1. INTRODUCTION

Within the past few years, as computer technology continues to develop, people want more compact electronic devices. Human Computing Interaction, particularly gesture recognition and object recognition, is becoming increasingly important. In this project, we introduce a method for controlling the mouse events and cursor movements through live webcam. In today's world, most cell phones communicate with the user via touch screen technology. However, this technology is still prohibitively expensive for use on desktops and laptop computers.

The most efficient and expressive way of human communication is through hand gesture, which is a universally accepted language. It is pretty much expressive such that the dumb and deaf people could understand it. In this work, real-time hand gesture system is proposed. Experimental setup of the system uses fixed position low-cost web camera high- definition recording feature mounted on the top of monitor of computer or a fixed camera on a laptop, which captures snapshot using Red Green Blue [RGB] color space from fixed distance.

In this project we use the edge detection which is generally an image processing discipline. It is used to find the edges of the digital images through some mathematical methods.There are generally two types of edges detections are there. They are Search Based Edge Detection and Zero Crossing Based Edge Detection. There are a finite number of edge detection methods are present. One of the most famous one is Canny Edge Detector.

1.1   Motivation

In the present-day scenario, most of the mobile phones are using touch screen technology to interact with the user. But this technology is still not cheap to be used in desktops and laptops. Our objective was to create a virtual mouse system using a Web camera to interact with the computer in a more user-friendly manner that can be an alternative

Approach for the touch screen. This activity on the gesture-based user interface (UI) has been proliferating in the last decade. The main reason for these technologies becomes more popular is because it can be applied to many different fields easily and efficiently.

Human-Computer Interaction today greatly emphasizes on developing more spontaneous and natural interfaces. The Graphical User Interface (GUI) on Personal Computers (PCs) is quite developed, well defined and provides an efficient interface for a user to interact with the computer and access the various applications effortlessly with the help of mice, trackpad, etc.

1.2 Problem Statement

Computer vision-based mouse can easily be applied to the web services, smart home systems, robot manipulation, and games. That is why tracking non-rigid motions from sequential videos have been a great interest to the computer vision community. We grew up interacting with the physical objects around us. How we manipulate these objects in our lives every day, we use gestures not only to interact with objects but to interact with each other and this brings us a step closer to Human-object relationship by using gesture recognition technique.In this research still webcam has been used to recognize the gestures. There is no need for 3D or stereo cameras and above research has also been tested on low-cost 1.3-megapixel laptop webcam.

The work is dedicated to a computer vision-based mouse that acts as an interface between the user and various computing devices in the dynamic environment. This paper presents the technique to perform numerous mouse operations thus obviating the need for hardware used for interaction between the user and the computing device. The same approach can be applied to endless tasks such as browsing images, playing games, changing T.V channels, etc. There is a threshold value for distance (in meters) between the user and camera which can further be varied according to the camera's resolution. It means if the subject who wants to be recognized with his hand gestures in some environment, the subject has to come close to certain fixed distance to the camera. This research was done on 1.3-megapixel webcam with a threshold value of 2m.The main elements of this technology are as follows:

1.3 The Project Summary:

The aim of the project is to provide a virtual mouse that can be used in any system which consisting of a in built or external webcam connected to it. This can be achieved through the movements of the finger or using the hand gestures. This virtual mouse can perform all the operation that can be performed by a physical or an ordinary mouse. The main motive of this project is to give a user friendly, easy to handle and low-cost mouse that can be easily used by anyone. This project will change the human computer interaction without a doubt. In this project the input is taken from the webcam itself. The raw video is the input of the virtual mouse which will be converted into image and there by processing it with the help of computer vision (cv2).

The user wears colored tapes to provide information to the system. Individual frames of the video are separately processed. The processing techniques involve an image subtraction

algorithm to detect colors. Once the colors are detected the system performs various operations to track the cursor and performs control actions, the details of which are provided below.

1.4 PURPOSE:

- ✓ The main objective of this system is to control the movements of mouse cursors and scrolling functions of it through the help of hand gestures.
- ✓ It is also used to overcome problems in the real world such as situations where there is no space to use a physical mouse and also for the persons who have problems in their hands and are not able to control a physical mouse.
- ✓ Virtual mouse that uses the hand gestures and hand tips for performing mouse functions in computer by using Computer Vision.
- ✓ With help of this system, we can track the fingertip of the hand gestures by using in-built camera or web cam and perform the movements of cursor, functions of mouse like left click, right click, double click and scrolling operations.

1.5 SCOPE:

With the advancement in computer vision and artificial intelligence it is now possible for computers to capture and identify human hand gestures thus abridging the gap in human machine interaction. Hand gestures can be effectively used in giving commands to your computer. The computer will capture the gesture, interpret it and take the appropriate action based on its previous inputs and data. This makes our day-to-day tasks easy and enhance the

human computer interaction over the regular text-based commands and GUI**.**

Human computer interaction takes place mainly by means of input output devices such as mouse, keyboard, joystick etc. we type commands from the keyboard or select an option using the mouse. These tasks can be simplified is these traditional input output mechanisms are replaced byhand gestures.

Hand gesture recognition system has wide application in today's world. With the advent in computer vision, we can make our machine perceive knowledge of the surroundings just like humans. Today computers can interpret a gesture just like humans can. This helps in providing ease and naturalness for human computer interaction.

2.LITERATURE SURVEY

Gesture recognition gives the best interaction between human and machine. Gesture recognition is also important for developing alternative human computer interaction modalities. It enables human to interface with machine in a more natural way. Gesture recognition can be used for many applications like sign language recognition for deaf and dumb people, robot control etc. This technology has wide applications in the fields of augmented reality, computer graphics, computer gaming, prosthetics, and biomedical instrumentation.

Digital Canvas is an extension of this system which is gaining popularity among artists,by which the artist could create 2D or 3D images using the Virtual Mouse technology using the hand as brush and a Virtual Reality kit or a monitor as display set. This technology can be used tohelp patients who don't have control of their limbs. In case of computer graphics and gaming this technology has been applied in modern gaming consoles to create interactive games where a person's motions are tracked and interpreted as commands.

Researchers around the world are now focused on to make our devices more interactive and trying to make the devices operational with minimal physical contact. In this research, we propose an interactive computer system which can operate without any physical keyboard and mouse. This system can be beneficial to everyone, especially to the paralyzed people who face difficulties to operate physical keyboard and mouse. We used computer vision so that user can type on virtual keyboard using a yellow-colored cap on his fingertip, and can also navigate to mouse controlling system. Once the user is in mouse controlling mode, user can perform all the mouse operations only by showing different number of fingers. We validated both module of our system by a 52 years old paralyzed person and achieved around 80% accuracy on average.

There are some related works carried out on virtual mouse using hand gesture detectionby wearing a glove in the hand and also using color tips in the hands for gesture recognition, but they are no more accurate in mouse functions. The recognition is not so accurate because of wearing gloves; also, the gloves are also not suited for some users, and in some cases, the recognition is not so accurate because of the failure of detection of color tips. Some efforts have been made for camera-based detection of the hand gesture interface.

Banerjee.A, A. Ghosh, K. Bharadwaj, H. Saikia introduced a new method using the web camas the source to control the movements of the mouse using color detection. In their work, have

tried to control mouse cursor movement and click events using a camera based on color detection technique. Here real time video has been captured using a Web Camera. The user wears colored tapes to provide information to the system. Individual frames of the video are separately processed. The processing techniques involve an image subtraction algorithm to detect colors. Once the colors are detected the system performs various operations to track the cursor and performs control actions, the details of which are provided below.

No additional hardware is required by the system other than the standard webcam which is provided in every laptop computer.

Following are the steps in our approach:

➢ Capturing real time video using Web-Camera.

➢ Processing the individual image frame.

➢ Flipping of each image frame.

➢ Conversion of each frame to a grey scale image.

➢ Color detection and extraction of the different colors (RGB) from flipped gray scale image

➢ Conversion of the detected image into a binary image.

➢ Finding the region of the image and calculating its centroid.

➢ Tracking the mouse pointer using the coordinates obtained from the centroid.

➢ Simulating the left click and the right click events of the mouse by assigning different color pointers.

For the system to work they need a sensor to detect the hand movements of the user. The webcam of the computer is used as a sensor. The webcam captures the real time video at a fixed frame rate and resolution which is determined by the hardware of the camera. The frame rate and resolution can be changed in the system if required.

✓ Computer Webcam is used to capture the Real Time Video

✓ Video is divided into Image frames based on the FPS (Frames per second) of the camera

✓ Processing of individual Frames

When the camera captures an image, it is inverted. This means that if we move the color

pointer towards the left, the image of the pointer moves towards the right and vice-versa. It's similar to an image obtained when we stand in front of a mirror (Left is detected as right and right is detected as left). To avoid this problem we need to vertically flip the image. The image captured is an RGB image and flipping actions cannot be directly performed on it. So the individual color channels of the image are separated and then they are flipped individually. Afterflipping the red, blue and green colored channels individually, they are concatenated and a flippedRGB image is obtained.

As compared to a colored image, computational complexity is reduced in a gray scale image. Thus the flipped image is converted into a gray scale image. All the necessary operations were performed after converting the image into gray scale.

The most important step in the whole process is color detection. The red, green and blue color object is detected by subtracting the flipped color suppressed channel from the flipped Gray-Scale Image. This creates an image which contains the detected object as a patch of grey surrounded by black space.

The grey region of the image obtained after subtraction needs to be converted to a binary image for finding the region of the detected object. A grayscale image consists of a matrix containing the values of each pixel. The pixel values lay between the ranges 0 to 255 where 0 represents pure black and 255 represents pure white color. They use a threshold value of 20% to convert the image to a binary image. This means that all the pixel values lying below 20% of the maximum pixel value is converted to pure black that is 0 and the rest is converted to white that is 1. Thus the resultant image obtained is a monochromatic image consisting of only black and white colors. The conversion to binary is required because MATLAB can only find the properties ofa monochromatic image.

For the user to control the mouse pointer it is necessary to determine a point whose coordinates can be sent to the cursor. With these coordinates, the system can control the cursor movement. An inbuilt function in MATLAB is used to find the centroid of the detected region. The output of function is a matrix consisting of the X (horizontal) and Y (vertical) coordinates of the centroid. These coordinates change with time as the object moves across the screen.

- Centroid of the image is detected.

- Its co-ordinates are located and stored in a variable.

3.PROPOSED METHODOLOGY

3.1 EXISTING SYSTEM

Though the wireless or Bluetooth mouse technology is invented still, that tec hnology is not completely device free. A Bluetooth mouse has the requirement of battery power and connecting dongle. Presence of extra devices in a mouse increases the difficulty to use it.

For the system to work we need a sensor to detect the hand movements of the user. The webcam of the computer is used as a sensor. The webcam captures the real-time video at a fixed frame rate and resolution which is determined by the hardware of the camera. The frame rate and resolution can be changed in the system if required.

- Computer Webcam is used to capture the Real-Time Video.

- Video is divided into Image frames based on the FPS (Frames per second) of the camera.

- Processing of individual Frames.

When the camera captures an image, it is inverted. This means that if we move the color pointer towards the left, the image of the pointer moves towards the right and vice-versa. It's similar to an image obtained when we stand in front of a mirror (Left is detected as right and right is detected as left). To avoid this problem we need to vertically flip the image. The image captured is an RGB image and flipping actions cannot be directly performed on it. So the individual color channels of the image are separated and then they are flipped individually. Afterflipping the red, blue and green colored channels individually, they are concatenated and a flipped RGB image is obtained.

The given below figure is an example of already existing virtual mouse which uses three different colored tapes which are wrapped around the index, thumb and middle fingers. The unique operations of the mouse are performed with the different types of combinations of these colors. These three colors generally be red, blue and green. These RGB colors are used in order to reduce the processing time of the image which is taken from the live webcam. They are Search Based Edge Detection and Zero Crossing Based Edge Detection. There are a finite number of edge detection methods are present.

*Fig 3.1 Virtual mouse using the colored tapes*

3.1.1 Disadvantages of Existing System:

- The virtual mouse which are already present are quite inaccurate i.e., they need a more than required amount of light to process the movements of the hand or the finger.

