left click and movement of cursor over the desktop by hand gesture recognition.

b) Limitations –

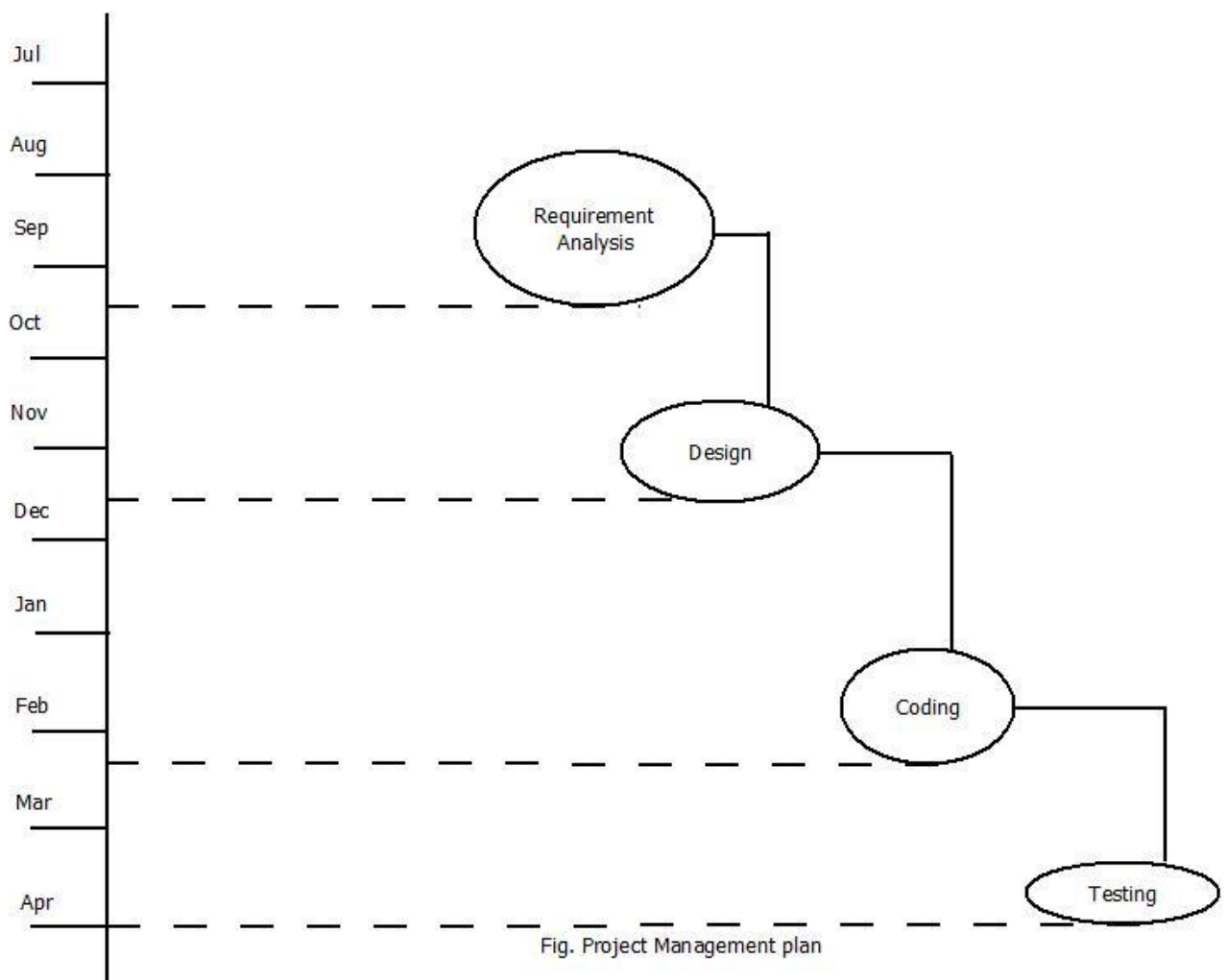
Inability to detect and track hand if the background is very similar to the skin color, to detect and track hand in extreme light.

1.4 Timeline of the project:

We have used classic lifecycle paradigm also called “Water fall model”. We had completed software requirement analysis by the mid of September 2018 which encompasses both system and software requirement gathering. By the end of December 2018 we had completed project planning and design. On the basis of design prepared in the previous stage by the mid of February 2019 we completed coding stage.

