Gesture Controlled Snake Game

Yash Dhamecha 111608067, Swanand Gadekar 111608069, Rajat Pawar 111608074

April 15,2019

INDEX

1 Abstract:

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand. The need to improve communication between humans and computers has been instrumental in defining new communication models, and accordingly, new ways of interacting with machines. The use of gestures as a means of communication has been a challenging task. The latest generation of smartphones boasts powerful processors and built-in video cameras, making them capable of executing complex and computationally demanding applications. Thus, the integration of gesture recognition systems in smartphone applications might be a close reality. Current focuses in the field include emotion recognition from the face and hand gesture recognition. Many approaches have been made using cameras and computer vision algorithms to interpret sign language. However, the identification and recognition of posture, gait, proxemics, and human behaviors is also the subject of gesture recognition techniques. Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive text user interfaces or even GUIs (graphical user interfaces), which still limit the majority of input to keyboard and mouse. In this project, we present studies of a gesture recognition prototype. We present the idea to implement object tracing for drawing figures.

2 Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

Through this project, we aim to understand gesture recognition and its practical basic knowledge to present a working prototype. Such that we can use our hand as mouse pointer.

2.1.2 Document Conventions

Bold text has been used to emphasize section and sub-section headings. Highlights is to point out words in the glossary and italicized text is for used to label and recognize diagrams

2.1.3 Intended Audience and Reading Suggestions

This document is to be read by our project managers. The SRS has been organized approximately in order of increasing specificity. The developers and project managers need to become intimately familiar with the SRS

2.1.4 Product Scope

This idea can be extended to blind people or physically handicapped people so that can access daily requirements. It can be used in Automotive sector, Consumer electronics sector, Transit sector, To unlock smartphones, Defence, Home automation, Sign language Interpretation.

2.1.5 References

1.Research papers of IV – Labs VNIT.
2. Video Reference https://youtu.be/TwYUm7i61JM
3. IEEE Research Papers
https://ieeexplore.ieee.org/document/6179213/
https://ieeexplore.ieee.org/document/7176405/

2.2 Overall Description

2.2.1 Product Perspective

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand. In this project, we present studies of a gesture recognition prototype. We present the idea to implement object tracing for drawing figures.

2.2.2 Product Functions

Most of the non-disabled people do not know signed type of language. Therefore, for deaf people performing daily activities turns out to be hard, especially in public areas. Additionally, it is difficult and expensive to make non-disabled people learn sign language. For these reasons, automatic systems that translate sign language into text are required. Some systems that recognize gestures and translate sign language into text have been proposed. Most of these systems use image processing techniques to perform the translation. Other approaches use different kinds of sensors to acquire the orientation of the hand and fingers.

2.2.3 User Classes and Characteristics

Anyone can use this software but it will be extremely beneficial to handicapped people if developed on large stage.

2.2.4 Operating Environment

Terminal Based, using python programming language.
Microsoft Windows 7/8/10 (32-bit or 64-bit) / Linux.
2 GB RAM minimum, 8 GB RAM recommended.
2 GB of available disk space minimum, 4 GB Recommended.
1280 x 800 minimum screen resolution.
Python.
Knowledge of Libraries Used in Machine Learning.

2.2.5 Design and Implementation Constraints

1. Synchronization
2. Communication protocol
3. Time requirement (Need to improve time to time)

2.3 External Interface Requirements

2.3.1 User Interfaces

The user interface screen is described in table:

Screen name	Description
Terminal	Interface provided by Linux

2.3.2 Hardware Interfaces

This software is Terminal based only.

2.3.3 Software Interfaces

This software is terminal based which we can access directly by running its executable file. User can communicate with OS using this this software using web cam provided on PCs or laptops. Basic rules provided for accessing tracker is sufficient for user to use this software efficiently.

2.3.4 Communications Interfaces

Software contains user friendly Interface which will communicate with user directly. User need to install certain libraries for this software to run on his/her system. User also needs web cam on its PC.

2.4 System Features

2.4.1 System Feature 1

Description
1. Colour Recognization. Used to recognize different colors on the basis of different rgb values.
2. Object Detection. It recognises different object on basis of specific properties like shape, size etc
3. Object Tracing Used to trace the movements of object by recording it's position at different intervals of time
4. Gesture Recognization Recognises hand /finger movement and execute appropriate action as per gestures
5. Video Processing. It's a part of computer vision, used to filter out important part of video so that it can be used in future
6. Image recognization. It's a part of video processing in which consecutive frames of video are taken and previous frame of video is used to detect object in next frame

Stimulus/Response Sequencesuser performs some action first the gesture is captured by videocam and sent for video processing, the processed video goes through some filtering and image processing. Then the processed image is used to identify the gesture and appropriate action is executed accordingly.

Functional Requirements
For proper functioning of our project we require certain python libraries like tensor flow, openCV are required

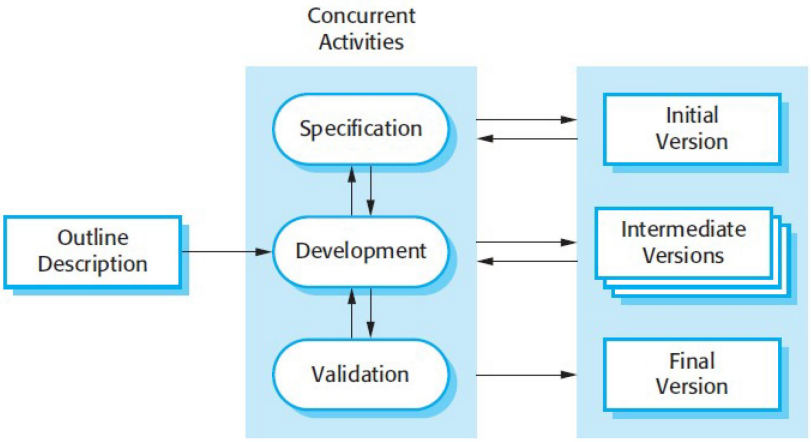
2.4.2 System Feature 2 (and so on)

2.5 Other Requirements

Appendix A: Glossary
IEEE - Institute of electrical and electronics engineering
VNIT - Vishweshwarya national institute of technology

Python - High level programming language generally used in machine learning
ML - Machine learning
Tensorflow - Python library used in ML
openCV - Python library used in ML

Appendix B: Analysis Models
Incremental process model will be used for our project.



Appendix C: To Be Determined List
Research papers of IV – Labs VNIT.
Video Reference
https://youtu.be/TwYUm7i61JM
IEEE Research Papers
https://ieeexplore.ieee.org/document/6179213/
https://ieeexplore.ieee.org/document/7176405/

3 Literature Survey

Abstract

In order to offer new possibilities to interact with machine and to design more natural and more intuitive interactions with computing machines, our research aims at the automatic interpretation of gestures based on computer vision. In this paper, we propose a technique which commands computer using six static and eight dynamic hand gestures. The three main steps are: hand shape recognition, tracing of detected hand (if dynamic), and converting the data into the required command. Experiments show 93.09

3.1 Introduction

Gesture recognition is the mathematical interpretation of a human motion by a computing device. Modern research of the control of computers changes from standard peripheral devices to remotely commanding computers through speech, emotions and body gestures . Our application belongs to the domain of hand gesture recognition which is generally divided into two categories i.e. contact-based and vision-based approaches. The second type is simpler and intuitive as it employs video image processing and pattern recognition. The aim is to recognize six static and eight dynamic gestures while maintaining accuracy and speed of the system. The recognized gestures are to command the computer.