- In some other systems even though the accuracy of the mouse movements is high, the usage of unwanted properties like colored tapes or clothes are wrapped around the finger or sensors are used in order to detect the movements of fingers through color detection.

3.2 PROPOSED SYSTEM

- The system is to present an approach for Human-computer Interaction (HCI), where we have tried to control the mouse cursor movement and click events of the mouse using hand gestures.

- Hand gestures were acquired using a web cam.

- This method mainly focuses on the use of a Web Camera to develop a virtual human-computer interact interaction device in a cost-effective manner.

- The Hand movements are recognized with the help of media pipe which offers a cross platform customizable Machine Learning solutions for live and streaming media.

- These media pipe is trained with more than 30 thousands of pictures of hands of different people.

- The movements of the hands are tracked using the 21 points in the hand.
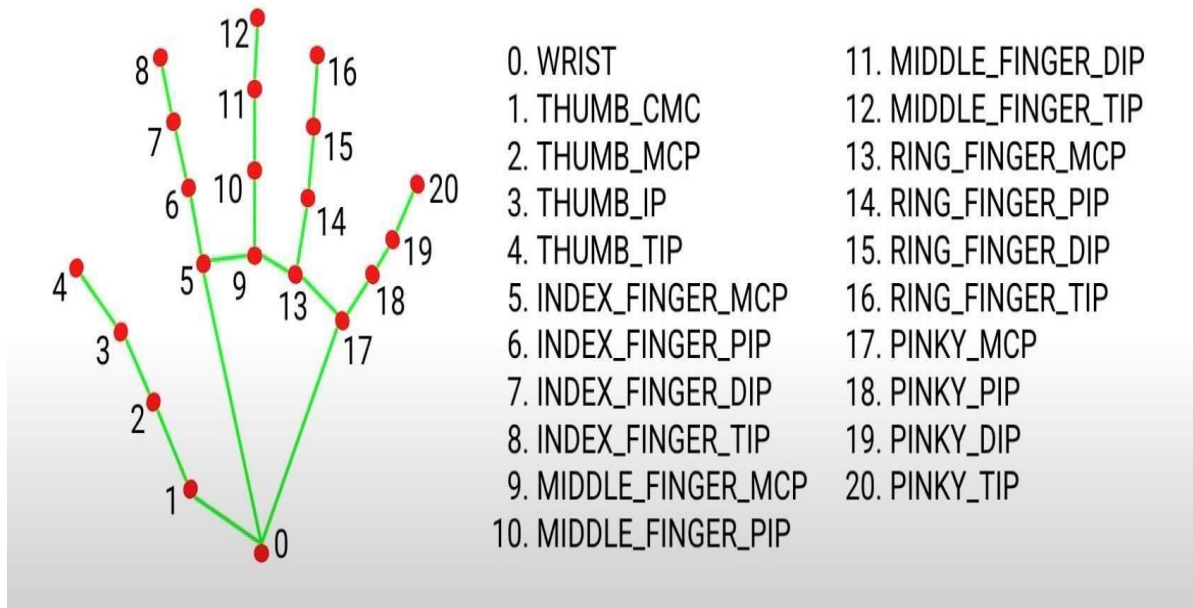
Fig 3.2 All 21 points of a single hand.

3.2.1 Advantages of proposed system:

- The main advantage of using hand gestures is to interact with computer as a non-contact human computer input modality.

- Reduce hardware cost by eliminating use of mouse.

- Convenient for users not comfortable with touchpad.

- The framework may be useful for controlling different types of games and other applications dependent on the controlled through user defined gestures.

- Less prone to physical damage and mechanical wear and tear.

- Avoidance of mouse related wrist damage.

3.3 PROJECT ALGORITHM

Edge Detection, is an Image Processing discipline that incorporates mathematics methods to find edges in a Digital Image. Edge Detection internally works by running a filter/Kernel over a Digital Image, which detects discontinuities in Image regions like stark changes in brightness/Intensity value of pixels. There are generally two types of edges detections are there. They are Search Based Edge Detection and Zero Crossing Based Edge Detection. There are

finite number of edge detection methods are present. One of the most famous one is Canny Edge Detector.

3.3.1 Algorithm

Procedure: Click_Sparm Gaussian_filter (noise x)

 Detect the remaining edges after filtering

 For i=1➔k do

If $f_x(x)\%N==0$ then

Return

end if

end for

Employ a sobel algorithm

For i=1➔k do

Parallel loop(click for)

If threshold p>0 then

If threshold p>0 then

P⬅$G_{(x,y)}$

Θ⬅$\arctan(G_{(x)} \div G_{(y)})$

End if

End for

Perform hysteresis

End procedure

In Canny edge Detection we have following steps for the image processing, those are

- Grayscale conversion

- Gaussian blur

- Determine the intensity gradients

- Non-maximum suppression

- Double thresholding

- Edge tracking by hysteresis
- Cleaning up

3.4 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

3.4.1   ECONOMICAL FEASIBILITY

3.4.2   TECHNICAL FEASIBILITY

3.4.3   SOCIAL FEASIBILITY

3.4.1 ECONOMICAL FEASIBILITY

The purpose of an economic feasibility study (EFS) is to demonstrate the net benefit of a proposed project for accepting or disbursing electronic funds/benefits, taking into consideration the benefits and costs to the agency, other state agencies, and the general public as a whole. This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

3.4.2 TECHNICAL FEASIBILITY

A technical feasibility study assesses the details of how you intend to deliver a product or service to customers. Think materials, labor, transportation, where your business will be located, and the technology that will be necessary to bring all this together.

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources.This will lead to high demands being placed on the client. The developed system must havea modest

requirement, as only minimal or null changes are required for implementing this system. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client.

3.4.3 SOCIAL FEASIBILITY:

Social feasibility is one of the feasibility study where the acceptance of the people is considered regarding the product to be launched. It describes the effect on users from the introduction of the new system considering whether there will be a need for retraining the workforce.

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.3 PRELIMINARY INVESTIGATION:

Preliminary investigations include the process of gathering, preparing, preserving and analyzing evidence in order to assess whether a crime occurred. The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, i.e., preliminary investigation begins.The activity has three parts:

- Request Clarification

- Feasibility Study

- Request Approval

3.5.1 REQUEST CLARIFICATION:

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

FEASIBILITY STUDY:

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- Operational Feasibility
- Economic Feasibility
- Technical Feasibility

Operational Feasibility:

Operational feasibility study is the measure of how well a proposed system solves the problems ,and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility:

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer-based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at any time.The Virtual Private Network is to be developed using the existing resources of the organization. So ,the project is economically feasible.

Technical Feasibility:

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

4.SYSTEM DESIGN

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

4.1 SYSTEM ARCHITECTURE

The architecture of a system describes its major components, their relationships (structures), and how they interact with each other. Software architecture and design includes several contributory factors such as Business strategy, quality attributes, human dynamics, design, and IT environment.
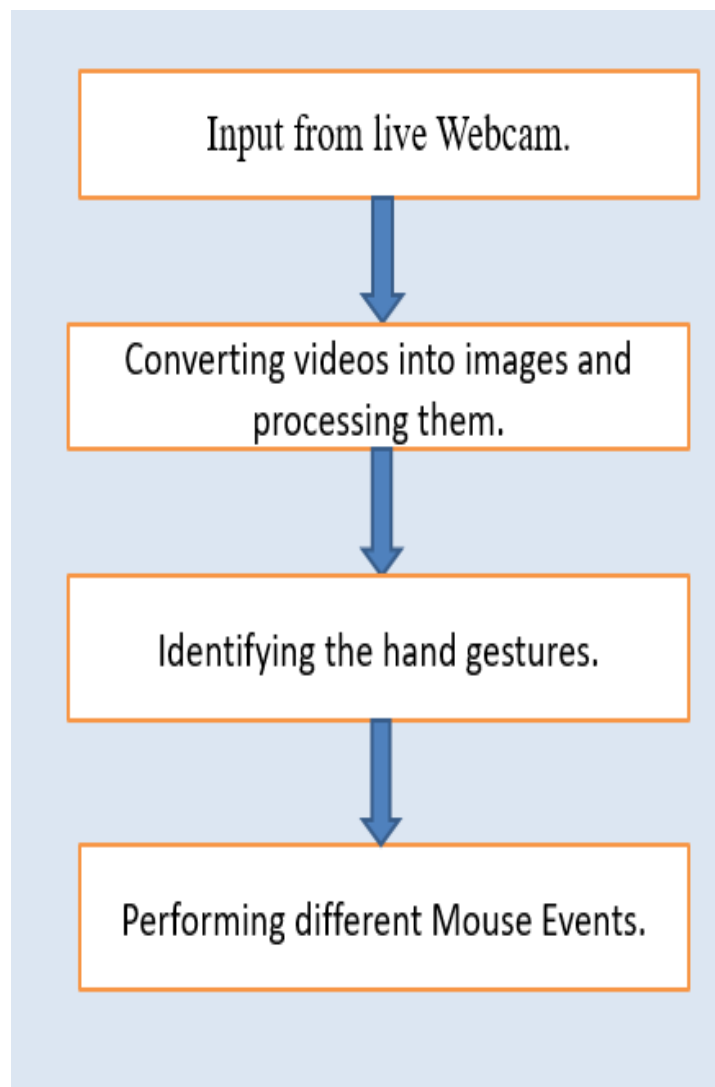
Fig 4.1 System Architecture

4.2 UML DIAGRAMS:

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the software system and its components. It is a graphical language, which provides a vocabulary and set of semantics and rules. The UML focuses on the conceptual and physical representation of the system. It captures the decisions and understandings about systems that must be constructed. It is used to understand, design, configure, maintain, and control information about the systems.

UML represents Unified Modeling Language. UML is an institutionalized universally useful showing dialect in the subject of article situated programming designing. The fashionable is overseen, and become made by way of, the Object Management Group. The goal is for UML to become a regular dialect for making fashions of item arranged PC programming. In its gift frame UML is contained two noteworthy components: a Meta-show and documentation. Later on, a few types of method or system can also likewise be brought to; or related with, UML.

The Unified Modeling Language is a popular dialect for indicating, Visualization, Constructing and archiving the curios of programming framework, and for business demonstrating and different non-programming frameworks. The UML speaks to an accumulation of first-rate building practices which have verified fruitful in the showing of full-size and complicated frameworks.

The UML is a essential piece of creating gadgets located programming and the product development method. The UML makes use of commonly graphical documentations to specific the plan of programming ventures.

4.2.1 Goals

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.

2. Provide extendibility and specialization mechanisms to extend the core concepts.

3. Be independent of particular programming languages and development process.

4. Provide a formal basis for understanding the modeling language.

5. Encourage the growth of OO tools market.

6. Support higher level development concepts such as collaborations, frameworks, patterns

and components.

7. Integrate best practices.

The UML is a language for:

- ✓ Visualizing

- ✓ Specifying

- ✓ Constructing

- ✓ Documenting

Visualizing

Through UML we see or visualize an existing system and ultimately, we visualize how the system is going to be after implementation. Unless we think, we cannot implement. UML helps to visualize, how the components of the system communicate and interact with each other .Specifying

Specifying means building, models that are precise, unambiguous and complete UML addresses the specification of all the important analysis design, implementation decisions that must be made in developing and deploying a software system.

Constructing

UML models can be directly connected to a variety of programming language through mapping a model from UML to a programming language like JAVA or C++ or VB. Forward Engineering and Reverse Engineering is possible through UML.

Documenting

The Deliverable of a project apart from coding are some Artifacts, which are critical in controlling, measuring and communicating about a system during its developing requirements, architecture, desire, source code, project plans, tests, prototypes release etc.

4.3 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented

use cases), and any dependencies between those use cases. A use case diagram shows a set of use cases and actors and their relationships.

Use case diagrams address the static use case view of a system. These diagrams are especially important in organizing and modeling the behaviors of a system. A use case in software engineering and systems engineering is a description of a system's behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario-driven threads through the functional requirements.

