Cursor Control Interface using Gestures

A Project Report

For

CSE4015 – Human Computer Interaction

Group Members:
Sai Saathvik - 17BCE0012
Vijay Surya - 17BCE0147
Anurag Maheshwari – 17BCE0593
Arjun Dixit – 17BCE0932

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Under the Guidance of
Prof. Senthilnathan

Winter Semester 2019 - 2020

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION

We hereby declare that the project entitled "**Cursor Control Interface using Gestures**" submitted by Sai Saathvik, Vijay Surya, Anurag Maheshwari and Arjun Dixit to the School of Computer Science and Engineering, Vellore Institute of Technology, Vellore towards the partial fulfillment of the requirements for the course CSE4015 – Human Computer Interaction is a record of bonafide work carried out by me under the supervision of Prof. Senthilnathan. We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for any other course or purpose of this institute or of any other institute or university.

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Certificate

The project report entitled "**Cursor Control Interface using Gestures**" submitted by Sai Saathvik (17BCE0012), Vijay Surya (17BCE0147), Anurag Maheshwari (17BCE0593) and Arjun Dixit(17BCE0932) has been found satisfactory in terms of scope, quality and presentation as partial fulfillment of the course CSE4015 – Human Computer Interaction Vellore Institute of Technology, Vellore-14, India.

Prof. Senthilnathan

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

ACKNOWLEDGEMENT

The project "**Cursor Control Interface using Gestures**" was made possible because of inestimable inputs from everyone involved, directly or indirectly. We would first like to thank our guide **Prof. Senthilnathan**, who was highly instrumental in providing not only a required and innovative base for the project but also crucial and constructive inputs that helped make my final product. Our guide has helped us perform research in the specified area and improve our understanding in the area of reinforcement learning and we are very thankful for his support all throughout the project.

We would also like to acknowledge the role of the HOD, **Prof Santhi V**, who was instrumental in keeping us updated with all necessary formalities and posting all the required formats and document templates through the mail, which we were glad to have had.

It would be no exaggeration to say that the Dean of SCOPE, **Prof Saravanan R**, was always available to clarify any queries and clear the doubts we had during the course of my project.

Finally, we would like to thank **Vellore Institute of Technology**, for providing us with a flexible choice and execution of the project and for supporting our research and execution related to the project.

1. AIM

To develop a cursor-controlled interface using gestures which will be differentiated based on colors and there by the task assigned to them.

2. ABSTRACT

This project presents an approach to develop a real-time hand gesture recognition enabling human computer interaction. It is “Vision Based” that uses only a webcam and Computer Vision (CV) technology, such as image processing that can recognize several hand gestures. The applications of real time hand gesture recognition are numerous, due to the fact that it can be used almost anywhere where we interact with computers ranging from basic usage which involves small applications to domain-specific specialized applications. Gesture recognition is an area of active current research in computer vision. Existing systems use hand detection primarily with some type of marker. Our system, however, uses a real-time hand image recognition system.

TABLE OF CONTENTS

	Title	Page No.
i	Declaration	2
ii	Certificate	3
iii	Acknowledgement	4
iv	Table of Contents	6
1.	Aim	5
2.	Abstract	5
3.	Software Requirements	7
	3.1 Functional Requirements	7
	3.2 Non-Functional Requirements	8
	3.3 Hardware Requirements	9
4.	Modules	9
	4.1 Skin Detection	9
	4.2 Hand Contour Extraction	9
	4.3 Hand Tracking	9
	4.4 Gesture Recognition	9
	4.5 Cursor Control	10
5.	Stakeholder Identification	
	5.1 Requirement Facilitators	
	5.2 Customers	10
	5.3 Project Team	
	5.4 End Users	
	5.5 Maintenance Team	
6.	Diagrams	10
	6.1 Pert Chart	10
	6.2 Use Case Diagram	11
	6.3 Data Flow Diagram	12
	6.4 Entity Diagram	13
	6.5 UI Representation using Storyboarding	14,15
7	Result	16
8	Conclusion	18
9	Future Enhancement	18
10	References	19

3. SOFTWARE REQUIRMENT

3.1 Functional Requirements (FR)

Functional requirements specify the main technical functionalities and specifications that the system should incorporate.

3.1.1 FR-01 Face Detection

This software shall utilize a face detection system to filter out faces from the video capturing device. By applying face detection, the system can disregard the region where the face is located and thus reducing the amount of calculation needed to perform hand detection. The face detection unit will be implemented with the help of OpenCV

3.1.2. FR-02 Skin Detection

This software shall perform skin colour detection and filter out all objects that do not contain the colour of skin. By filtering object of non-skin coloured, the system can then use its remaining resources and focus on hand detection and gesture recognitions. This also allows the system to pinpoint possible locations of user's hands. The skin detection can be achieved by using the Marvin Image Processing Framework (discussed in section 4.1.2).