3.2 AREAS OF GESTURE RECOGNITION

3.2.1 A. Hand Gesture Recognition

This are of Gesture technology is the most prominent, it is highly adaptive interface in which the detection is done on the movement of fingers and hand. The detection can be of static gestures as well as dynamic gestures (using effective tools accordingly). Complex machines can be operated by this recognition without any physical contact between the user and the machine.

3.2.2 B. Facial Gesture Recognition

Human facial expressions can be detected by this technology . This tells the current status of mood of the user whether he is happy, sad, anguish, enthusiastic. This technique is mainly used for personal security and law enforcement.

3.2.3 C. Sign Language Recognition

Specific Human hand gestures are defined for particular alphabets/texts that software is capable of transcribing. Communication between user and machine become very easy and effective by this technology for deaf people.

3.2.4 D. Devised Gesture Technology

A glove stylus or some position tracker kind of device is tightly connected to wrist of the user . The other rend of glove or stylus is connected to the machine by connectors like RS-232 and then by certain data acquisitions and measurements the position and orientation of hand is detected. Sensors are many times used for giving information more accurately.

3.2.5 E. Vision Based Technology

There are two main methods: -

1. Model Based method: Three-dimensional model of hand is created then its recognition is performed.
2. Image Based method: First of all the image is captured by the camera and after that detection is done

3.2.6 F. Electric Field Sensing

Sensors of electric field are used for measuring the proximity of human body. These calculations are mainly to find out the distance of human body relative to an object or from an object. This technology has a vast scope in industries , serving numerous application

3.3 RECOGNITION ALGORITHM

Sensor data is a time series, so this article selects the RNN, LSTM and GRU models which have great success in the timing problem as the gesture recognition model. A brief introduction to these algorithms.

The network structure of RNN is shown in Fig. 6. The purpose of the RNN is to use to process sequence data. In the traditional neural network model, from the input layer to the hidden layer to the output layer, the layers are fully connected, and the nodes between each layer are disconnected. However, this kind of neural network is incapable of many timing problems. RNN is called recurrent neural network, that is, the current output of a sequence is also related to the previous output. The specific manifestation is that the network will memorize the previous information and apply it to the calculation of the current output. The nodes between the hidden layers are connected, and the input of the hidden layer includes not only the output of the input layer but also the output of the hidden layer at the previous time. In theory, RNN can handle any length of sequence data. However, in practice, in order to reduce complexity, it is usually assumed that the current state is only related to the previous several states. The formula for this network is derived as follows:

$$ht = f((wh)(h[t\hat{a}1]) + (wi)(xt))(5)$$

$$y = f(w(ht))(6)$$

x t represents the input of t=1,2,3...,

h t is the hidden layer step t,it is the networkâs memory unit.

y t is the output of step t. Where f is generally a nonlinear activation function such as Tanh or ReLU. The recurrent neural network has a certain memory function, but it does not handle the long-term dependency problem well. Long-term dependency is such a problem that it is difficult to learn relevant information when the prediction point is far from the relevant information of the dependence. Long Short Term Mermory network (LSTM) is a special type of RNN that can solve long-term dependencies. The formula for this network is derived as follows:

$$z = tanh(w\hat{A}\Delta[xt, ht\hat{a}1] + bc)(7)$$

$$zi = \ddot{I}(wi\hat{A}\Delta[xt, ht\hat{a}1] + bi)(8)$$

$$zf = \ddot{I}(wf\hat{A}\Delta[xt, ht\hat{a}1] + bf)(9)$$

$$zo = \ddot{I}(wo\hat{A}\Delta[xt, ht\hat{a}1] + bo)(10)$$

$$ct = zfct - 1 + ziz$$

$$ht = zotanh(ct)$$

$$yt = \ddot{I}(w'ht)$$

x t represents the input of t=1,2,3..., h t , c t is the hidden layer step t,it is the networkâs memory unit. y t is the output of step t. In this article, we also discuss another variant of the RNN called the Gated Recurrent Unit (GRU). GRU maintains the effect of LSTM while making the structure simpler, so it isalso very popular. The network structure of LSTM is shown in Fig 8. The formula for this network is derived as follows:

$$r = \ddot{I}(wr\hat{A}\Delta[xt, ht\hat{a}1] + br)(14)$$

$$z = \ddot{I}(w\hat{A}\Delta[xt, ht - 1] + bh)(15)$$

$$h = tanh(wh\hat{A}\Delta[xt, r\hat{A}\Delta ht\hat{a}1] + bh)$$

$$ht = (1\hat{a}z)ht\hat{a}1 + zh$$

$$yt = \ddot{I}(w\hat{A}\Delta ht)$$

x t represents the input of t=1,2,3..., h t is the hidden layer step t,it is the networkâs memory unit. y t is the output of step t. This article mainly includes these three algorithms and compares these three algorithms. Use these three algorithms to train the preprocessed gesture data separately. We will get the accuracy and loss of the three models.

3.4 TRACING OF DETECTED HAND (IF DYNAMIC)

Recognition of a static gesture requires only the hand shape. Once hand shape is classified as static gesture by the trained classifier, command is given to the computer. Unlike static gesture, dynamic gesture requires both the hand shape as well as the motion of hand. For tracing dynamic hand gestures, hand area is segmented out using HSV (Hue, Saturation, Value) skin color algorithm in a frame, followed by cropping blob area. Centroid of the blob is detected and traced. The main idea in this stage consists in retrieving the coordinates of the traced handâs center in each frame. These coordinates will be used in order to know which computer command corresponds to which motion. Coordinates will be used differently for each gesture, depending on detected hand shape.

Five out of eleven hand shapes are used for dynamic hand gestures and rest for static hand gestures. These dynamic hand shapes are categorized into unidirectional and multi-directional hand gestures. Unidirectional hand gestures require shape and direction of motion of hand for commanding whereas multidirectional gestures require the position of hand along with its shape. Out of five dynamic hand shapes three are used for unidirectional gestures namely: swap, scroll and zoom, and remaining are used for multidirectional gestures of pointer and cursor. Each unidirectional gesture can further be used for differentiating two hand gestures depending on the direction of motion, e.g. swap can be left or right, depending on the direction of motion of hand.

Tracing involves extracting position of hand which is done by skin color detection, skin cropping, blob detection and centroid extraction. Hence tracing on the whole is a comparatively time consuming process. This process of tracing can be avoided after certain frames for unidirectional gestures as it only requires the direction of motion which can be derived from the few initial frames. Hence, the direction of unidirectional dynamic gestures can be determined by comparing centroid of initial frames.

3.4.1 Applications

A. Socially assistive robotics Patients rehabilitation can be assisted with the help of robots. This can be done by equipping proper seniors on the body of human and thereby reading the values of the results the sensors give on the screens.
B. Sign Language Recognition There are certain types of software which can transcribe symbols into text just like Googleâs or Appleâs speech recognition software work.
C. Virtual Controllers When it takes lot of time or money in finding a physical controller, gestures come in handy and they can be used as an alternative. Example of such usage could be controlling television or devices in car such as radio, air-conditioner.
D. Immersive Game technology Latest gaming consoles such as x-box and PS, a separate gaming controller is provided which recognizes all the gesture created

by the player. In this way, the gamer can control the character in the virtual world.