Use cases describe the system from the user's point of view. Use cases describe the interaction between one or more actors (an actor that is the initiator of the interaction may be referred to as the 'primary actor') and the system itself, represented as a sequence of simple steps. According to Bittner and Spence, "Use cases, stated simply, allow description of sequences of events that, taken together, lead to a system doing something useful." Each use casedescribes how the actor will interact with the system to achieve a specific goal. One or morescenarios may be generated from a use case, corresponding to the detail of each possible way of Achieving that goal.

Activity diagrams are mainly used as a flow chart consists of activities performed by the system. But activity diagram is not exactly a flow chart as they have some additional capabilities. These additional capabilities include branching, parallel flow, swim lane etc.

Use cases typically avoid technical jargon, preferring instead the language of the end user or domain expert. Use cases are often co-authored by systems analysts and end users. The UML use case diagram can be used to graphically represent an overview of the use cases for a given system and a use-case analysis can be used to develop the diagram. Use cases are not normalized by any consortium, unlike the UML use case diagram by OMG.

Within systems engineering, use cases are used at a higher level than within software engineering, often representing missions or stakeholder goals. The Deliverable of a project apart from coding are some Artifacts, which are critical in controlling, measuring and communicating about a system during its developing requirements, architecture, desire, source code, project plans, tests, prototypes release etc. The UML is a essential piece of creating gadgets located programming and the product development method. The UML makes use of commonly graphical

documentations to specific the plan of programming ventures. This allows the specification of simple runtime scenarios in a graphical manner. Use case diagrams address the static use case view of a system.
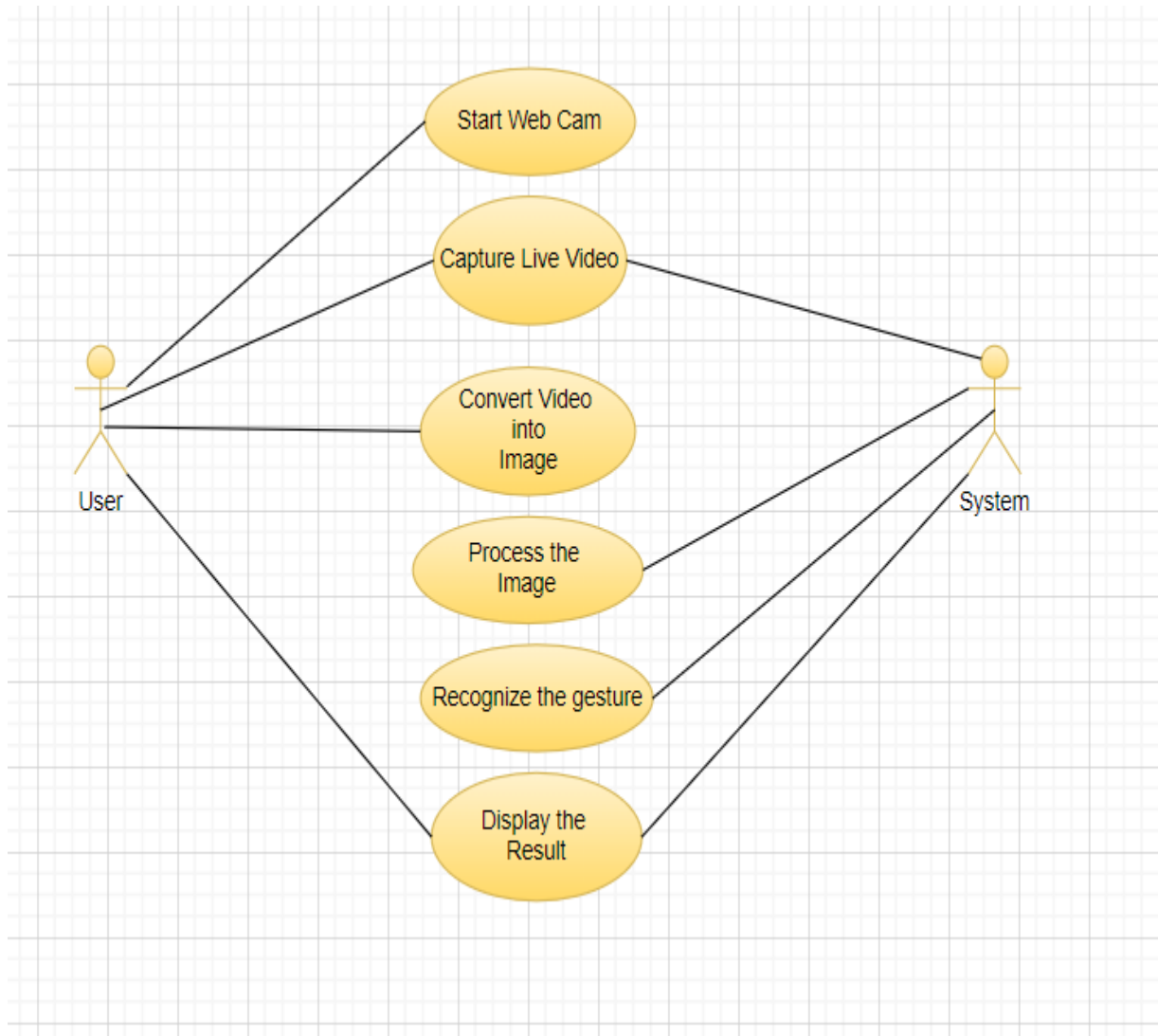
Fig 4.2 Use Case Diagram

4.4 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram thatshows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

A sequence diagram simply depicts interaction between objects in a sequential order i.e.

the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram.

Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner. Use case diagrams address the static use case view of a system. These diagrams are especially important in organizing and modeling the behaviors of a system. A use case in software engineering and systems engineering is a description of a system's behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario-driven threads through the functional requirements. Use case diagrams address the static use case view of a system. These diagrams are especially important in organizing and modeling the behaviors of a system. A use case in software engineering and systems engineering is a description of a system's behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario-driven threads through the functional requirements.

STEPS TO CREATE SEQUENCE DIAGRAM

1. Set the context for the interaction, whether it is a system, subsystem, operation or class.

2. Set the stage for the interaction by identifying which objects play a role in interaction.

3. Set the lifetime for each object.

4. Start with the message that initiates the interaction.

5. Visualize the nesting of messages or the points in time during actual computation.

6. Specify time and space constraints, adorn each message with timing mark and attach suitable time or space constraints.

7. Specify the flow of control more formally, attach pre and post conditions to each message.

A system sequence diagram should specify and show the following:

- External actors

- Messages (methods) invoked by these actors

- Return value (if any) associated with previous messages

- Indication of any loops or iteration area
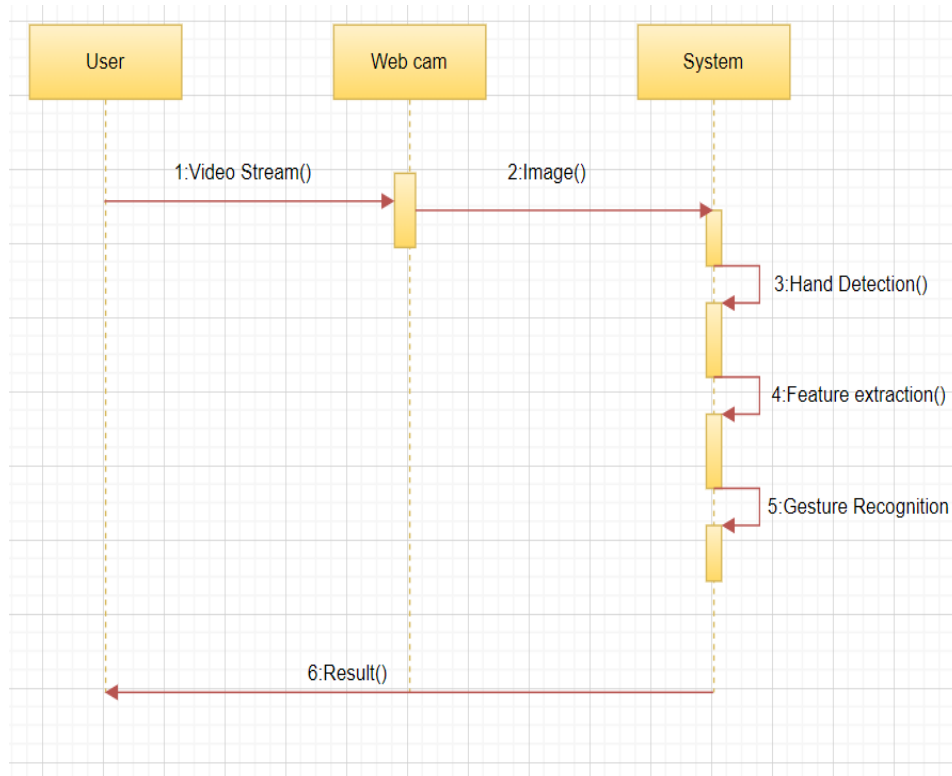
Fig 4.3 Sequence Diagram

4.5 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by- step workflows of components in a system. Use case diagrams address the static use case view of a system. These diagrams are especially important in organizing and modeling the behaviors of a system. A use case in software engineering and systems engineering is a description of a system's behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario-driven threads through the functional requirements.

Activity diagrams are mainly used as a flow chart consists of activities performed by the system .But activity diagram is not exactly a flow chart as they have some additional capabilities. These additional capabilities include branching, parallel flow, swim lane etc.

The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are associated with constraints and conditions.
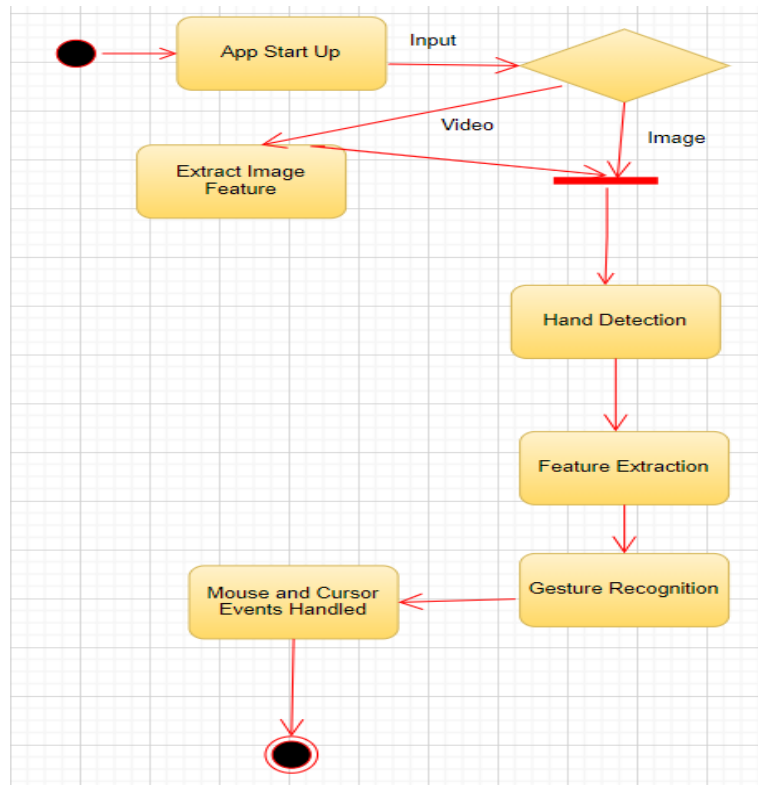
Fig 4.4 Activity Diagram

4.6 DEPLOYMENT DIAGRAM:

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware's used to deploy the application.

How to Draw a Deployment Diagram:

Follow the simple steps below to draw a deployment diagram. You can either use the deployment diagram examples below to get a head start or use our uml diagram tool to start from the beginning.

Step 1: Identify the purpose of your deployment diagram. And to do so, you need to identify then odes and devices within the system you'll be visualizing with the diagram.

Step 2: Figure out the relationships between the nodes and devices. Once you know how they are connected, proceed to add the communication associations to the diagram.