After completion of coding stage testing stage carried out in the mid of March 2019. Various criteria of testing were taken into account which includes unit testing, integration testing and validation testing. First, each and every module of the project was tested under the unit testing, after the unit testing; integrated testing was carried out by integrating all modules testing in unit testing. After unit testing the module prepared was crossed checked with the design.

1.5 Project Management Plan:



1.6 Project Cost:

In this project the Cost Estimation based on COCOMO (Constructive Cost Model) formula for this Model is follows:

Effort = Constant * (Size) scale factor * Effort Multiplier

Effort in terms of person-months

Constant: 2.45 in 1998 based on Organic Mode –

Size: Estimated Size in KLOC –

Scale Factor: combined process factors

Effort Multiplier (EM): combined effort factors

The basic COCOMO equations take the form

Effort applied (E) = a^b (KLOC) b^b Man-months]

Development Time (D) = c^b (Effort Applied) d^b [months]

People required (P) = Effort applied / Development Time [count]

Where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project. The coefficients a^b , b^b , c^b and d^b are given in the following table

Software Project	a ^b	b ^b	c ^b	d ^b
Organic	2.4	1.05	2.5	0.38
Semidetached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Organic Mode

$$\text{EFFORT} = 2.4 * (0.4)^{1.05} = 0.443$$

$$\text{Development Time} = 2.5 * (0.443)^{0.38} = 1.835$$

$$\text{People required} = 0.443/1.835 = 0.241$$

2. Background study and literature overview

2.1 Literature overview:

The main goal of gesture recognition research is creation of a system that can identify specific human hand gestures and use them to convey information or for device control as well as applications control. Hand Gesture Recognition System is a branch of Human Computer Interaction in which Human hand gestures are recognized by the computer system and then perform pre-defined task as per the application for controlling software as well as hardware.

Haitham Hasan and Sameem Abdul-Kareem 2013 [1] proposed a Human Computer Interaction for Vision Based Hand Gesture Recognition. With the development of ubiquitous computing, current user interaction approaches with keyboard, mouse and pen are not sufficient. Due to the limitation of these devices the usable command set is also limited. The work here is summarized as detecting and recognizing hand gesture from real time gesture images and with the help of neural network analyzes the efficiency of the network for the hand gesture recognition problem. From the research it is not able that any developed system should be both flexible and expandable in order to maximize efficiency, accuracy and understand ability of a developed system.

Prof. Sachin A Urabinahatti and Prof. Apeksha S Bandekar 2014 [2] proposed a Human-Computer Interaction using Hand Gestures. This paper provides a summary of previous surveys done in this area and focuses on the different application domain which employs hand gestures for efficient interaction. The use of hand gestures as a natural interface serves as a motivating force for research in gesture taxonomies, its representations and recognition techniques. The main goal of this survey is to provide appearance based methods is mostly used for detection of fingertip in hand. Soft computing provides a way to define things which are not certain but with an approximation makes use of learning models and training data. It is effective in getting results where the exact positions of hand or fingers are not possible.

Ram Pratap Sharma and Gyanendra K. Verma 2015 [3] proposed a Human Computer Interaction using Hand Gesture. The entire procedure of tracking gesture to their

representation and converting them to some purposeful command is known as gesture recognition. The present system which we have implemented although seems to be user friendly as compared to modern device or command based system but it is less robust in detection and recognition as we have seen in the previous step. We need to improve our system and try to build more robust algorithm for both recognition and detection even in the cluttered background and a normal lighting condition. We also need to extend the system for some more class of gestures as we have implemented it for only 6 classes. However we can use this system to control applications like power point presentation, media player.

2.2 Critical appraisal of other people's work:

Prof. Sachin A Urabinahatti and Prof. Apeksha S Bandekar 2014 [2] proposed a Human-Computer Interaction using Hand Gestures. The system developed here real time hand gesture recognition best power point slide control. The main problem of gesture recognition lies in complexity of the classification algorithm. The work presented in this project recognizes static gesture only.