3.5 CONCLUSION

We propose a vision based hand gesture recognition method using transfer learning. The method was made robust by avoiding skin color segmentation, blob detection, skin area cropping and centroid extraction for unidirectional dynamic gestures. Prototype was tested successfully on seven different volunteers at different backgrounds and light conditions with an accuracy of 93.09

At last we are concluding that, building an efficient interaction between machines and human is the primary goal of GRS (gesture recognition system). There are numerous applications that have been discussed so far â Mouse control, Traffic control, Remote control, Robot control, Aid to physically challenged etc. In this paper, the idea of defining operations on specific gestures is proposed and ways to implement it using colored marker approach, instrumented glove approach and Python OS module/Open cv module. This will surely make day to day life easier if get implemented successfully. In future, we can enhance the capability for various lighting conditions and achieve more accuracy. We can add more number of gestures and allow users to surf the web using only gestures.

As seen convexity hull algorithm can implement just for finger point detection and number recognition. A similar study can be done on English alphabets as weil as Marathi alphabets by using contour analysis process. Because it contains methods for preliminary handling of the images, contours extraction, their filtrations and a Recognition. Also, it contains tools for automatie generation of templates for recognition of printing symbols. Recognition can be done by matching templates of hand by considering Contour curve shapes. In contour analysis by considering vectors value it is ease to recognize hand gestures iITespective of scaling and shape. Even by building an application which converts sign language to voice output can eliminate a baITier of communication between speech impaired people.

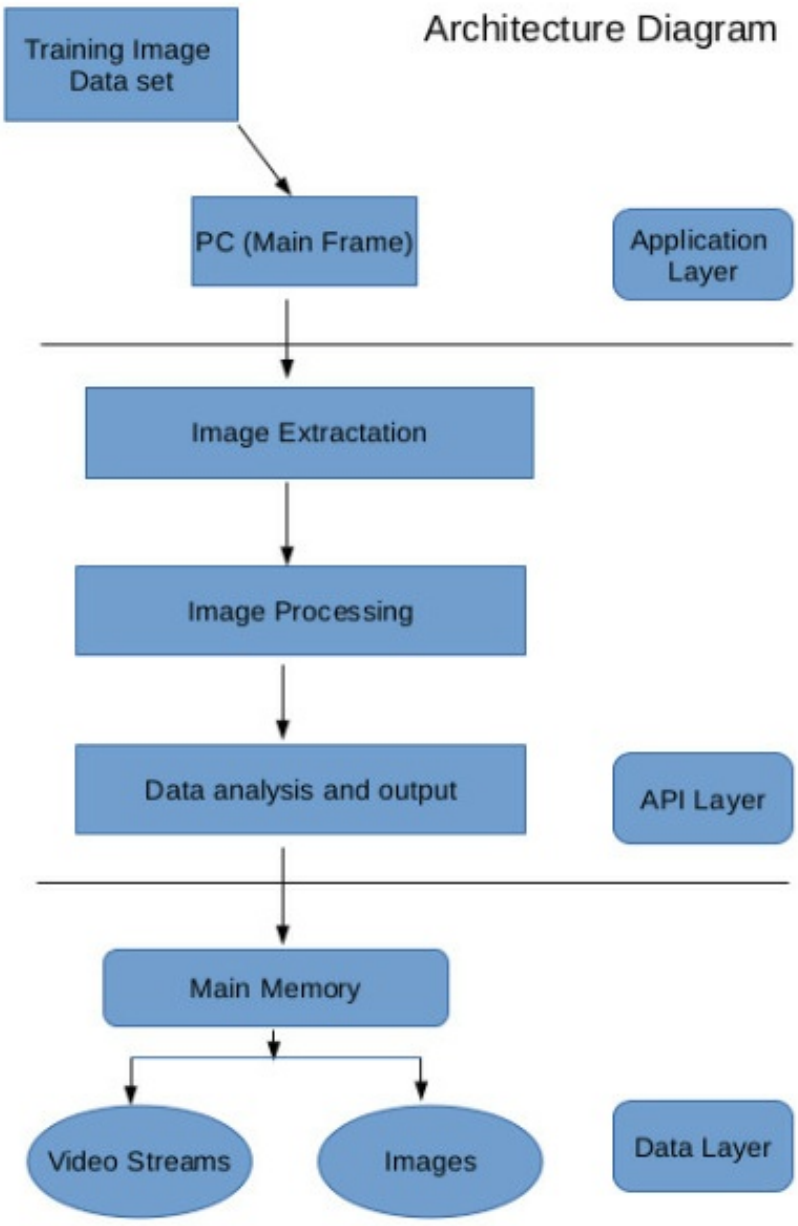
3.6 COMPARISON OF RESEARCH PAPERS

Comparison between different research paper			
Paper Name	Technology	Accuracy	Feasibility
Hand Gesture Recognition Using Deep Learning	DL	93.5%	HIGH
Gesture Recognition using Open-CV	Open-CV	85.23%	MEDIUM
A Remote Conversation Support System for Deaf- mute Persons Based on Bimanual Gestures Recognition Using Finger-worn Devices	DL	91.72%	MEDIUM
Sign Language Recognition Using Image Based Hand Gesture Recognition Techniques	DL	96.5%	MEDIUM
Real-time Object Detection and Tracking in an Unknown Environment	DL	86.89%	HIGH
Object Detection Algorithms for Video	Open-CV	92.00%	HIGH
Gesture Recognition Method Based On Deep Learning	DL	97.36%	HIGH

3.7 References

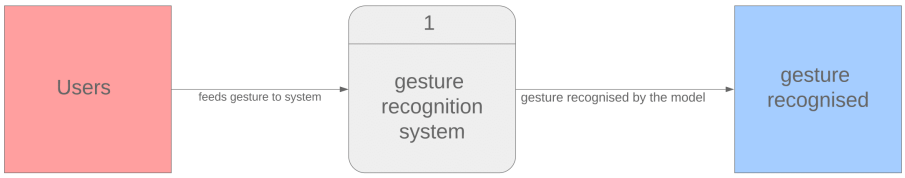
1. Hand Gesture Recognition Using Deep Learning -Soeb Hussain and Rupal Saxena Xie Han, Jameel Ahmed Khan, Prof. Hyunchul Shin.
2. Gesture Recognition using Open-CV - Mohd. Baqir Khan, Kavya Mishra, Mohammed Abdul Qadeer.
3. A Remote Conversation Support System for Deaf- mute Persons Based on Bimanual Gestures Recognition Using Finger-worn Devices - Kengo Kuroki, Yiming Zhou, Zixue Cheng, Zixian Lu, Yinghui Zhou, Lei Jing
4. Sign Language Recognition Using Image Based Hand Gesture Recognition Techniques - Ashish S. Nikam , Aarti G. Ambekar
5. Real-time Object Detection and Tracking in an Unknown Environment - Shashank Prasad, Shubhra Sinha 6. Object Detection Algorithms for Video Surveillance Applications - Apoorva Raghunandan, Mohana, Pakala Raghav and H. V. Ravish Aradhya
7. Head Gesture Recognition System Using Gesture Cam - Rushikesh T. Bankar , Dr. Suresh S. Salankar
8. Gesture Recognition Method Based On Deep Learning - Tong Du, Xuemei Ren, Huichao Li