3.1.3. FR-03 Filtered Object Detection

Once the program has filtered out most of the unwanted parts of the picture after using the skin detection, the software shall read and recognize "clusters" skin coloured objects also known as "blobs".

3.1.4. FR-04 Object Location

Upon detection, the system shall be able to compute the location of the object using simple trigonometry math.

3.1.5. FR-05 Hand Calibration

Depending on user's preferences, the system shall perform adjustments according to user's dominant hand. This means that if the user is right handed, the mouse control gesture mode should be recognized near the right side of the face instead of the whole field of view. This is achieved through trigonometry math conversions.

3.1.6. FR-06 Mouse Movement Gesture Control Mode

After obtaining the location of the hand, the software shall use the detected location as the mouse cursor point. As the user moves his/her hands, the mouse should follow promptly on the screen.

3.1.7. FR-07 Browsing Gesture Control Mode

The software shall allow the user to use the "Browsing Gesture Mode". In this mode, user's hand gesture will only be recognized for commands including previous page, next page, scroll up and scroll down.

3.2 Non-Functional Requirements (NFR)

Non-functional requirements specify the criteria in the operation and the architecture of the system.

3.2.1 NFR-01 Efficiency in Computation

This software shall minimize the use of Central Processing Unit (CPU) and memory resources on the operating system. When HGR is executing, the software shall utilize less than 80% of the system's CPU resource and less than 100 megabytes of system memory.

3.2.2 NFR-02 Extensibility

The software shall be extensible to support future developments and add-ons to the HGR software. The gesture control of HGR shall be at least 50% extensible to allow new gesture recognition features to be added to the system.

3.2.3. NFR-03 Portability

The HGR software shall be 100% portable to all operating platforms that support Java Runtime Environment (JRE). Therefore, this software should not depend on the different operating systems.

3.2.4. NFR-04 Performance

This software shall minimize the number of calculations needed to perform image processing and hand gesture detection. Each captured video frame shall be processed within 350 milliseconds to achieve 3 frames per second performance.

3.2.5. NFR-05 Reliability

The HGR software shall be operable in all lighting conditions. Regardless of the brightness level in user's operating environment, the program shall always detect user's hands.

3.2.6. NFR-06 Usability

This software shall be easy to use for all users with minimal instructions. 100% of the languages on the graphical user interface (GUI) shall be intuitive and understandable by non-technical users.

3.3. Hardware Requirements

HGR software does not require special equipment except for a personal computer (PC) and a webcam. The CPU of this computer should have at least two cores to handle the enormous amount of calculations needed for the image processing unit.

4. Modules

4.1. Skin Detection

This module is basic part of the whole system. As it is responsible for distinguishing between the skin and other colour of the surrounding. A person using this whole set up will come in front of the screen and in removing it in order stop hindrance while other colours are used to perform various functionality.

4.2. Hand Contour Extraction

Herein we deal with detecting layout basically the structure of user's hand in order to separate it while the colour detection scheme goes in further to develop a better view and understanding the movement. The key requirement behind this model is to ensure there is proper and smooth flow between the system and user thereby avoiding lags if any.

4.3. Hand Tracking

In this part using the image obtained by the web camera of laptop we try to detect the user hand basically referring to any movement what so ever which further helps in performing various tasks. For example, movement of finger moves the cursor. So such tasks need precision therefore there needs to proper understanding and great efficiency behind them so as to achieve better and more appropriate results.

4.4. Gesture Recognition

In this module we designate different gestures for different task and whenever such a flow is observed by the system the designated action for that particular gesture takes place.

4.5. Cursor Control

Lastly, this module is to increase the efficiency by which system and the web camera and the gestures are linked.