Ram Pratap Sharma and Gyanendra K. Verma 2015 [3] proposed a Human Computer Interaction using Hand Gesture. The system which they implemented is less robust in detection and recognition. They need to improve their system and try to be more robust algorithm for both recognition and detection even in cluttered background and a normal lighting condition. And also they need to extend the system for more classes of gestures

2.3 Investigation of current project and related work:

The proposed system is vision based, which uses image processing techniques and inputs from a computer webcam. Vision based gesture recognition tracking and gesture recognition. The input frame would be captured from the webcam and systems are generally broken down into four stages, skin detection, hand contour extraction, hand the skin region would be detected using skin detection. The hand contour would then be found and used for hand

tracking and gesture recognition. Hand tracking would be used to navigate the computer cursor and hand gestures would be used to perform mouse functions such as right click, left click, scroll up and scroll down. The scope of the project would therefore be to design a vision based CC system, which can perform the mouse function previously state

3. Requirement analysis

3.1 Functional Requirements:

- By using hand gesture, system helps to move cursor anywhere on screen.
- Desktop application can open, operate and close through system interaction.
- System performs various functionalities of the gallery can be performed using gestures, these include: next image, previous image.
- System performs functionalities of a media player that includes play, pause, next song, previous song.
- System can use gestures for right and left click to change slides during a presentation.

3.1 System Requirements :

- OpenCV 2.4.10 needs to be installed prior to running. It is an open source computer vision programming functions library aimed at developing applications based on real time computer vision technologies.
- Tools: visual studio, c sharp.
- Windows: administrator permission are needed for some parts of the program to function properly. Supporting Webcam Drivers.

4. System design

4.1 Architectural Design:

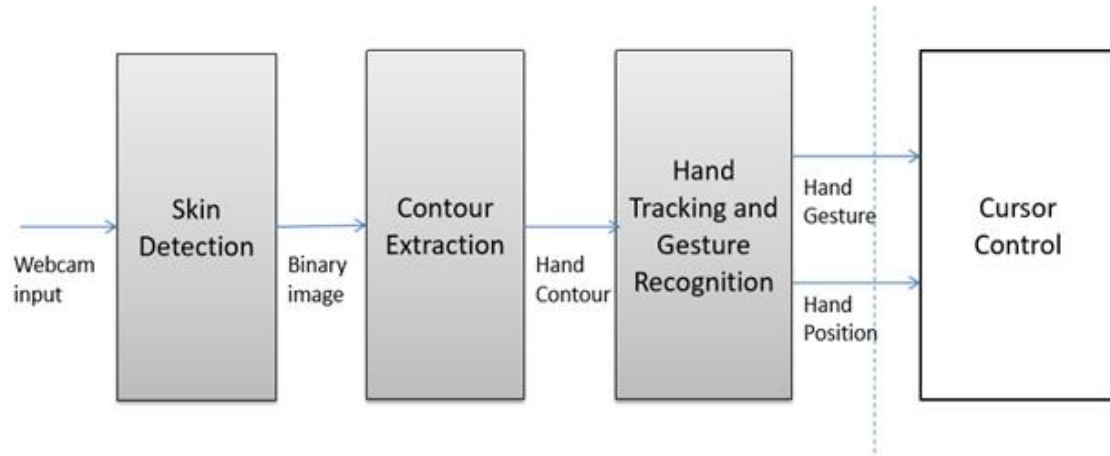
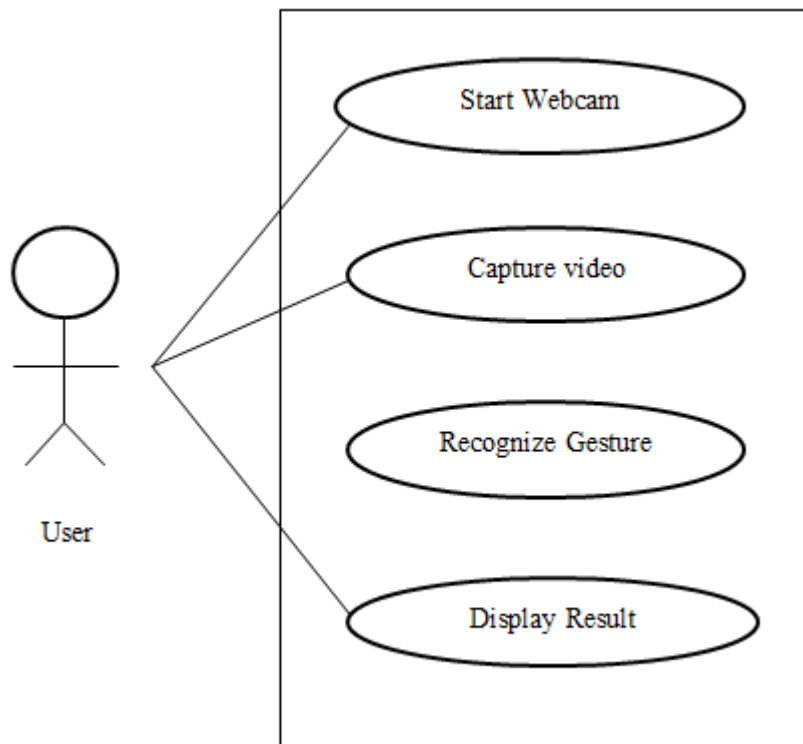