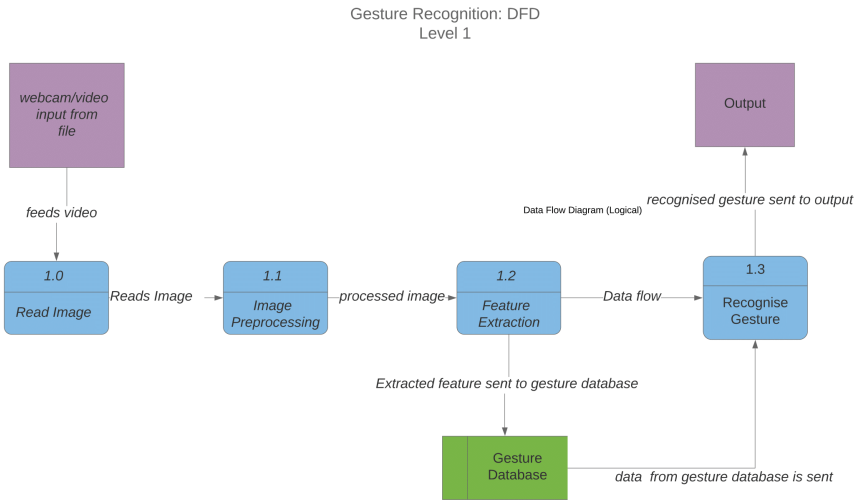
4 Architecture Diagram


Architecture Diagram
Training Image Data set
PC (Main Frame)
Application Layer
Image Extractation
Image Processing
Data analysis and output
API Layer
Main Memory
Video Streams
Images
Data Layer

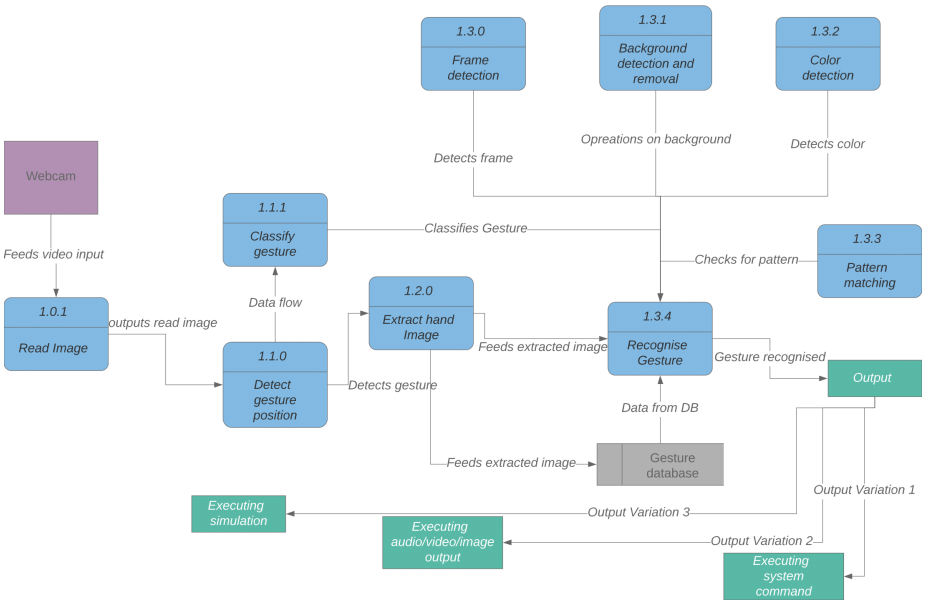
5 Data Flow Diagram

Gesture Recognition:
DFD Level 0



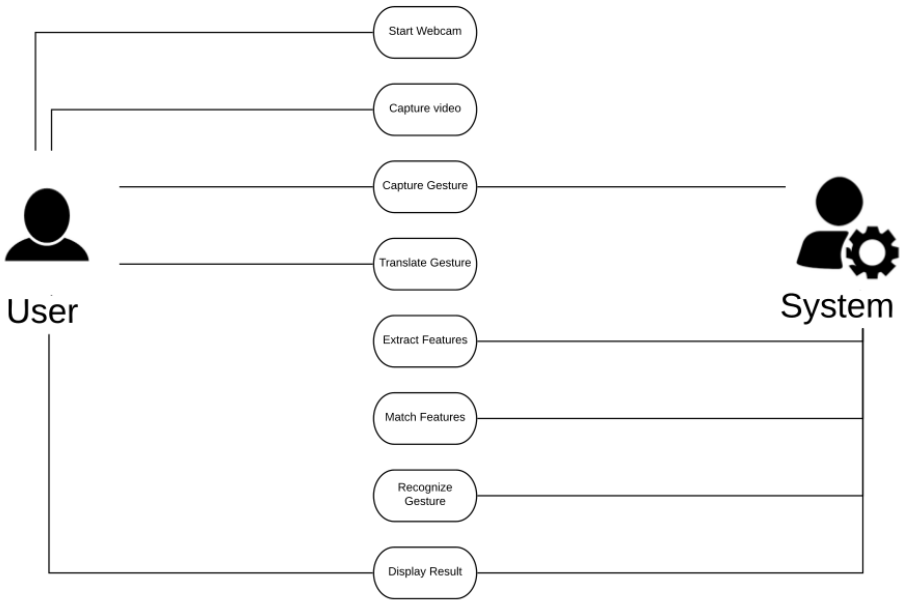

Gesture Recognition: DFD
Level 1
webcam/video input from file
feeds video
1.0
Read Image
Reads Image
1.1
Image Preprocessing
processed image
1.2
Feature Extraction
Data flow
1.3
Recognise Gesture
Output
recognised gesture sent to output
Data Flow Diagram (Logical)
Extracted feature sent to gesture database
Gesture Database
data from gesture database is sent