5. Stakeholder Identification

5.1 Requirement Facilitators

5.2 Customers

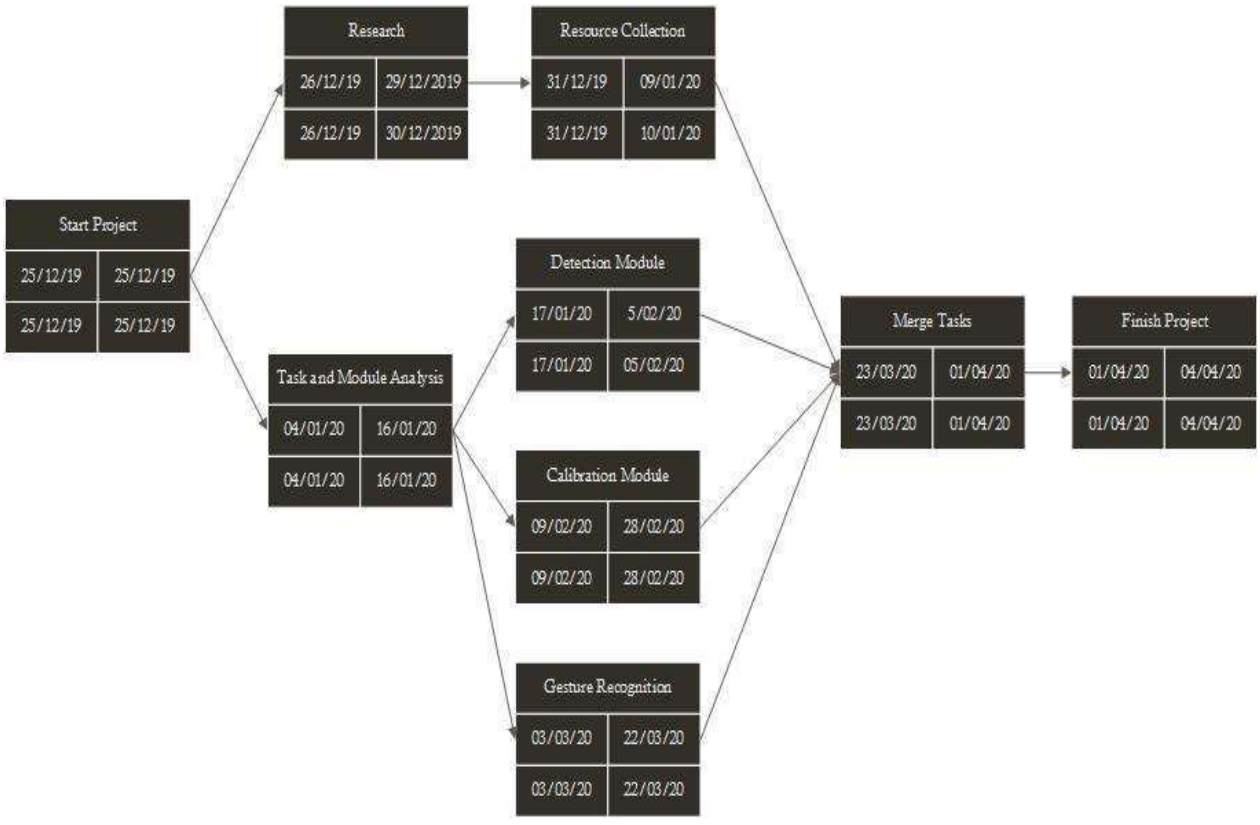
5.3 Project Team

5.4 End Users