Step 3: Identify what other elements like components, active objects you need to add tocomplete the diagram.

Step 4: Add dependencies between components and objects as required.
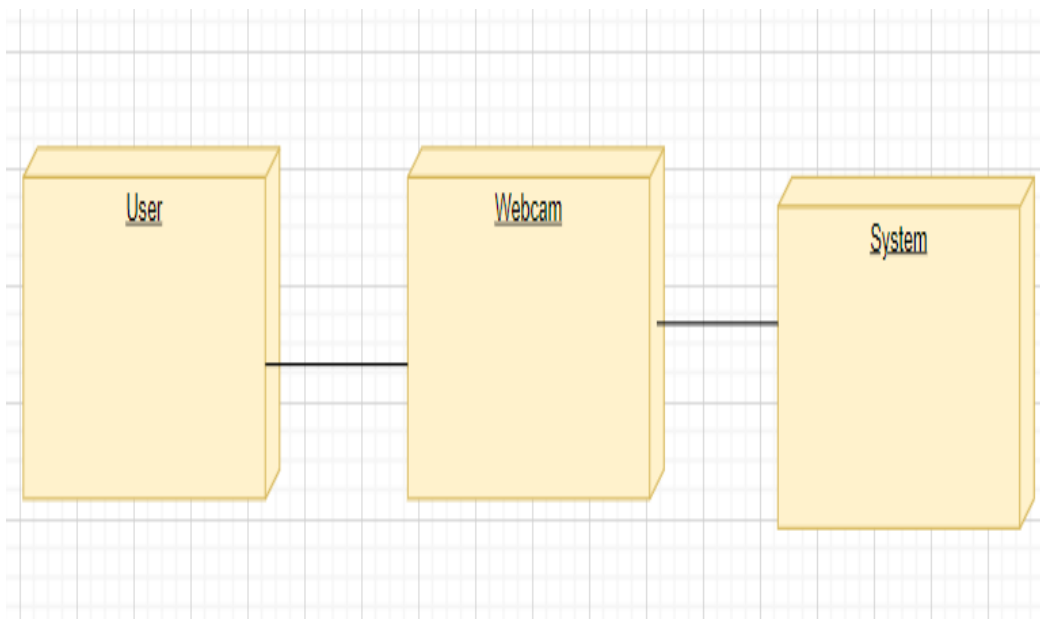
Fig 4.5 Deployment Diagram

4.7 COMPONENT DIAGRAM:

Component diagrams are used to describe the physical artifacts of a system. This artifact includes files, executables, libraries etc. So, the purpose of this diagram is different, Component diagrams are used during the implementation phase of an application. But it is prepared well in advance to visualize the implementation details. Initially the system is designed using different UML diagrams and then when the artifacts are ready component diagrams are used to get an idea of the implementation.

Purpose of Component Diagrams:

The purpose of the component diagram can be summarized as:

• Visualize the components of a system.

- Construct executable by using forward and reverse engineering.

- Describe the organization and relationships of the components.
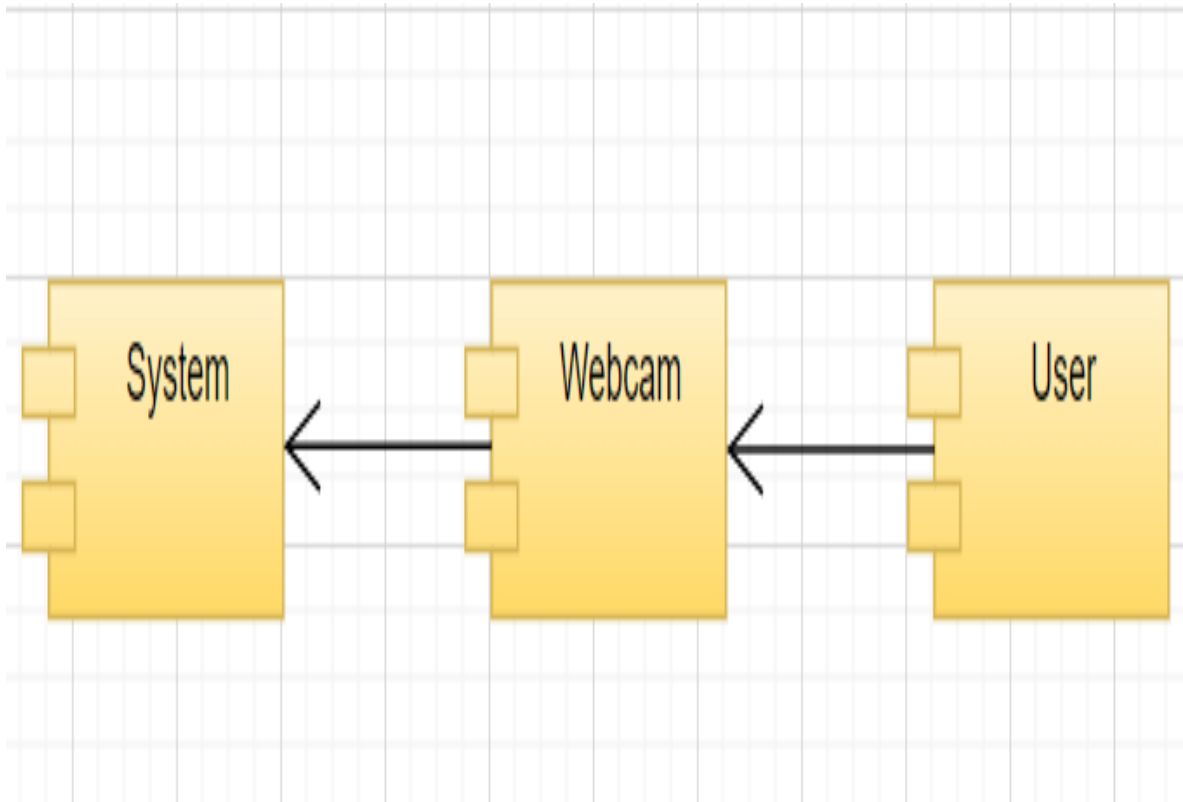
Fig 4.6 Component Diagram

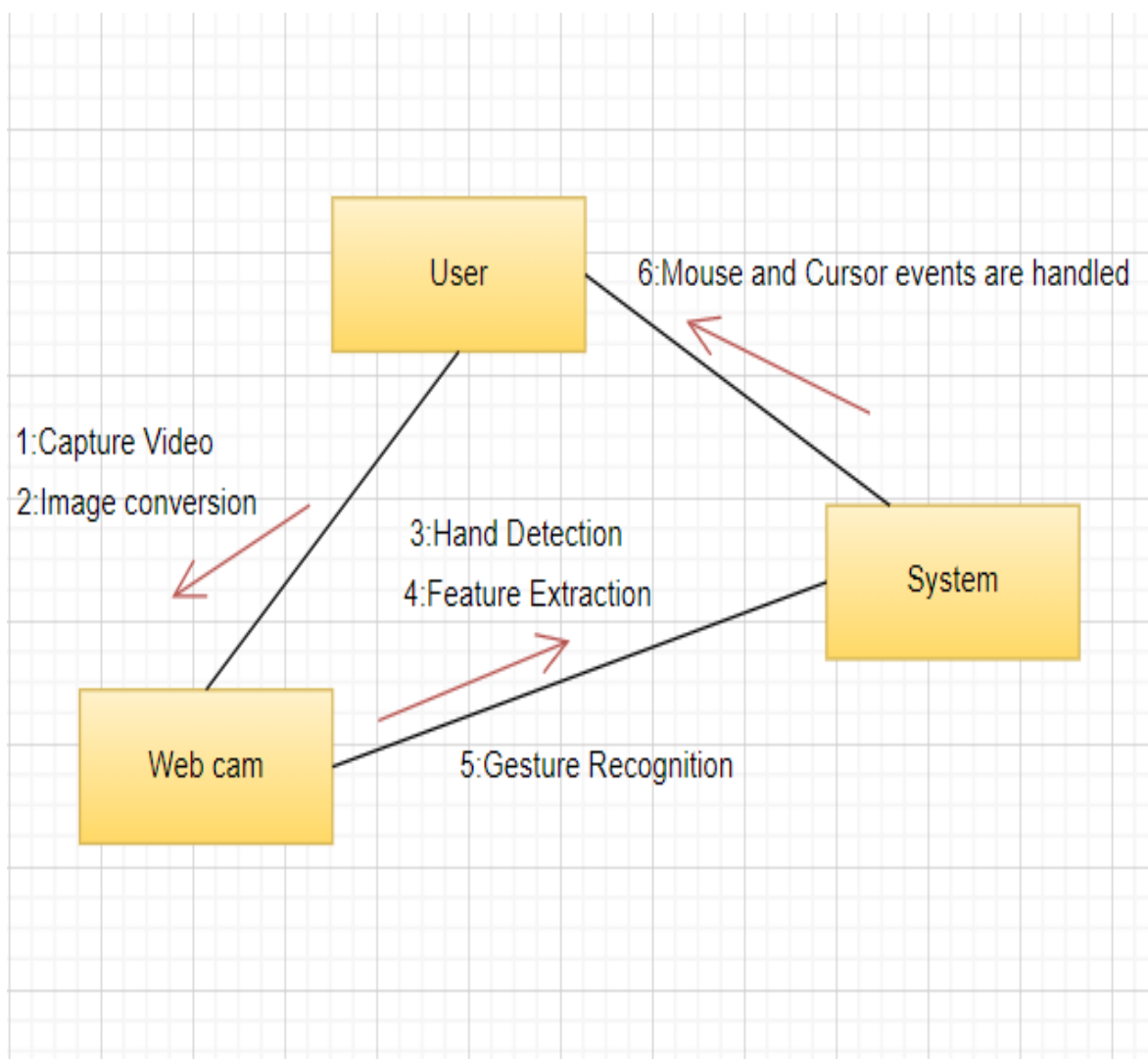## 4.8 COLLABORATION DIAGRAM:

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.

Notations of a collaboration diagram:

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time.

The four major components of a collaboration diagram are:

1. **Objects**- Objects are shown as rectangles with naming labels inside. The naming label follows the convention of object name: class name. If an object has a property or state that specifically influences the collaboration, this should also be noted.

2. **Actors**- Actors are instances that invoke the interaction in the diagram. Each actor has a name and a role, with one actor initiating the entire use case.

3. **Links**- Links connect objects with actors and are depicted using a solid line between two elements. Each link is an instance where messages can be sent.

4. **Messages**- Messages between objects are shown as a labeled arrow placed near a link. These messages are communications between objects that convey information about the activity and can include the sequence number.

Fig 4.7 Collaboration Diagram

5.TOOLS REQUIRED

5.1 INTRODUCTION TO PYTHON:

5.1.1 Python

What is a Script?

Up to this point, I have concentrated on the interactive programming capability of Python. This is a very useful capability that allows you to type in a program and to have it executed immediately in an interactive mode.

Scripts are reusable

Basically, a script is a text file containing the statements that comprise a Python program. Once you have created the script, you can execute it over and over without having to retype it each time.

Scripts are editable

Perhaps, more importantly, you can make different versions of the script by modifying the statements from one file to the next using a text editor. Then you can execute each of the individual versions. In this way, it is easy to create different programs with a minimum amount of typing.

You will need a text editor

Just about any text editor will suffice for creating Python script files. You can use Microsoft Notepad, Microsoft WordPad, Microsoft Word, or just about any word processor if you want to.

5.1.2 Difference between a script and a program:Script:

Scripts are distinct from the core code of the application, which is usually written in a different language, and are often created or at least modified by the end-user. Scripts are often interpreted from source code or byte code, where as the applications they control are traditionally compiled to native machine code.

Program:

The program has an executable form that the computer can use directly to execute the instructions. The same program in its human-readable source code form, from which executable programs are derived (e.g., compiled).

what is Python?

Chances you are asking yourself this. You may have found this book because you want to learn to program but don't know anything about programming languages. Or you may have heard of programming languages like C, C++, C#, or Java and want to know what Python is andhow it compares to "big name" languages. Hopefully I can explain it for you.