Gesture Recognition:
DFD Level 2

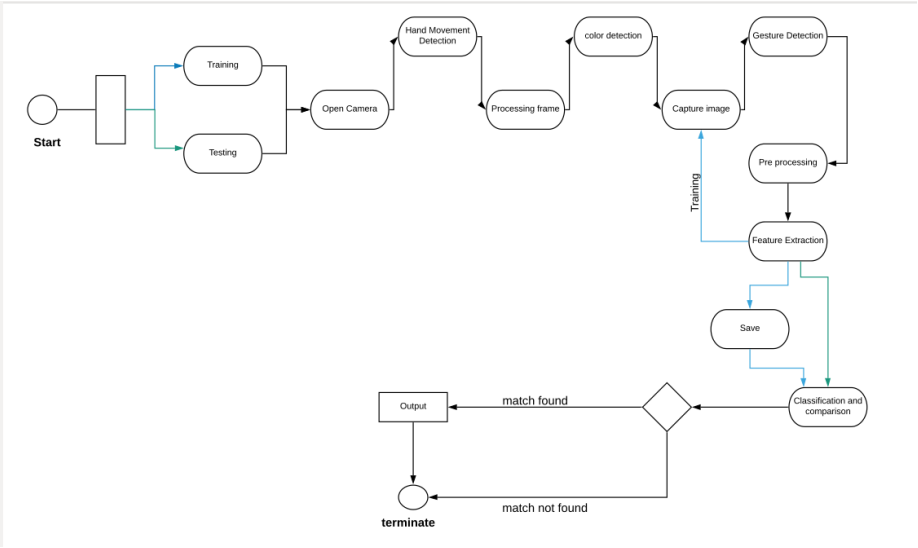


6 UML Diagrams

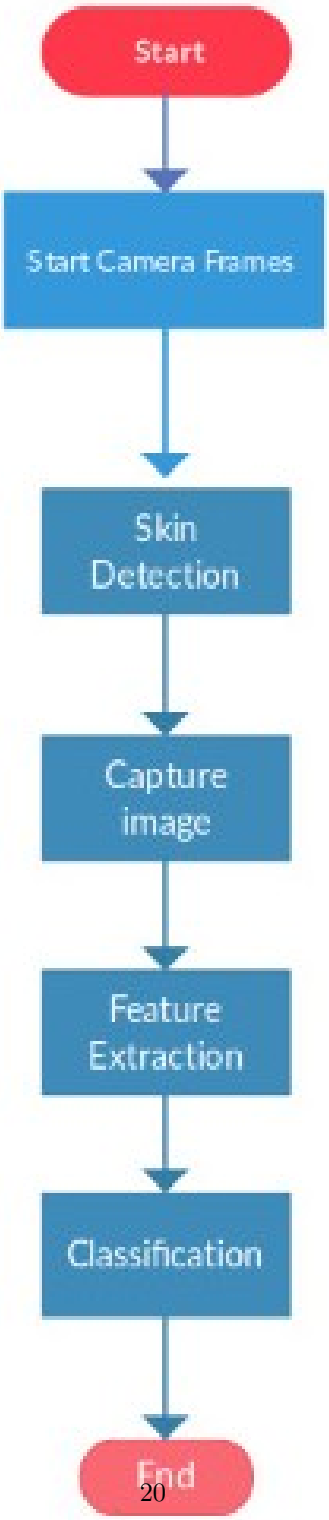
6.1 UML Diagram



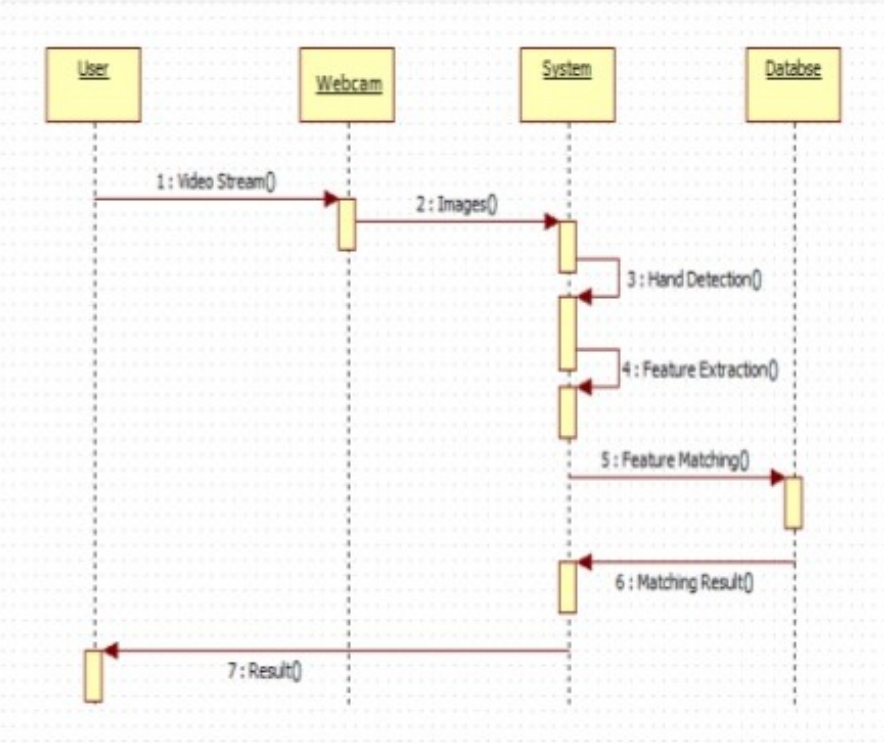
6.2 Activity Diagram



6.3 State Diagram

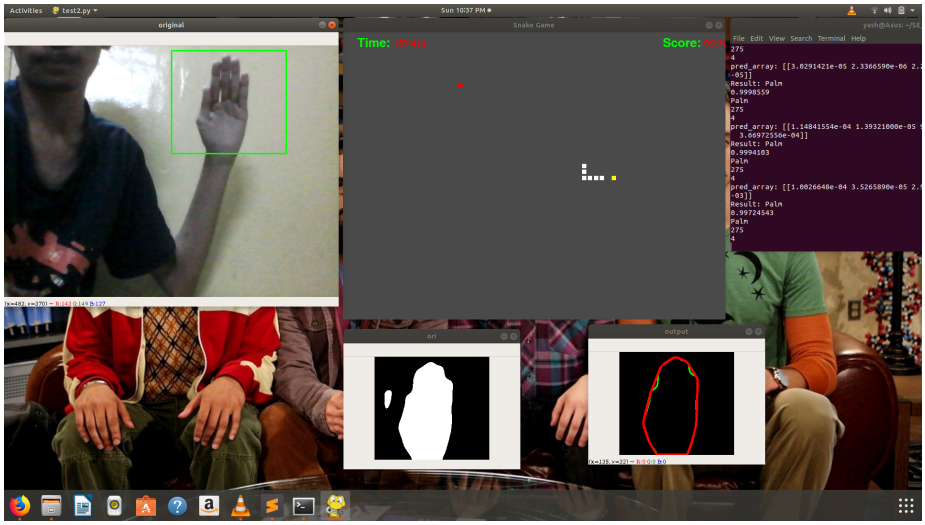

Start
Start Camera Frames
Skin Detection
Capture image
Feature Extraction
Classification
End