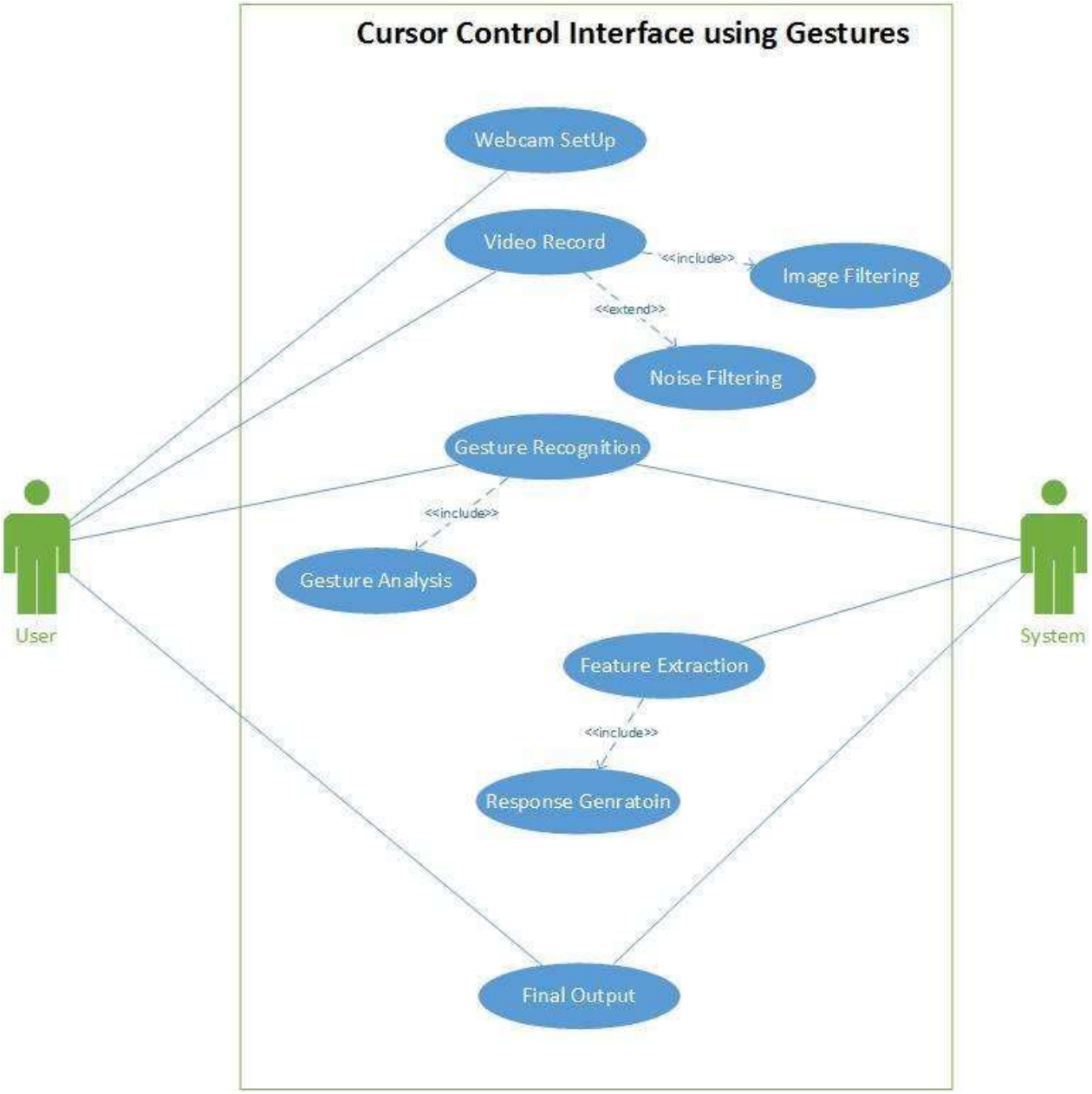
5.5 Maintenance Team

6. Diagrams –

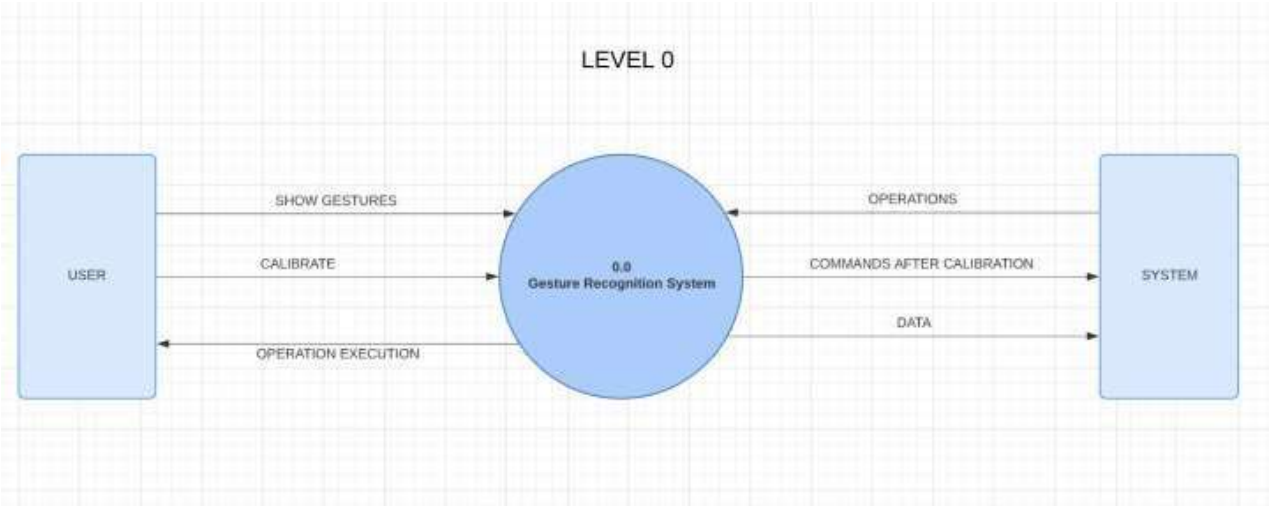
Pert Chart

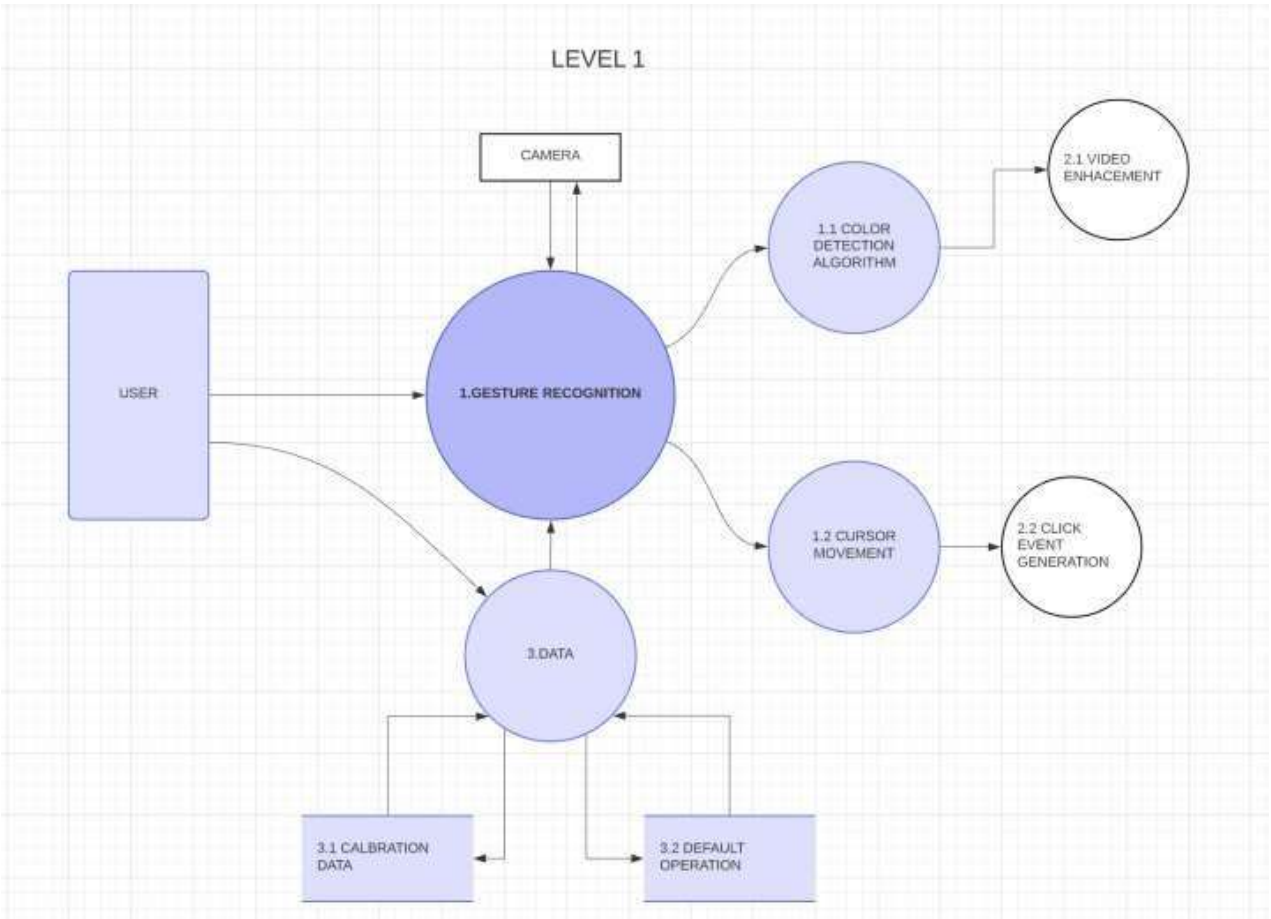