5.1.3 Python concepts:

If you're not interested in the how's and whys of Python, feel free to skip to the next chapter. In this chapter I will try to explain to the reader why I think Python is one of the best languages available and why it's a great one to start programming with.

- Open-source general-purpose language.

- Object Oriented, Procedural, Functional.

- Easy to interface with C/Obj C/Java/Fortran.

- Easy-is h to interface with C++ (via SWIG).

- Great interactive environment.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

5.2 HISTORY OF PYTHON:

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol68, Small Talk, and Unix shell and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

5.3 PYTHON FEATURES:

Python's features include –

- ❖ **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.

- ❖ **Easy-to-read** − Python code is more clearly defined and visible to the eyes.

- ❖ **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.

- ❖ **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

- ❖ **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

- ❖ **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- ❖ **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.

- ❖ **Databases** − Python provides interfaces to all major commercial databases.

- ❖ **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

- ❖ **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.

- It can be used as a scripting language or can be compiled to byte-code for building large applications.

- It provides very high-level dynamic data types and supports dynamic type checking.

- IT supports automatic garbage collection.

- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

5.4 DYNAMIC VS STATIC:

Types Python is a dynamic-typed language. Many other languages are static typed, such as C/C++ and Java. A static typed language requires the programmer to explicitly tell the computer what type of "thing" each data value is. For example, in C if you had a variable that was to contain the price of something, you would have to declare the variable as a "float" type.

This tells the compiler that the only data that can be used for that variable must be a floating point number, i.e. a number with a decimal point.If any other data value was assigned to that variable, the compiler would give an error when trying to compile the program.

Python, however, doesn't require this. You simply give your variables names and assign values to them. The interpreter takes care of keeping track of what kinds of objects your program is using. This also means that you can change the size of the values as you develop the program.Say you have another decimal number you need in your program.

With a static typed language, you have to decide the memory size the variable can take when you first initialize that variable. A double is a floating-point value that can handle a much larger number than a normal float (the actual memory sizes depend on the operating environment).If you declare a variable to be a float but later on assign a value that is too big to it, your program will fail; you will have to go back and change that variable to be a double. With Python ,it doesn't matter. You simply give it whatever number you want and Python will take care of manipulating it as needed. It even works for derived values.

For example, say you are dividing two numbers. One is a floating-point number and one is an integer. Python realizes that it's more accurate to keep track of decimals so it automatically calculates the result as a floating-point number. In this chapter I will try to explain to the reader

why I think Python is one of the best languages available and why it's a great one to start programming with.

5.5 VARIABLES:

Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory. Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals or characters in these variables.

5.5.1 Standard Data Types:

The data stored in memory can be of many types. For example, a person's age is stored as a numeric value and his or her address is stored as alphanumeric characters. Python has various standard data types that are used to define the operations possible on them and the storage method for each of them.

Python has five standard data types –

- Numbers

- String

- List

- Tuple

- Dictionary

5.5.2 Python Numbers:

Number data types store numeric values. Number objects are created when you assign a value to them.

5.5.3 Python Strings:

Strings in Python are identified as a contiguous set of characters represented in the quotation marks. Python allows for either pairs of single or double quotes. Subsets of strings can be taken using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.

5.5.4 Python Lists:

Lists are the most versatile of Python's compound data types. A list contains items

separated by commas and enclosed within square brackets ([]). To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different datatype. The values stored in a list can be accessed using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the list and working their way to end -1. The plus (+) sign is the list concatenation operator, and the asterisk (*) is the repetition operator.

5.5.5 Python Tuples:

A tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses. The main differences between lists and tuples are: Lists are enclosed in brackets ([ ]) and their elements and size can be changed, while tuples are enclosed in parentheses (( )) and cannot be updated. Tuples can be thought of as read-only lists.

5.5.6 Python Dictionary:

Python's dictionaries are kind of hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs. A dictionary key can be almost any Python type, but are usually numbers or strings. Values, on the other hand, can be any 109 arbitrary Python object. Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).

5.6 DIFFERENT MODES IN PYTHON:

Python has two basic modes: normal and interactive. The normal mode is the mode where the scripted and finished .py files are run in the Python interpreter. Interactive mode is a command line shell which gives immediate feedback for each statement, while running previously fed statements in active memory. As new lines are fed into the interpreter, the fed program is evaluated both in part and in whole

Some Python Libraries:

1. Open CV
2. Math
3. NumPy
4. TensorFlow
5. PyAuto GUI
6. Pillow
7. Time
8. Win32api
9. Media pipe
10. Win32com

5.6.1 Open CV:

OpenCV (Open-Source Computer Vision Library) is an open-source library that includes several hundreds of computer vision algorithms. The so-called OpenCV 2 API, which is essentially a C++ API, as opposed to the C-based OpenCV 1.x API (C API is deprecated and not tested with "C" compiler since OpenCV 2.4 releases).

OpenCV has a modular structure, which means that the package includes several shared or static libraries. The following modules are available:

- **Core functionality** - a compact module defining basic data structures, including the dense multi-dimensional array Mat and basic functions used by all other modules.

- **Image Processing**- an image processing module that includes linear and non-linear image filtering, geometrical image transformations (resize, affine and perspective warping, generic table-based remapping), colour space conversion, histograms, and so on.

- **Video Analysis** - a video analysis module that includes motion estimation, background subtraction, and object tracking algorithms.

- **Camera Calibration and 3D Reconstruction**- basic multiple-view geometry algorithms, single and stereo camera calibration, object pose estimation, stereo correspondence algorithms, and elements of 3D reconstruction.

- **2D Feature Framework**- salient feature detectors, descriptors, and descriptor matchers.

- **Object Detection**- detection of objects and instances of the predefined classes (fo rexample, faces, eyes, mugs, people, cars, and so on).

- **High-level-GUI**- an easy-to-use interface to simple UI capabilities.

- **Video I/O**- an easy-to-use interface to video capturing and video codecs.

5.6.2 NumPy:

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more. Some of the functionalities of the NumPy is mentioned below:

- Fast and versatile, the NumPy vectorization, indexing, and broadcasting concepts are the

de-facto standards of array computing today.

- NumPy offers comprehensive mathematical functions, random number generators ,linear algebra routines, Fourier transforms, and more.

- NumPy supports a wide range of hardware and computing platforms, and plays well with distributed, GPU, and sparse array libraries.

- The core of NumPy is well-optimized C code. Enjoy the flexibility of Python with the speed of compiled code.

- NumPy's high level syntax makes it accessible and productive for programmers from any background or experience level.

- Distributed under a liberal BSD license, NumPy is developed and maintained publicly on GitHub by a vibrant, responsive, and diverse community.

5.6.3 TensorFlow:

TensorFlow has APIs available in several languages both for constructing and executing a TensorFlow graph. The Python API is at present the most complete and the easiest to use, but other language APIs may be easier to integrate into projects and may offer some performance advantages in graph execution.

• This library features eager execution, which allows you to create, manipulate machine learning models, and make the debugging way easier.

• With TensorFlow, you can easily move your ML models in clouds, on any device and on-premises in any browser.

• TensorFlow comes with an easy to learn architecture. You can easily develop your concept into code and make your publications even easier.

• It has a solution to all of your common machine learning issues. You can easily implement it and go for giving your best.

5.6.4 PyAuto GUI:

PyAuto GUI lets your Python scripts control the mouse and keyboard to automate interactions with other applications. The API is designed to be simple. Py Auto GUI works on Windows, macOS, and Linux, and runs on Python 2 and 3.

PyAuto GUI has several features:

- Moving the mouse and clicking in the windows of other applications.

- Sending keystrokes to applications (for example, to fill out forms).
- Take screenshots, and given an image (for example, of a button or checkbox),and find it on the screen.
- Locate an application's window, and move, resize, maximize, minimize, orclose it (Windows-only, currently).
- Display alert and message boxes.

5.6.5 Win32API:

PyWin32 is a library of Python extensions for Windows that enables you to use the features ofthe Win32 application programming interface (API) on Python.

5.6.6 Media Pipe:

Media Pipe is a Framework for building machine learning pipelines for processing time-series data like video, audio, etc. This cross-platform Framework works in Desktop/Server, Android, iOS, and embedded devices like Raspberry Pi and Jetson Nano.

Media Pipe supports multimodal graphs. To speed up the processing, different calculators run in separate threads. For performance optimization, many built-in calculators come with options for GPU acceleration. otherwise, the system will break. The graph ensures this so that flow is handled correctly according to the timestamps of packets. The Framework handles synchronization, context sharing, and inter-operations with CPU calculators. Media Pipe Toolkit comprises the **Framework** and the **Solutions.**
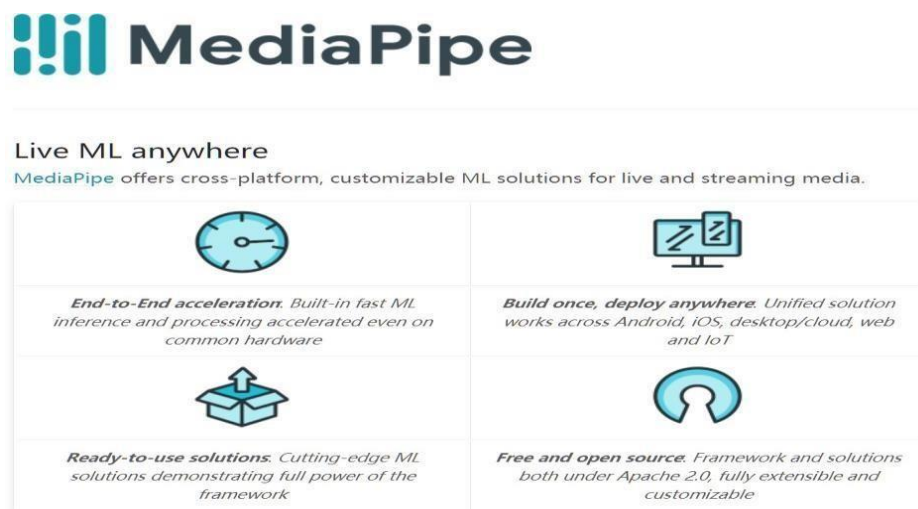
Fig 5.1 Media Pipe

5.6.7 Win32com:

The COM support can be thought of as existing in 2 main portions - the C++ support code (the core Python COM module), and helper code, implemented in Python. The total package is known as "win32com". The win32com support is stand-alone. It does not require Python win.

To facilitate an orderly framework, the Python "ni" module has been used, and the entire package is known as "win32com". As is normal for such packages, win32com itself does not provide any functionality.

Some of the modules are described below:

- **win32com.pythoncom - core C++ support.**

    This module is rarely used directly by programmers - instead the other "helper" modules are used, which themselves draw on the core python com services.

- **win32com.client package**

    Support for COM clients used by Python. Some of the modules in this package allow for dynamic usage of COM clients, a module for generating .py files for certain COM servers, etc.

- **win32com.server package**

    Support for COM servers written in Python. The modules in this package provide most of the underlying framework for magically turning Python classes into COM servers, exposing the correct public methods, registering your server in the registry, etc.

- **win32com.axscript**

    ActiveX Scripting implementation for Python.

- **win32com.axdebug**

    Active Debugging implementation for Python.

- **win32com.mapi**

    Utilities for working with MAPI and the Microsoft Exchange Server.

5.6.8 Time:

There is a popular time module available in Python which provides functions for working with times and for converting between representations. Here, it is used to calculate the frame rate of the hands that has been captured by the live web camera.

5.6.9 Pillow:

In today's digital world, we come across lots of digital images. In case, we are working with Python programming language, it provides lot of image processing libraries to add image processing capabilities to digital images.

Some of the most common image processing libraries are: OpenCV, Python Imaging Library (PIL), Scikit-image, Pillow. However, in this tutorial, we are only focusing on **Pillow module** and will try to explore various capabilities of this module. Pillow is built on top of PIL (Python Image Library). PIL is one of the important modules for image processing in Python. However, the PIL module is not supported since 2011 and doesn't support python 3.