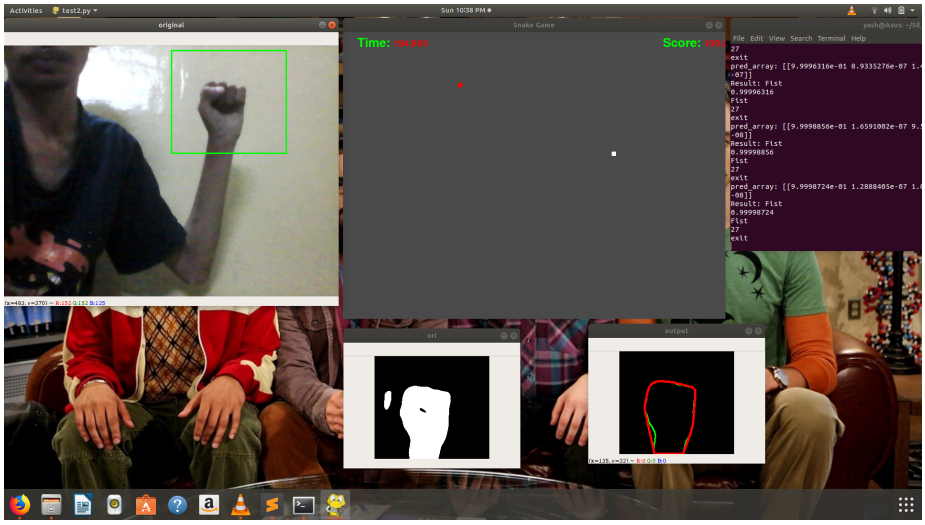
6.4 Sequence Diagram

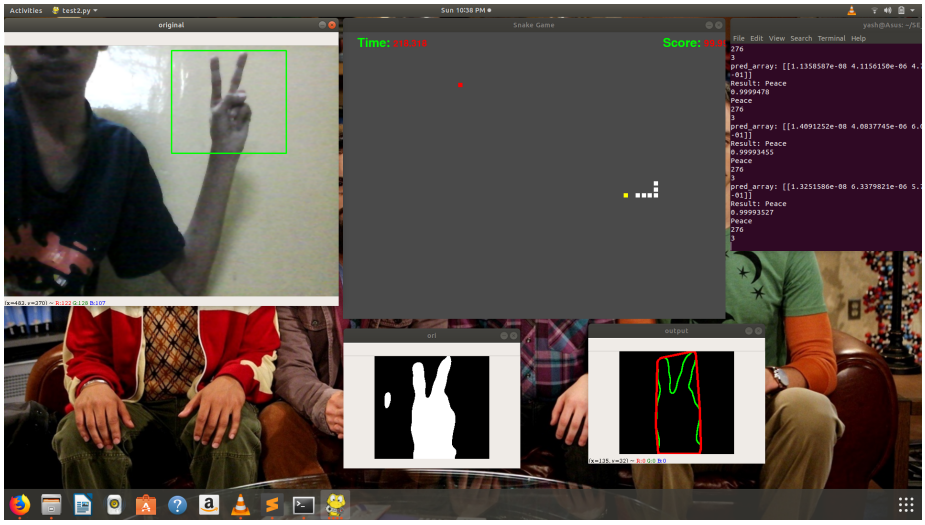


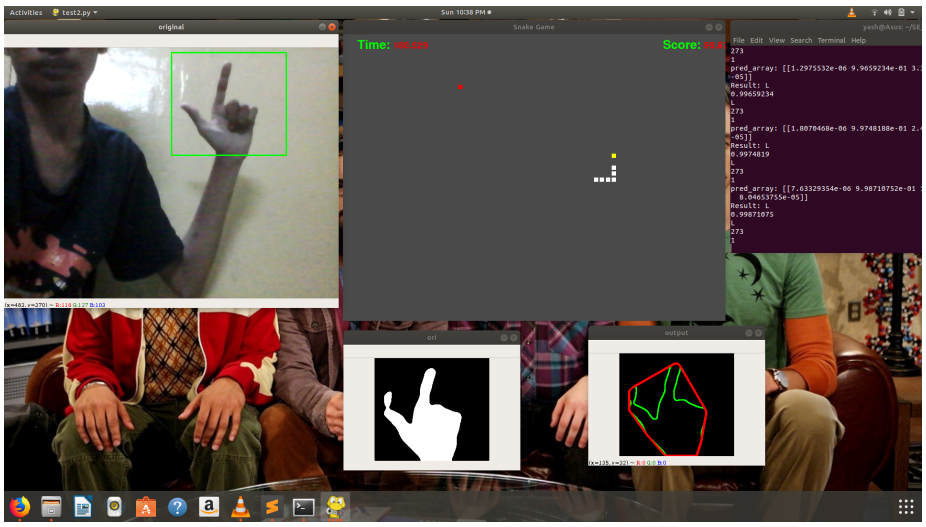
7 Implementation

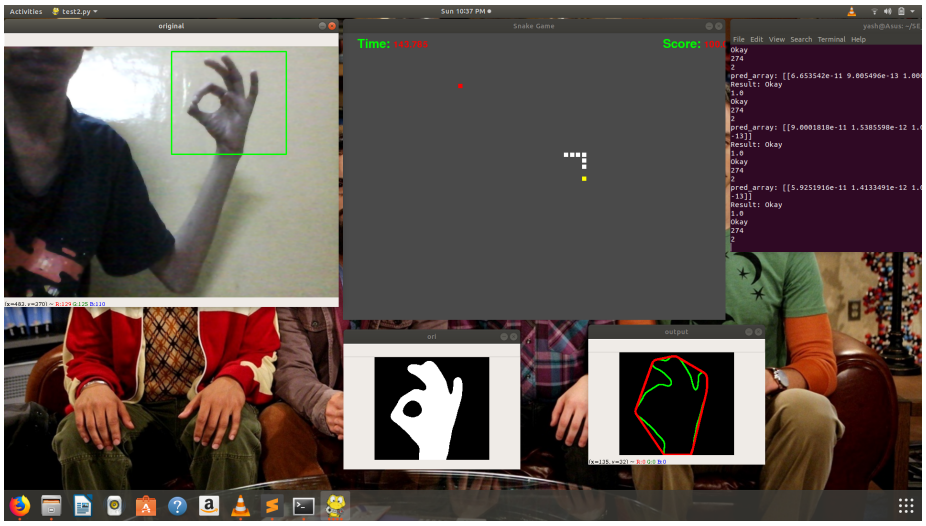
In our project, we can play the snake game using different hand gestures. We assigned game controls to different gestures:
1.Palm - Right
2.Fist - Exit
3.Peace - Left
4.L - Up
5.Ok - Down




original
Snake Game
Time:
Score:
ori
output


original
Snake Game
Time:
Score:
ori
output


Activities
test2.py
original
Snake Game
Time:
Score:
ori
output


Activities
test2.py
original
Snake Game
Time:
Score:
ori
output

8 Project Estimation

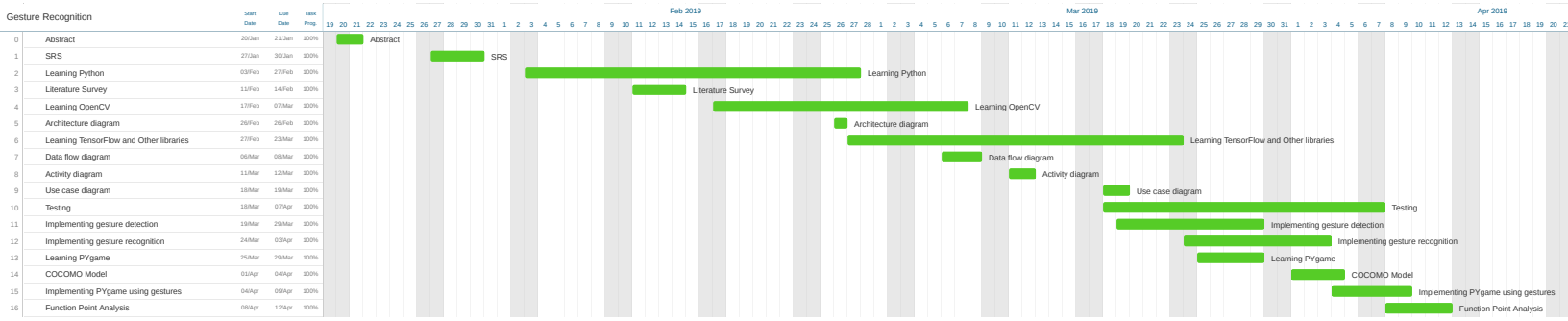
8.1 Lines of code

Number of lines of code: 450

8.2 Cocomo Model

Efforts:- 3 Persons 8 Months of Work
Project Type:
Organic-
A software project is said to be an organic type if the team size required is adequately small, the problem is well understood and has been solved in the past and also the team members have a nominal experience regarding the problem.
Basic cocomo model a = 2.4, b = 1.05
E = a(KLOC)^b
E = 1465.8326167