Fig. Architecture Diagram of Proposed System

ACTIVATI

4.2 User Interface:



4.3 Algorithmic description of each modules:

4.3.1 Skin detection:

1. Take input from webcam.
2. Convert the RGB color space image into YCbCr color space.
3. Skin color detection in image comparing standard skin color values.
4. Convert image in to binary image.

4.3.2 Hand Contour Extraction:

1. Take input as a binary image.
2. Perform edge detection.
3. Find all contours of image.
4. Find biggest contour in an image
5. Obtain hand contour.

4.3.3 Hand Tracking:

1. Get hand contour as an input.
2. Find Convexity defects of hand contour.
3. Find convexity hull of hand contour
4. Determine fingertips by analyzing convexity defects
5. Count fingertips.

4.2.4. Cursor Control:

1. Take count of fingertips as a input.
2. Based on count apply events.
3. If count=0 or 1 do mouse move using system mouse_event function.
4. If count ≥ 4 then do mouse click using system mouse_event function.

4.4 System Modelling :

4.4.1 Dataflow Diagram:

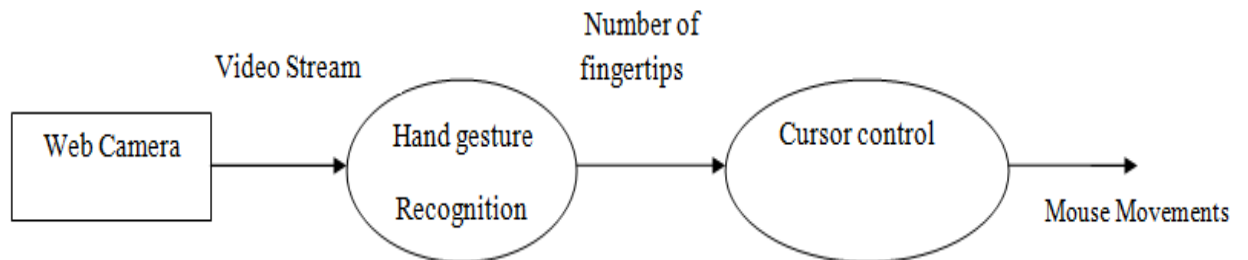


Fig. DFD level – 0

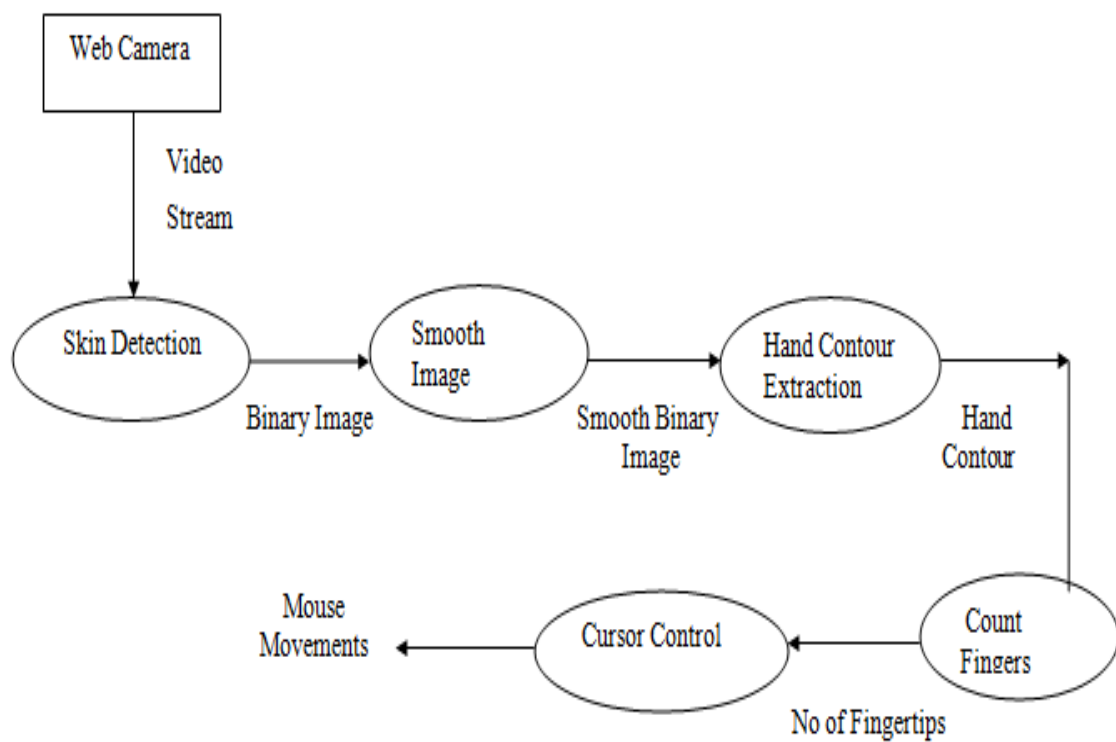
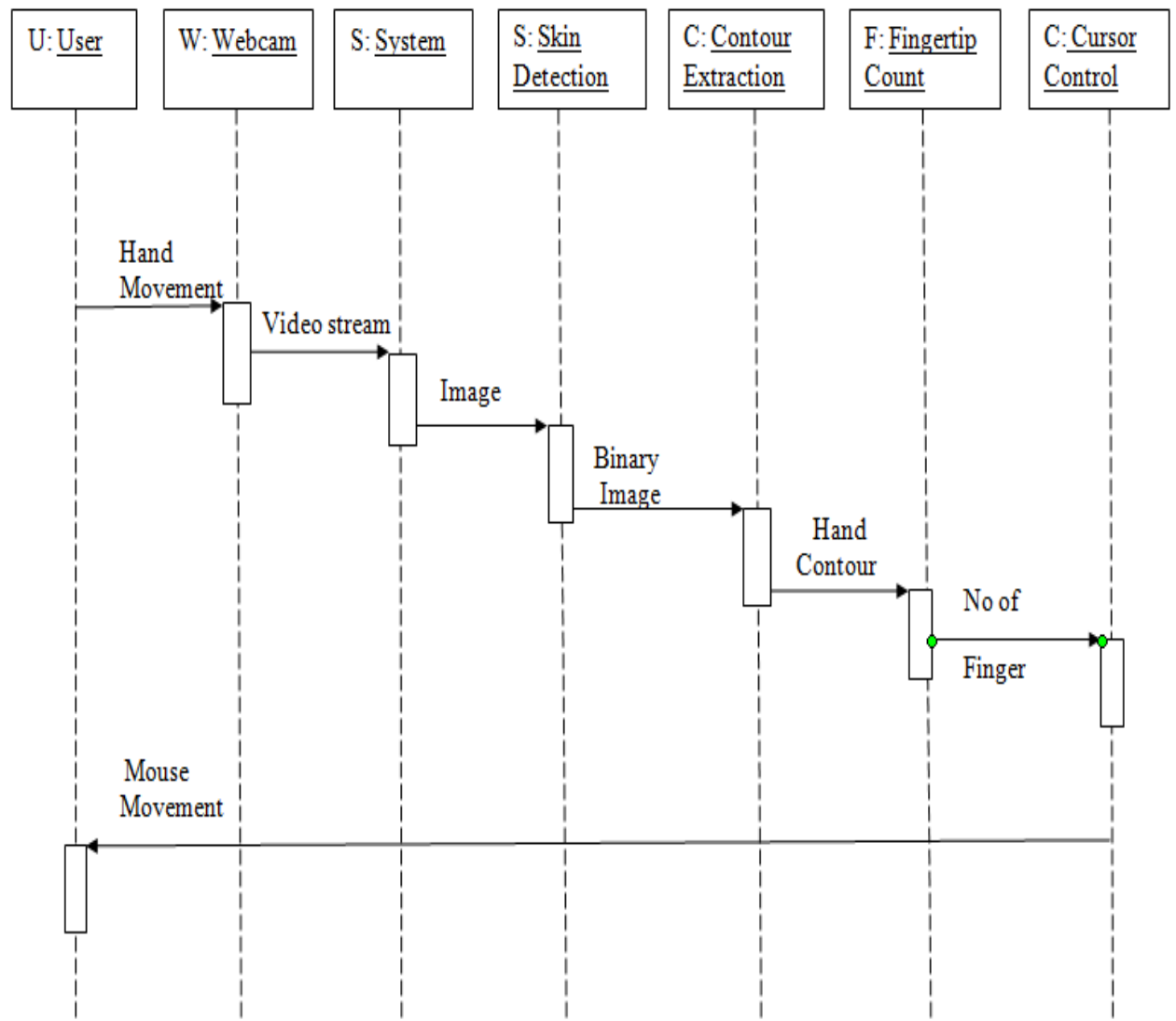