Pillow module gives more functionalities, runs on all major operating system and support for python 3. It supports wide variety of images such as "jpeg", "png", "bmp", "gif", "ppm", "tiff". You can do almost anything on digital images using pillow module. Apart from basic image processing functionality, including point operations, filtering images using built-in convolution kernels, and color space conversions.

- **Image Archives**

    - ✓ The Python Imaging Library is best suited for image archival and batch processing applications. Python pillow package can be used for creating thumbnails, converting from one format to another and print images, etc.

- **Image Display**
    - ✓ You can display images using Tk Photo Image, Bit map Image and Windows DIB interface, which can be used with Python Win and other Windows-based toolkitsand many other Graphical User Interface (GUI) toolkits.
    - ✓ For debugging purposes, there is a show () method to save the image to disk which calls the external display utility.

- **Image Processing**
    - ✓ The Pillow library contains all the basic image processing functionality. You can do image resizing, rotation and transformation.

    - ✓ Pillow module allows you to pull some statistics data out of image using histogram method, which later can be used for statistical analysis and automatic contrast enhancement.

5.6.10 Math:

Python math module is defined as the most famous mathematical functions, which includes trigonometric functions, representation functions, logarithmic functions, etc. Furthermore, it also defines two mathematical constants, i.e., Pie and Euler number, etc.

These functions cannot be used with complex numbers; use the functions of the same name from the math module if you require support for complex numbers. The distinction between functions which support complex numbers and those which don't is made since most users do not want to learn quite as much mathematics as required to understand complex numbers.

Receiving an exception instead of a complex result allows earlier detection of the unexpected complex number used as a parameter, so that the programmer can determine how and why it was generated in the first place.

5.7 TESTING CODE:

As indicated above, code is usually developed in a file using an editor. To test the code, import it into a Python session and try to run it. Usually there is an error, so you go back to the file, make a correction, and test again. This process is repeated until you are satisfied that the codeworks. The entire process is known as the development cycle.

There are two types of errors that you will encounter. Syntax errors occur when the form of some command is invalid. This happens when you make typing errors such as misspellings, or call something by the wrong name, and for many other reasons. Python will always give an error message for a syntax error.

5.8 PYTHON NAMESPACE:

Generally speaking, a namespace (sometimes also called a context) is a naming system for making names unique to avoid ambiguity. Everybody knows a name spacing system from daily life, i.e. the naming of people in first name and family name (surname). An example is a network: each network device (workstation, server, printer, ...) needs a unique name and address. Yet another example is the directory structure of file systems. The same file name can be used in different directories, the files can be uniquely accessed via the pathnames. Many programming languages use namespaces or contexts for identifiers.

An identifier defined in a namespace is associated with that namespace. This way, the same identifier can be independently defined in multiple namespaces. (Like the same file names in different directories) Programming languages, which support namespaces, may have different rules that determine to which namespace an identifier belongs. Namespaces in Python are implemented as Python dictionaries, this means it is a mapping from names (keys) toobjects (values). The user doesn't have to know this to write a Python program and when using namespaces.

5.9 GARBAGE COLLECTION:

Garbage Collector exposes the underlying memory management mechanism of Python, the automatic garbage collector. The module includes functions for controlling how thecollector operates and to examine the objects known to the system, either pending collection orstuck in reference cycles and unable to be freed.

5.10 PYTHON XML PARSER:

XML is a portable, open-source language that allows programmers to develop applications that can be read by other applications, regardless of operating system and/or developmental language.

What is XML?

The Extensible Markup Language XML is a markup language much like HTML or SGML. This is recommended by the World Wide Web Consortium and available as an open standard.XML is extremely useful for keeping track of small to medium amounts of data without requiring a SQL-based backbone. XML Parser Architectures and APIs The Python standard library provides a minimal but useful set of interfaces to work with XML.

The two most basic and broadly used APIs to XML data are the SAX and DOM interfaces. Simple **API for XML SAX :** Here, you register call-backs for events of interest and then let the parser proceed through the document. This is useful when your documents are large or you have memory limitations, it parses the file as it reads it from disk and the entire file is never stored in memory.

Document Object Model DOM API: This is a World Wide Web Consortium recommendation wherein the entire file is read into memory and stored in a hierarchical tree − based form to represent all the features of an XML document. SAX obviously cannot process information as fast as DOM can when working with large files.

On the other hand, using DOM exclusively can really kill your resources, especially if used on a lot of small files. SAX is read-only, while DOM allows changes to the XML file. Since these two different APIs literally complement each other, there is no reason why you cannot use them both for large projects.

5.11 SOFTWARE INSTALLATION FOR MACHINE LEARNING PROJECTS

1. Visit the website https://www.jetbrains.com.com/pycharm/download and click the "DOWNLOAD" link under the Community section.

Download PyCharm

Windows   macOS   Linux

Professional

Full-featured IDE
for Python & Web
development

DOWNLOAD

Free trial

Community

Lightweight IDE
for Python & Scientific
development

DOWNLOAD

Free, open-source

Fig 5.2 Download PyCharm

2. Once the download is complete, run the exe for install PyCharm. The setup wizard should have started. Click "Next".
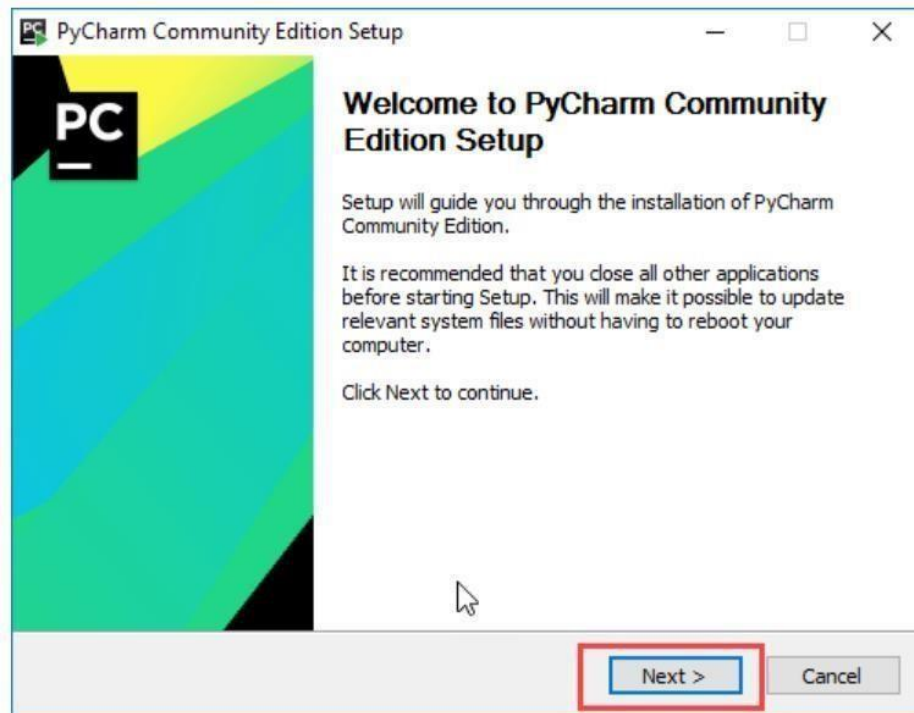
Fig 5.3 PyCharm Community Setup

3. On the next screen, Change the installation path if required. Click "Next".
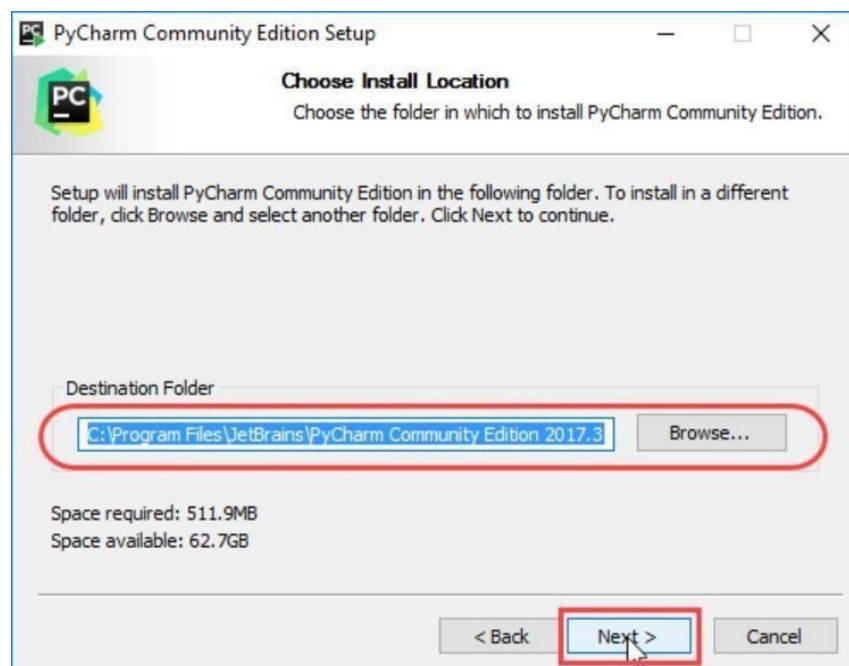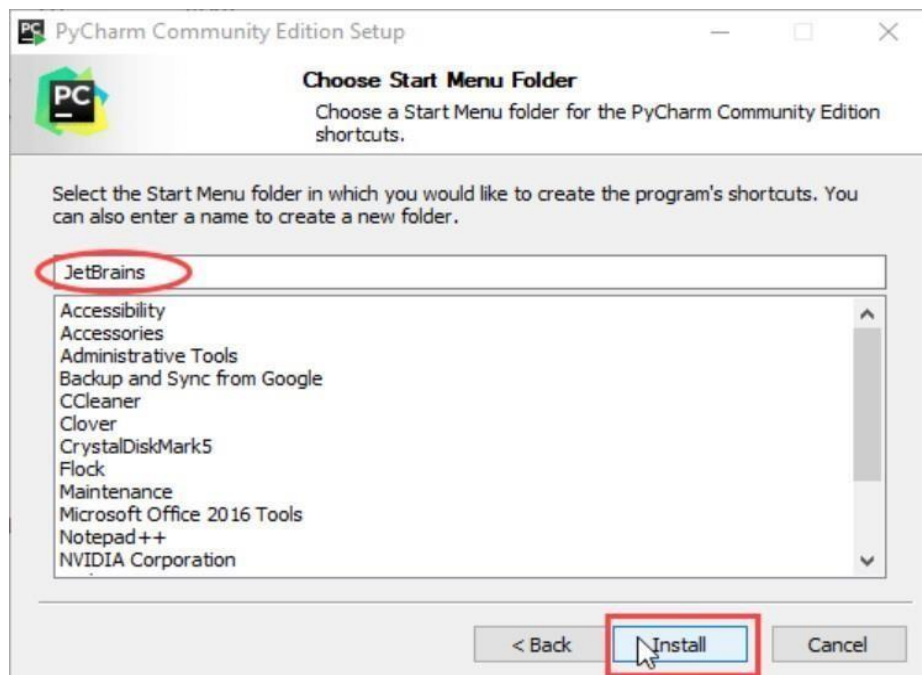