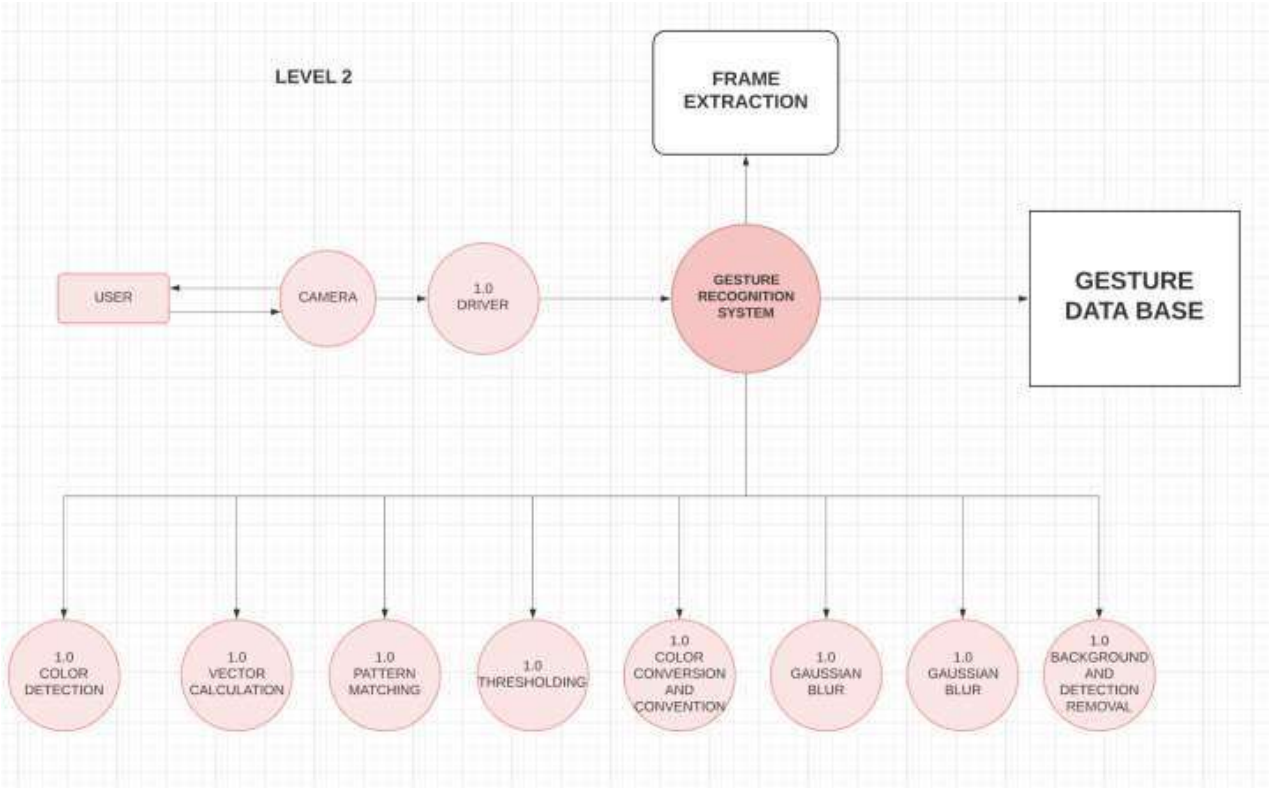
Use Case Diagram –



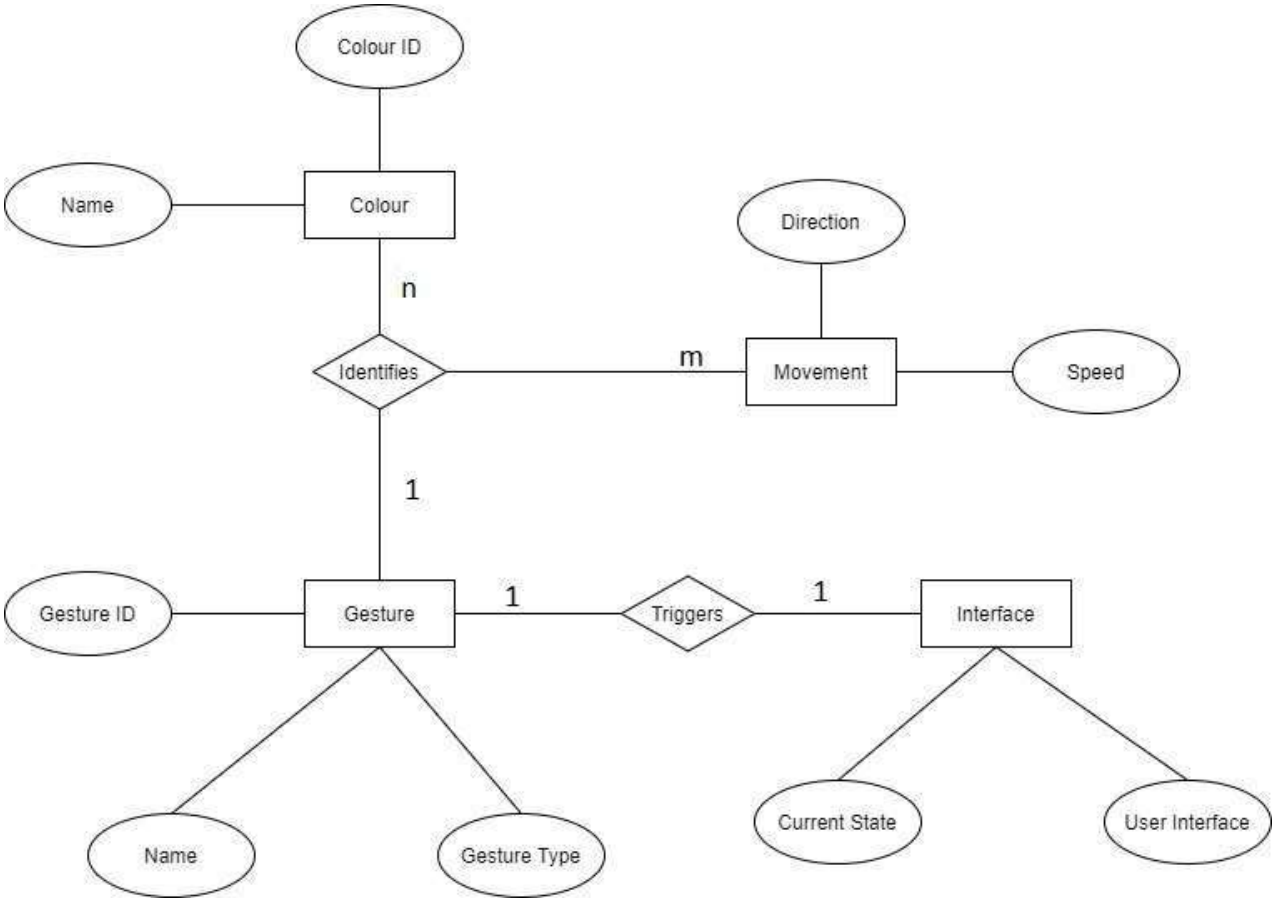
Data Flow Diagrams –