Fig. DFD level - 1

4.4.2 Sequence Diagram:



4.4.3 Activity Diagram:

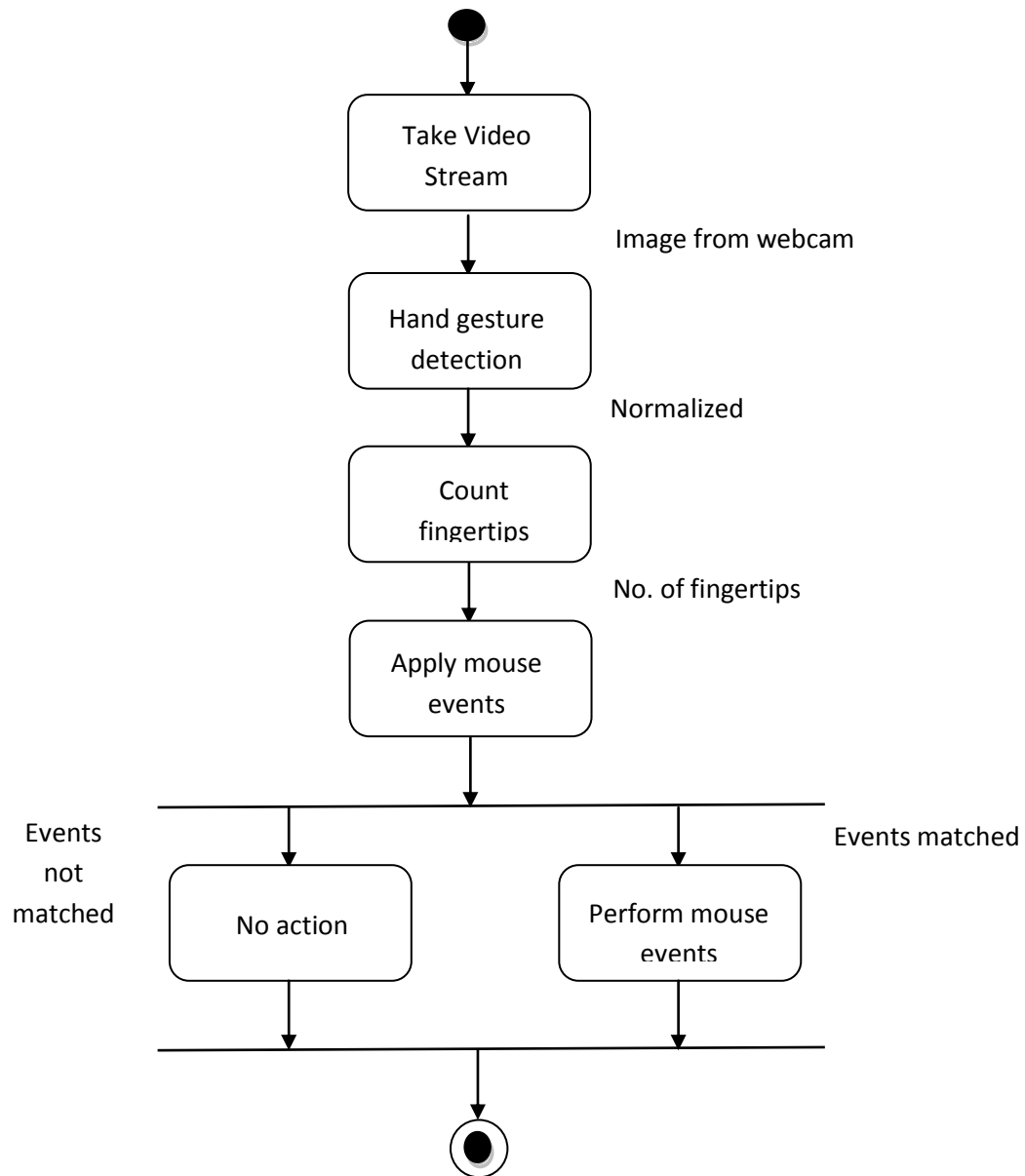


Fig. Activity Diagram

5. Implementation

5.1 Environmental Setting for Running the Project:

Steps to setup environment variable for Opencv

1. Click Start, then Control Panel, then System.
2. Click Advanced, then Environment Variables
3. Add the location of the bin folder of the Opencv installation for the path variable.