9 Project Scheduling Executed

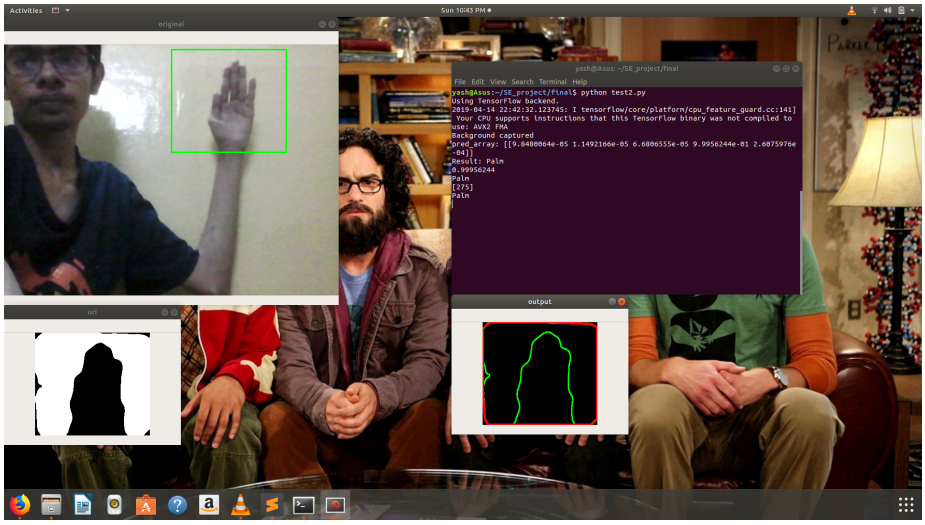

Gesture Recognition
Start Date
Due Date
Task Prog.
Feb 2019
Mar 2019
Apr 2019
0 Abstract 20/Jan 21/Jan 100%
1 SRS 27/Jan 30/Jan 100%
2 Learning Python 03/Feb 27/Feb 100%
3 Literature Survey 11/Feb 14/Feb 100%
4 Learning OpenCV 17/Feb 07/Mar 100%
5 Architecture diagram 26/Feb 26/Feb 100%
6 Learning TensorFlow and Other libraries 27/Feb 23/Mar 100%
7 Data flow diagram 06/Mar 08/Mar 100%
8 Activity diagram 11/Mar 12/Mar 100%
9 Use case diagram 18/Mar 19/Mar 100%
10 Testing 18/Mar 07/Apr 100%
11 Implementing gesture detection 19/Mar 29/Mar 100%
12 Implementing gesture recognition 24/Mar 03/Apr 100%
13 Learning PYgame 25/Mar 29/Mar 100%
14 COCOMO Model 01/Apr 04/Apr 100%
15 Implementing PYgame using gestures 04/Apr 09/Apr 100%
16 Function Point Analysis 08/Apr 12/Apr 100%

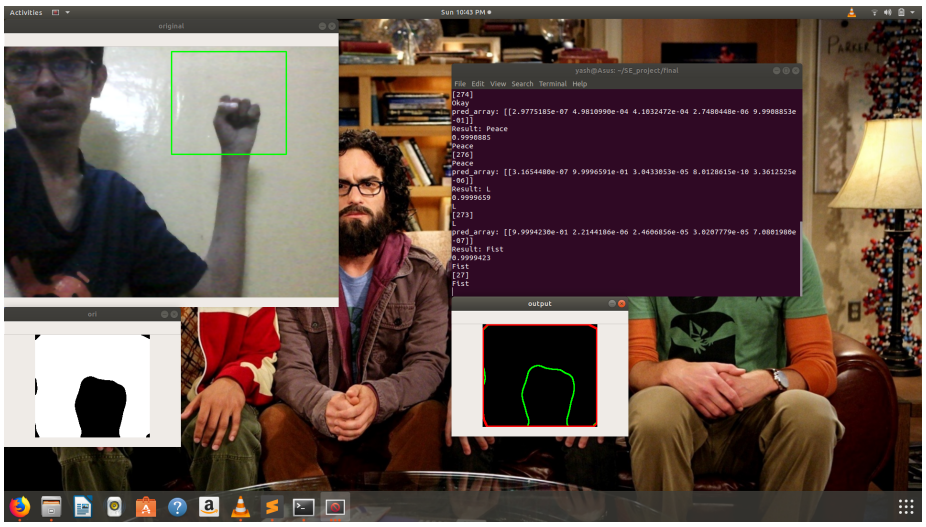
10 Testing

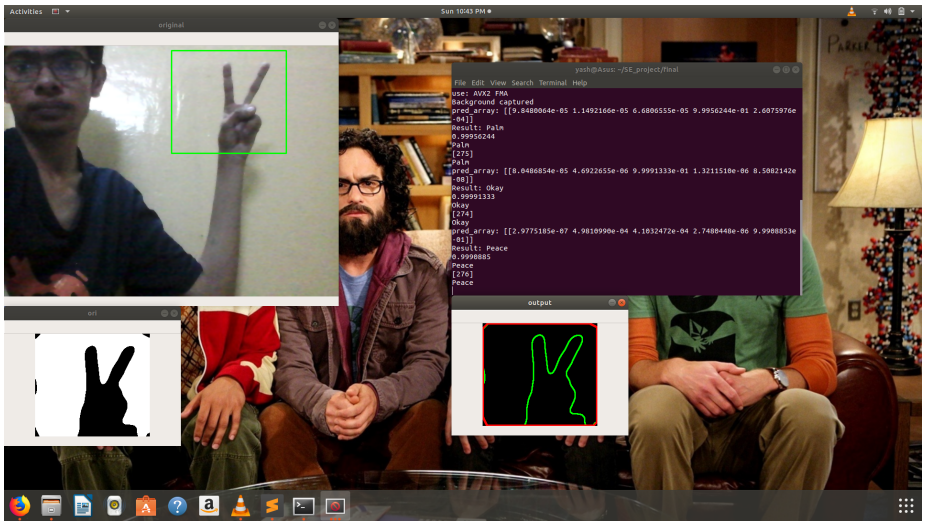
10.1 Unit Test Cases

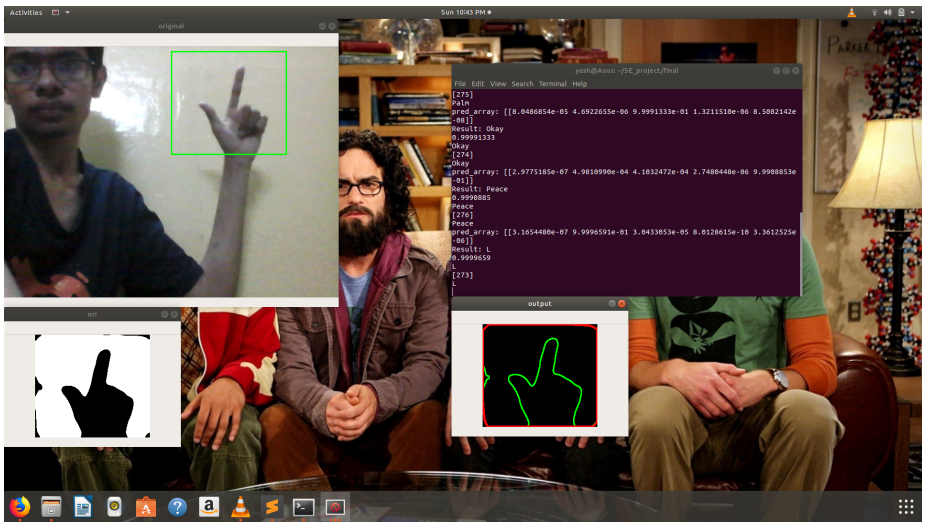
10.1.1 Individual Recognition Of different hand gestures