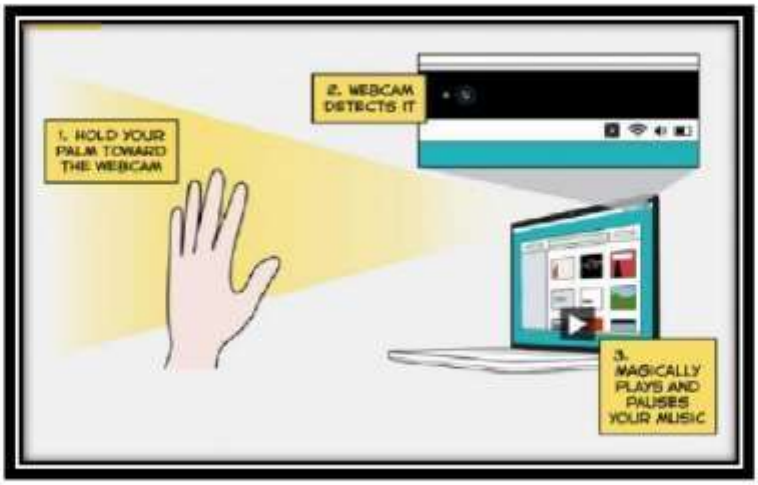




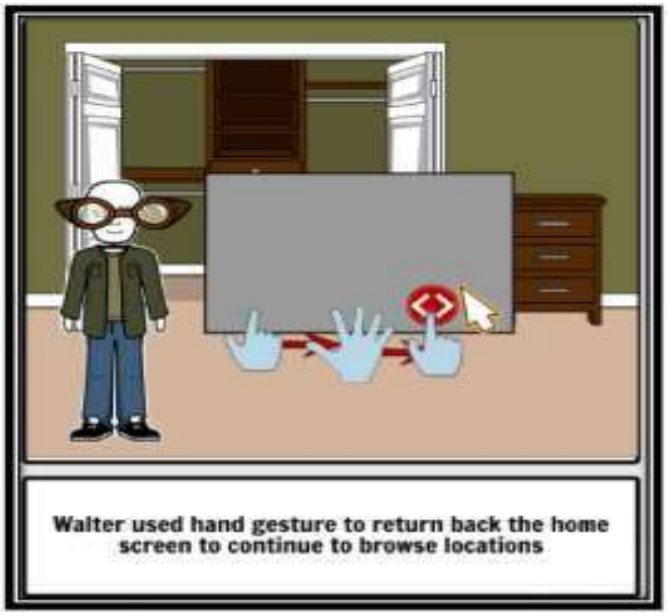
Entity Diagram –



UI Representation based on Storyboarding –




MASTER HOW TO
PLAY AND PAUSE
YOUR MUSIC--
--AND CONTROL YOUR MUSIC


Walter used hand gesture to return back the home screen to continue to browse locations