Then following is a typical for the opencv path variable:

C:\opencv\build\x64\vc10\bin

Steps to setup environment variable for emgucv

1. Click Start, then Control Panel, then System.
2. Click Advanced, then Environment Variables
3. Add the location of the bin folder of the emgucv installation for the path variable.

Then following is a typical for the emgucv path variable: C:\Emgu\bin

5.2 Detailed Description of Methods:

- Hand gesture recognition

1. Image Acquisition:

- a. It involves capturing and storing digital images taken from video camera.
- a. Image is the input acquired by this module.
- b. The camera used to acquire digital images is then conducted to skin detection unit.

2. Skin Detection :

- a. This module is mainly used for skin detection.
- b. RBG Image acquired from the camera is then converted to YCbCr color space.
- c. Skin is detected on the bases of standard skin color range.

d. YCbCr images are then converted into binary image.

3. Image Smoothing:

- a. After converting image into binary, the image has to smooth.
- b. For smoothing the object boundaries in image structuring elements are used.
- c. This process is done in image smoothing.

4. Hand Contour Extraction:

- a. After smoothing image perform edge detection.
- b. Find all contours in image and in that find hand contour that is biggest contour in an image.

5. Finger tips count:

- a. For finding the fingertips of hand gesture find convexity defects and convexity hull of the image.
- b. Based on defects points and convexity hull count fingertips of hand.

- Cursor Control

1. Mouse move:

- a. Finding the position of the current mouse cursor.
- b. Set the cursor position to new point using current position of cursor and scaling factor.

2. Mouse Click:

- a. Importing the Dll file for mouse events.
- b. Call the mouse_event function of the system set the flags of that function depending on count of fingertips.

5.3 Implementation Details:

Function skin_segmenmt (object, event)

```
{  
    Convert RGB Image into YCbCr color space  
    If skin image pixels in range Y(0-255),Cb(131-185),Cr(80-135)  
        Then convert old skin image with new binary skin image  
}
```

Function Smooth_Image(object,event)

```
{  
    Using structuring elements apply cvErode() and cvDilate() method on skin  
    Image  
    Make skin Image smoot using SmoothGaussian() method  
}
```

Function find_fingers(object, event)

```
{  
  
    Flip skin image horizontally  
  
    extract all contours from image using FindContours  
  
    while (contours is not equal to null)  
    {  
        if contour area > next contour area  
            biggestcontour=contour  
        applying convexity defect allgoritm to find the count of fingers  
    }
```

```
    }  
Function cursor control (object, event)  
{  
    If finger number is equal to 0 or 1  
    Then  
        Move cursor using cursor position  
    If finger number is greater than or equal to 4  
    Then  
        Do mouse click using mouse event function of system  
    }
```