Fig 5.4 Destination Folder

4. On the next screen, you can create a desktop shortcut if you want and click on "Next".
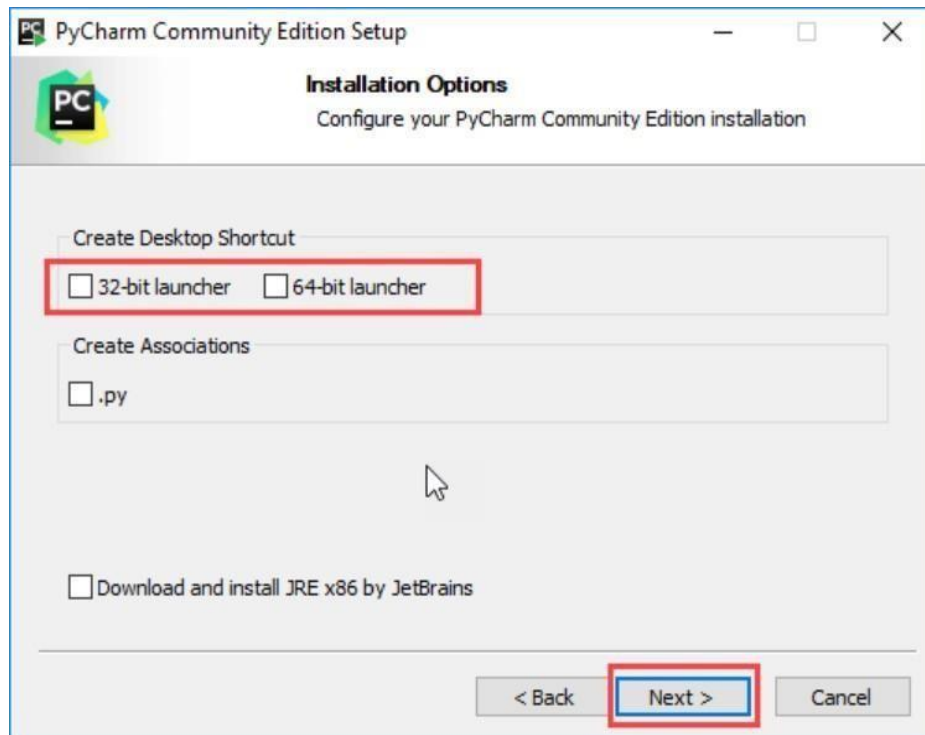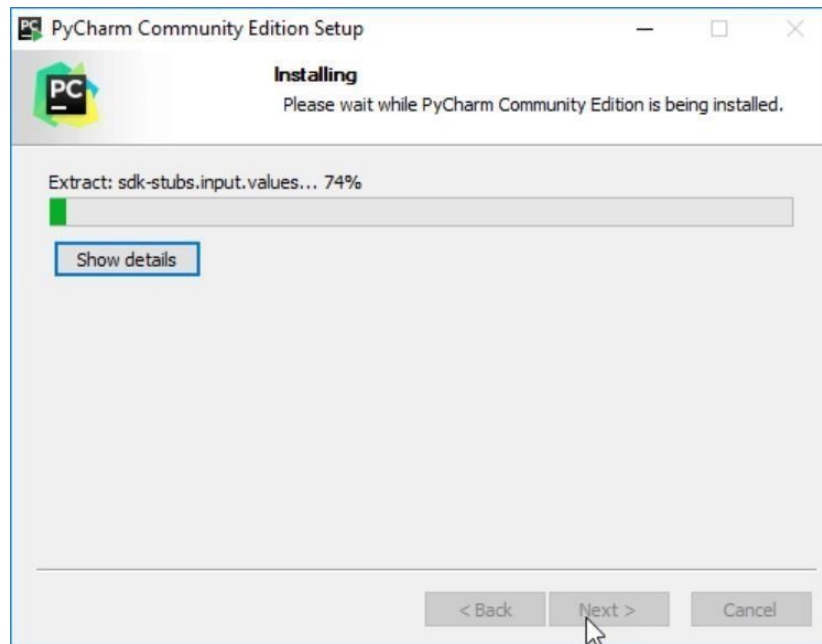

Fig 5.5 PyCharm Configuration

5. Choose the start menu folder. Keep selected JetBrains and click on "Install".

*Fig 5.6 Start Menu Selection*

6. Wait for the installation to finish.

Fig 5.7 Installation

7. Once installation finished, you should receive a message screen that PyCharm is installed. If you want to go ahead and run it, click the "Run PyCharm Community Edition" box first and click"Finish".
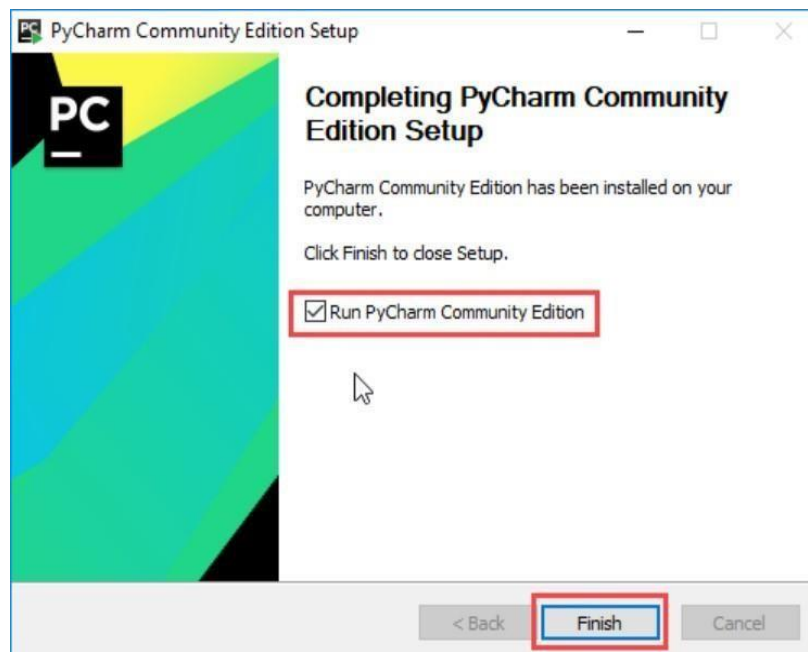
Fig 5.8 Completing PyCharm Setup

8. After you click on "Finish," the Following screen will appear.
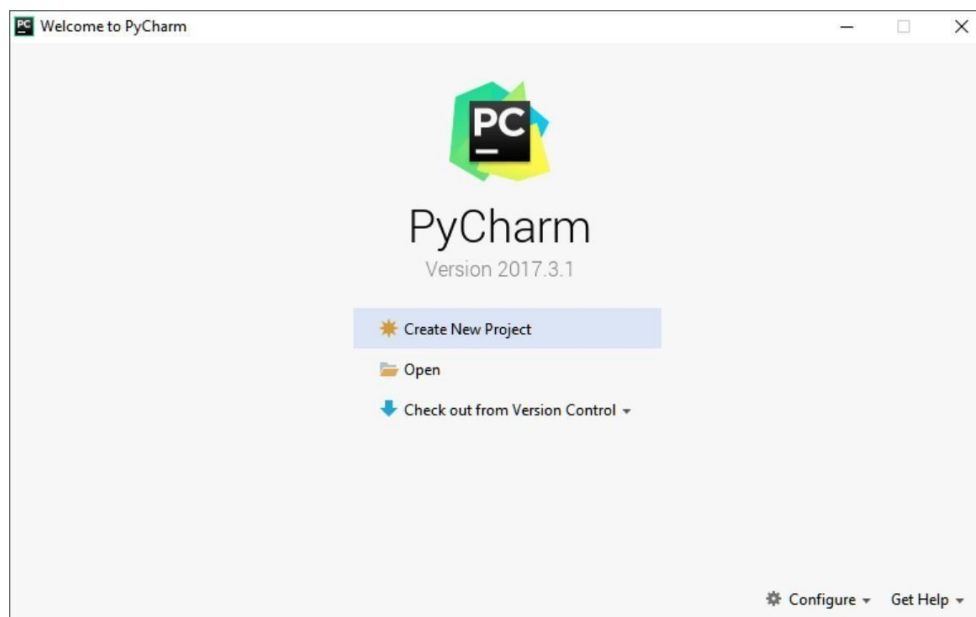
Fig 5.9 Creating a new Project using PyCharm

The Extensible Markup Language XML is a markup language much like HTML or SGML. This is recommended by the World Wide Web Consortium and available as an open standard.XML is extremely useful for keeping track of small to medium amounts of data without requiring a SQL-based backbone. XML Parser Architectures and APIs The Python standard library provides a minimal but useful set of interfaces to work with XML.The two most basic and broadly used APIs to XML data are the SAX and DOM interfaces. Simple API for XML SAX : Here, you register call-backs for events of interest and then let the parser proceed through the document. This is useful when your documents are large or you have memory limitations, it parses the file asit reads it from disk and the entire file is never stored in memory.

Document Object Model DOM API: This is a World Wide Web Consortium recommendation wherein the entire file is read into memory and stored in a hierarchical tree – based form to represent all the features of an XML document. SAX obviously cannot process information as fast as DOM can when working with large files.

On the other hand, using DOM exclusively can really kill your resources, especially if used on a lot of small files. SAX is read-only, while DOM allows changes to the XML file. Since these two different APIs literally complement each other, there is no reason why you cannot usethem both for large projects.

6.SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are varioustypes of tests. Each test type addresses a specific testing requirement

All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive.

To ensure that the system works according to the requisite specification, we need to test experimentally the system with the test data. Then test it with actual data, when the system is working properly and check its performance.

6.1 NEED FOR TESTING:

For the following reasons, testing was essential.

> ➢ To detect the presence of program errors and its failure.

> ➢ As wished-for by its designer the software behavior will be.

> ➢ Conformance with prerequisite specifications or user needs.

> ➢ Evaluates the operational trustworthiness of the system.

> ➢ Echo's the incidences of actual user inputs.

> ➢ Discover the error, which caused the output inconsistency.

> ➢ Checks for identifying faults and insufficiency of the requirements.

> ➢ Check whether the software is operationally practical.

> ➢ Implement the program using the real data processed by the program.

6.2 DIFFERENT LEVELS OF TESTS:

The basic levels of testing are:
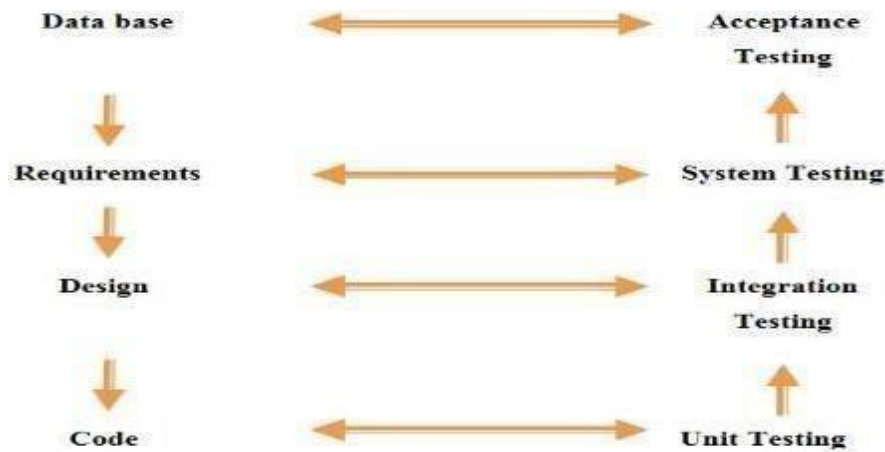
Fig 6.1 Levels of Testing

6.3 TYPES OF TESTS:

6.3.1 Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive.

Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.3.2 Integration testing:

Integration tests are designed to test integrated software components to determine if theyactually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at 134 exposing the problems .

6.3.3 Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

- ✓ **Valid Input:** Identify the human hand to interact with the system through web cam.

- ✓ **Invalid Input:** No hand(s) is identified through web cam.

- ✓ **Functions:** Identified functions must be exercised.

- ✓ **Output:** Mouse and Cursor events are executed.

- ✓ **Systems/Procedures:** The web cam must recognize the hand(s) of the user and detect the movements of the user's finger tips as well as the motions of them. The movements of the cursor and mouse events are handled and executed in the live web cam.

6.3.4 System test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

6.3.5 White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.3.6 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. You cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

6.4 UNIT TESTING:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

6.4.1 Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

6.4.2 Test objectives

• All field entries must work properly.

• Pages must be activated from the identified link.

• The entry screen, messages and responses must not be delayed.

6.4.3 Features to be tested

• Verify that the hands are recognizes.

• Make sure that the web cam is working properly.

• All actions of the mouse and cursors are working.