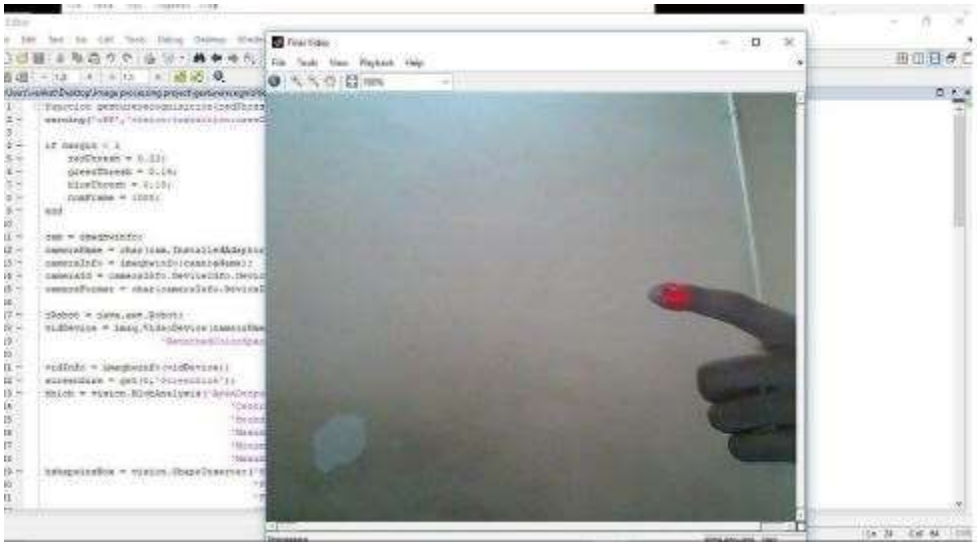
7. RESULT

Finally, once the whole setup is completed and run on a system. The Web camera turns on and start the recognition part as soon as colour identification gets completed various actions can be carried out as per they are assigned. The table below describes how the system behaves for different actions –

TESTING

INPUT	ACTIONS PERFORMED
One red colour	to control the pointer position
one green colour	for scroll up and down
One blue colour	For left click
Two blue colour	For right click
Three blue colour	For double click
Yellow color	No action
Uncontrolled environments	Tested with background noise
White background	Tested with white background
Out of camera view	Color goes out of the scope of camera

Now, below are some screenshots depicting the working of the system as a whole a



8. Conclusion –

We have successfully developed an object tracking based virtual mouse application, which has been implement just using webcam found in almost all devices, thus making the secondary devices obsolete. Only a device having a webcam on it, is enough to access everything that we otherwise were able to. Our application has been developed in MATLAB environment using MATLAB Image Processing Toolbox.

This object tracking technology has wide range of applications ranging from fields like augmented reality, gaming, prosthetics and computer graphics. It is also becoming quite popular among artists for drawing amazing digital canvas. Also, if we talk about prosthetics, it can help a person with no limbs give instructions to a device. In the field of gaming, Facebook has successfully developed OCULUS Rift device which has made the gaming realistic than ever, where a person's motions are tracked and are interpreted the. We have come a far away from using traditional secondary devices to interact with any device, but now with the help of this technology the application is limitless.

9. Future Enhancement -

We can further try to enhance our application mainly focusing on improving the efficiency, accessibility and security of our hand recognition and gesture application. We can try incorporating facial recognition system, enabling only those users to access the system which are already Authorized. Also, implementing better noise reduction and background cancelling techniques of Image Processing can be incorporated to enhance our application. Also, we'll try to add as many gestures and their combinations as possible to increase the speed of accessibility. There will be Hands free mode which allow to surf through the device and perform any action just using the gestures.

10. REFRENCES

- Hand Gesture Recognition and Cursor Control
  Salian, Shashank & Serai, Dhiren & Ganorkar, Pranav. (2015). Hand Gesture Recognition and Cursor Control. 10.13140/RG.2.1.3185.0082.
- Abhik Banerjee, Abhirup Ghosh, Koustuvmoni Bharadwaj," Mouse Control using a Web Camera based on
  Color Detection" ,IJCTT, vol.9, Mar 2014
- Angel, Neethu.P.S,"Real Time Static & Dynamic Hand Gesture Recognition", International Journal of
  Scientific & Engineering Research Volume 4, Issue3, March-2013.
- Chen-Chiung Hsieh and Dung-Hua Liou," A Real Time Hand Gesture Recognition System Using Motion
  History Image"icsps, 2010