6. Integration and Testing

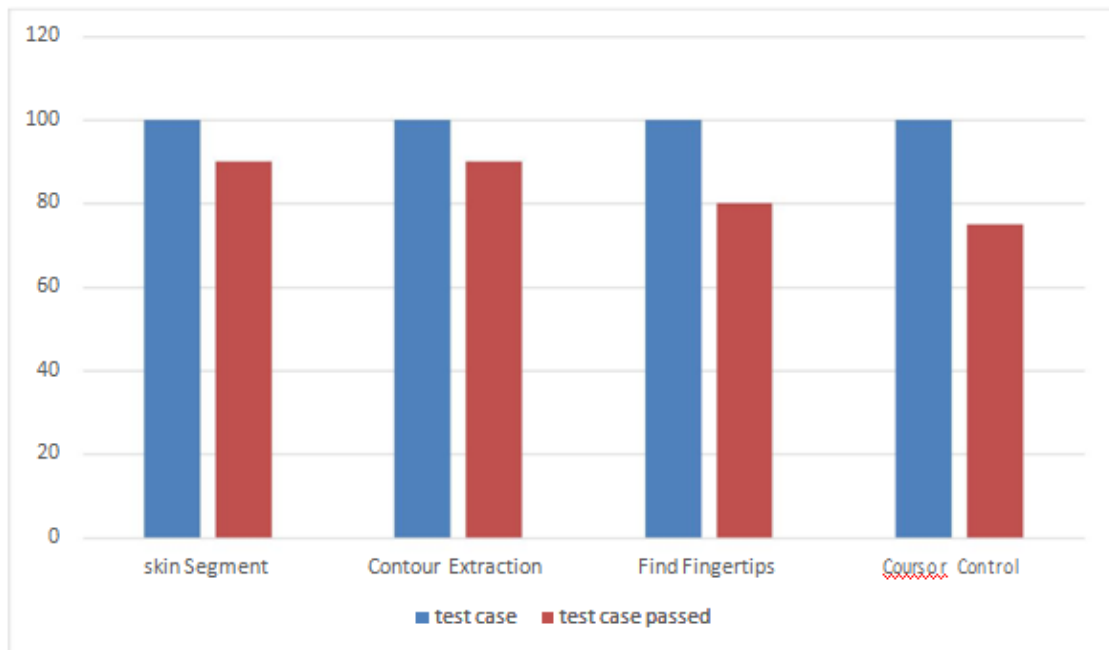
6.1 Integration and Testing

Test Performed	Expected	Actual
Skin Segment	Required to separate skin color and other color in the image.	Yes
Smooth Image	Required to smooth segmented image.	Yes
Find Finger	Required to find the number of fingers in the captured image.	Yes
Move Mouse	Required to move mouse.	Yes
Mouse Click	Required to mouse click.	Yes
Mouse Control Webcam	Required to control system by using gesture recognition	Yes

Fig. Testing

7. Performance Analysis

7.1 Performance Analysis:



8. Applications

8.1 Applications

The concept of gesture recognition has many applications include automated homes, operating visual devices like television solely with gestures and also medical applications Meanwhile, this project deals with opening of various applications like Microsoft office ,windows media player, internet explorer etc.

- Gesture based gaming.
- Human Motion Sensing and Human Psychology Detection.
- Painting in Ms-Paint and Operating On-Screen Keyboard.
- Air writing in seminars or in colleges and Air painting for artists.
- Sign Language Recognition.

9. Installation Guide and User Manual

9.1 Installation Guide and User Manual:

There are following step processes to run this application on PC.

1. Make sure you have Visual Studio C# installed

2. Download openCV 2.4.11 for windows

Now go to the folder where you have the downloaded the executable file and choose to 'Run as Administrator'.

Select: C:\Program Files\ as the path and click extract. It will create a folder called openCV 2.4.11 with the path: C:\Program Files \OpenCV 2.4.11

2.Download EmguCV Installer/Folder:

- a. For Method 1:download 32-bit installer of EmguCV:

libemgucv-windows-x86-2.2.1.1150.exe

- b. For Method 2:Download 64-bit installer of EmguCV

Libemgucv-windows-x64-2.2.1.1150.zip

4. Install EmguCV to windows OS directory.

C:\Emgu\emgucv-windows-x86 2.2.1.1150

C:\Emgu\libemgucv-windows-x64-2.2.1.1150

5. Install service pack to avoid platform dependency issue

Congrats! EmguCV installed successfully.

10.Ethics

10.1Ethics

10.1.1 Ethical Practice for CSE Students:

As Computer Sc. & Engineering Students, I believe it is unethical to:

- Take credit for someone else's work.
- Hire someone to write assignment.
- Make a copy of software for personal or commercial use.
- Download pirated software from internet.
- Distribute pirated software form internet.
- Deliberately provide inaccurate reference for a project or report study.
- Knowingly permit students work done by one student to be submitted by another student.
- Buy software with single user license and then install it on multiple computers.

11. References

11.1 References

1. [1] Haitham Hasan and Sameem Abdul-Kareem , "Human Computer Interaction for Vision Based Hand Gesture Recognition ", IEEE, 2013.

2. [2] Prof. Sachin A Urabinahatti and Prof. Apeksha S Bandekar , "Human-Computer Interaction using Hand Gestures",IJRITCC ,2, May 2014.
3. [3] Ram Pratap Sharma and Gyanendra K. Verma , "Human Computer Interaction using Hand Gesture",IMCIP,2015