6.5 INTEGRATION TESTING:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level –interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.
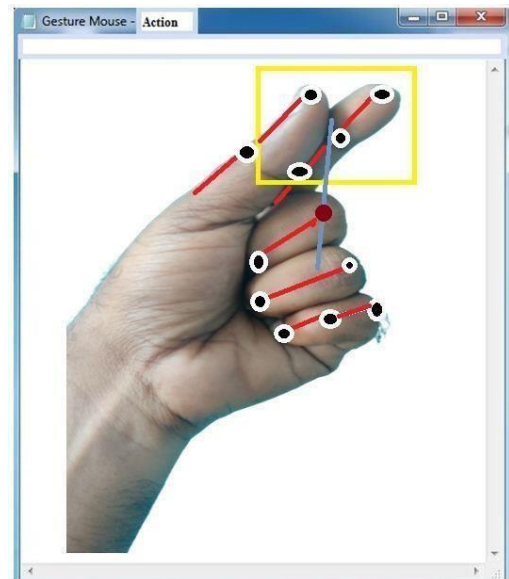
6.6 ACCEPTANCE TESTING:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.
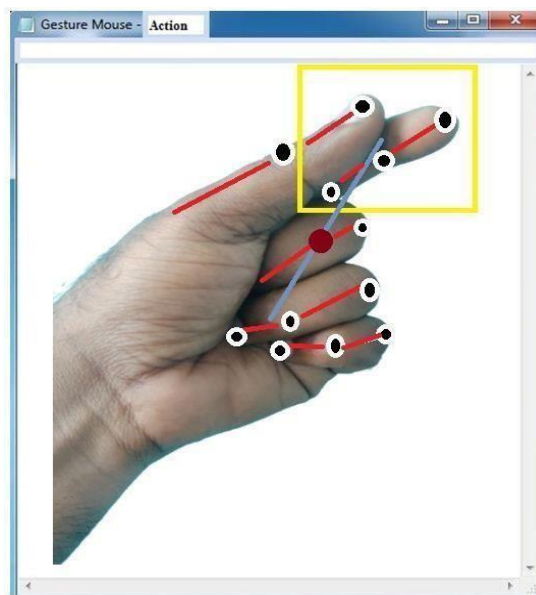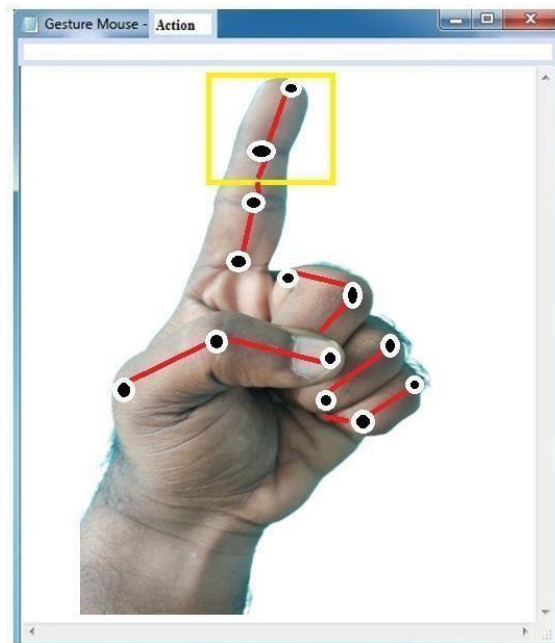
7.EXPERIMENTAL RESULTS
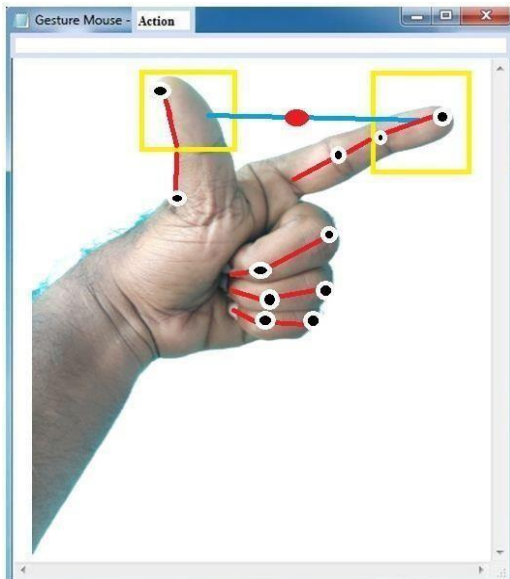
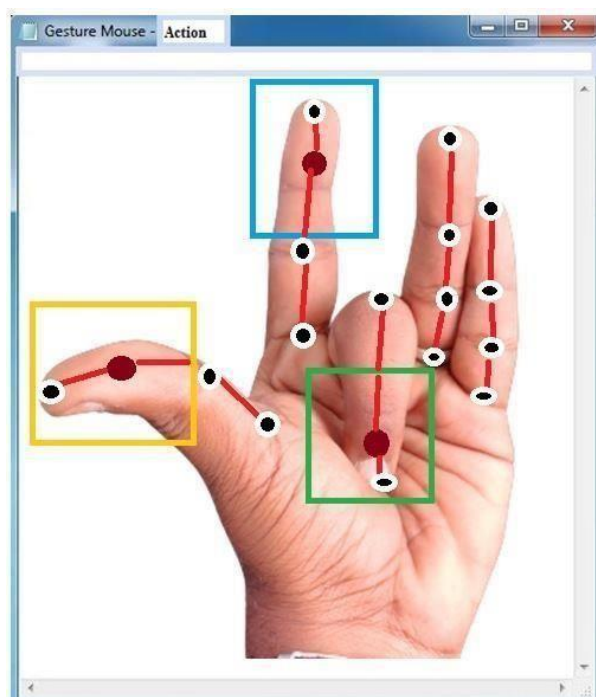*Fig 7.1 Mouse Movement – Open Gesture*

Fig.7.2 Left Button Click – Clouse Gesture

Fig.7.3 Perform Double Click – Clouse Gesture

Fig.7.4 Right Button Click – Open Gesture

Fig 7.5 Scrolling up movement

The most efficient and expressive way of human communication is through hand gesture, which is a universally accepted language. It is pretty much expressive such that the dumb and deaf people could understand it. In this work, real-time hand gesture system is proposed. Experimentalsetup of the system uses fixed position low-cost web camera high- definition recording feature mounted on the top of monitor of computer or a fixed camera on a laptop, which captures snapshot using Red Green Blue [RGB] color space from fixed distance.

In this project we use the edge detection which is generally an image processing discipline. It is used to find the edges of the digital images through some mathematical methods.There are generally two types of edges detections are there. They are Search Based Edge Detection and Zero Crossing Based Edge Detection. There are a finite number of edge detection methods are present. One of the most famous one is Canny Edge Detector.

CONCLUSION

The proposed system will definitely change the way we use the computer. Until now microphone, webcam and mouse are the integral part of the system. This system will completely change the necessity of the mouse which will also leads to the new way of interaction between the human and the computer since there is no need of physical contact with device.

Apart from that, the modules like Open Computer Vision (OpenCV), Py Auto GUI, MediaPipe,Win32api are used. The main objective of the virtual mouse system is to control the mouse cursor functions by using the hand gestures instead of using a physical mouse.

The proposed system can be achieved by using a webcam or a built-in camera which detects the hand gestures and hand tip and processes these frames to perform the particular mouse functions.

From the results of the model, we can come to a conclusion that the proposed AI virtual mouse system has performed very well and has a greater accuracy compared to the existing models and also the model overcomes most of the limitations of the existing systems.

Since the proposed model has greater accuracy, the virtual mouse can be used for real-world applications, and also, it can be used to reduce the spread of COVID-19, since the proposed mouse system can be used virtually using hand gestures without using the traditional physical mouse.

FUTURE SCOPE

The proposed virtual mouse has some limitations such as small decrease in accuracy of the right click mouse function and also the model has some difficulties in executing scrolling particularly. Also, the virtual mouse cannot perform the scroll down as well as drag operations, because of implementation of these events are too hard to handle with the gestures. These are some of the limitations of the proposed virtual mouse system, and these limitations will be overcome in our future work.

Even though the proposed system can recognize both hands of a human, the problem arises when multiple hands are recognized during the execution. The program always in execution (unless we terminate it), which causes problems in as the unwanted events may be trigger.

Furthermore, the proposed method can be developed to handle the keyboard functionalities along with the mouse functionalities virtually which is another future scope of Human-Computer Interaction (HCI).

REFERECES

[1]   A. Banerjee, A. Ghosh, K. Bharadwaj, H. Saikia, Mouse Control using a Web Camera based on Colour Detection, Int. J. Comput. Trends Technol. 9 (2014) 15–20. doi:10.14445/22312803/ijctt-v9p104.

[2]   A.A. Argyros, M.I.A. Lourakis, Vision-based interpretation of hand gestures for remote control of a computer mouse, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 3979 LNCS (2006) 40–51. doi:10.1007/11754336_5.

[3]   Yu-Luen Chen, Application of tilt sensors in human-computer mouse interface for people with disabilities, IEEE Trans. Neural Syst. Rehabil. Eng. 9 (2002) 289–294. doi:10.1109/7333.948457.

[4]   D.G. Evans, R. Drew, P. Blenkhorn, Controlling mouse pointer position using an infrared head-operated joystick, IEEE Trans. Rehabil. Eng. 8 (2000) 107–117. doi:10.1109/86.830955.

[5]   Y. Chen, E.R. Hoffmann, R.S. Goonetilleke, Structure of Hand/Mouse Movements,

IEEE Trans. Human-Machine Syst. 45 (2015) 790–798. doi:10.1109/THMS.2015.2430872.

[6]   K. Arai, R. Mardiyanto, Camera as Mouse and Keyboard for Handicap Person with Troubleshooting Ability, Recovery, and Complete Mouse Events, Int. J. Hum. Comput. Interact. 1(2010) 46–56. http://www.cscjournals.org/manuscript/Journals/IJHCI/Volume1/Issue3/IJHCI-13.pdf.

[7]   M. Mangaiyarkarasi, A. Geetha, Cursor Control System Using Facial Exoressions for Human-Computer Interaction, Int. J. Comput. Sci. …. (2014).

[8]   A. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E. "Computer vision-based mouse", Acoustics, Speech, and Signal Processing, Proceedings. (ICASS). IEEE International Conference, 2002

[9]   Y. Sato, Y. Kobayashi, H. Koike. Fast tracking of hands and finger tips in infrared images for augmented desk interface. In Proceedings of IEEE, International Conference on Automatic Face and Gesture Recognition (FG), 2000. pp. 462-467.

[10]   D.G. Evans, R. Drew, P. Blenkhorn, Controlling mouse pointer position using   an   infrared head-operated   joystick, IEEE Trans. Rehabil. Eng. 8(2000)107–117. doi:10.1109/86.830955.

[11]   Y. L. Chen, F. T. Tang, W. H. Chang, M. K. Wong, Y.Y. Shih, and T. S. Kuo, "The new design of an infrared-controlled human–computer interface for the disabled," IEEE Trans. Rehab. Eng., vol. 7, pp. 474–481, Dec. 1999.

[12]   Y. L. Chen, T. S. Kuo, W. H. Chang, and J. S. Lai, "A novel position sensors-controlled computer mouse interface for the disabled," in Proc. IEEE Engineering Medicine Biology, Chicago, IL, July 2000, pp. 3945–3948.

[13]   R. D. Richard, B. Stein, B. J. Andrews, K. B. James, and M. Wieler, "Application of tilt sensors in functional electrical stimulation," IEEE Trans. Rehab. Eng., vol. 4, pp. 63–71, Mar. 1996.

[14]   Z. A. Keirn and J. I. Aunon, "Alternative modes of communication between man and his surroundings," IEEE Trans Biomed. Eng., to be published.

[15]   The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People With Severe Disabilities Margrit Betke, Member, IEEE, James Gips, Member, IEEE, and Peter Fleming IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING, VOL. 10, NO. 1, MARCH 2002.

[16]   Cheoljong Yang; Yujeong Jang ; Jounghoon Beh ; Han, D. ; Hanseok Ko: "Gesture recognition using depth-based hand tracking for contactless controller application", Consumer Electronics (ICCE), 2012 IEEE International Conference on, On Page(s): 297-298.

[17]   G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, Inc., 2008.