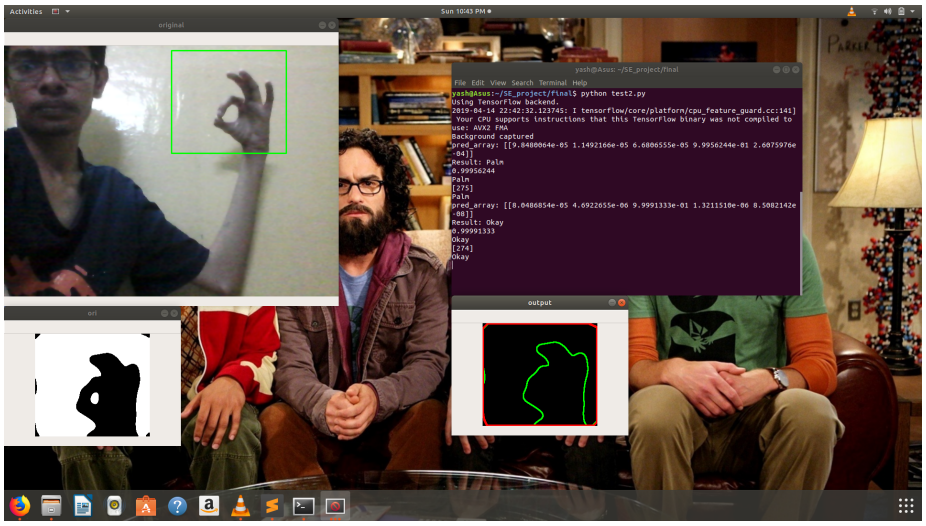
Our model recognises different hand gestures such as:
1.Palm
2.Fist
3.Peace
4.L
5.Ok




Activities
Sun 10:43 PM
original
yash@Asus: ~/SE_project/final
File Edit View Search Terminal Help
[274]
Okay
pred_array: [[2.9775185e-07 4.9810990e-04 4.1032472e-04 2.7480448e-06 9.9988853e
-01]]
Result: Peace
0.9998885
Peace
[276]
Peace
pred_array: [[3.1654480e-07 9.9996591e-01 3.8433853e-05 8.0128615e-10 3.3612525e
-06]]
Result: L
0.9999659
L
[273]
L
pred_array: [[9.9994230e-01 2.2144186e-06 2.4606856e-05 3.0207779e-05 7.0801980e
-07]]
Result: Fist
0.9999423
Fist
[27]
Fist
output

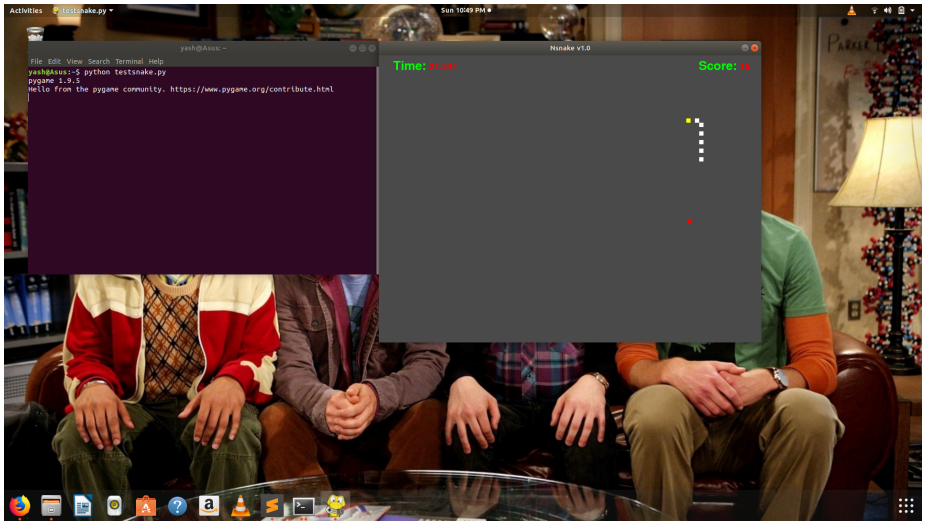

Activities
Sun 10:43 PM
original
yash@Asus: ~/SE_project/final
File Edit View Search Terminal Help
use: AVX2 FMA
Background captured
pred_array: [[9.8480064e-05 1.1492166e-05 6.6806555e-05 9.9956244e-01 2.6075976e
-04]]
Result: Palm
0.99956244
Palm
[275]
Palm
pred_array: [[8.0486854e-05 4.6922655e-06 9.9991333e-01 1.3211510e-06 8.5082142e
-08]]
Result: Okay
0.99991333
Okay
[274]
Okay
pred_array: [[2.9775185e-07 4.9810990e-04 4.1032472e-04 2.7480448e-06 9.9988853e
-01]]
Result: Peace
0.9998885
Peace
[276]
Peace
output


Activities
Sun 10:43 PM
original
yash@Asus: ~/SE_project/final
File Edit View Search Terminal Help
[275]
Paln
pred_array: [[8.0486854e-05 4.6922655e-06 9.9991333e-01 1.3211510e-06 8.5082142e
-08]]
Result: Okay
0.99991333
Okay
[274]
Okay
pred_array: [[2.9775185e-07 4.9810990e-04 4.1032472e-04 2.7488448e-06 9.9988853e
-01]]
Result: Peace
0.9998885
Peace
[276]
Peace
pred_array: [[3.1654480e-07 9.9996591e-01 3.8433053e-05 8.0128615e-10 3.3612525e
-06]]
Result: L
0.99996591
L
[273]
L
output


Activities
Sun 10:43 PM
original
yash@Asus: ~/SE_project/final
File Edit View Search Terminal Help
yash@Asus:~/SE_project/final$ python test2.py
Using Tensorflow backend.
2019-04-14 22:42:32.123745: I tensorflow/core/platform/cpu_feature_guard.cc:141]
Your CPU supports instructions that this TensorFlow binary was not compiled to
use: AVX2 FMA
Background captured
pred_array: [[9.8480064e-05 1.1492166e-05 6.6806555e-05 9.9956244e-01 2.6075976e
-04]]
Result: Paln
0.99956244
Paln
[275]
Paln
pred_array: [[8.0486854e-05 4.6922655e-06 9.9991333e-01 1.3211510e-06 8.5082142e
-08]]
Result: Okay
0.99991333
Okay
[274]
Okay
output

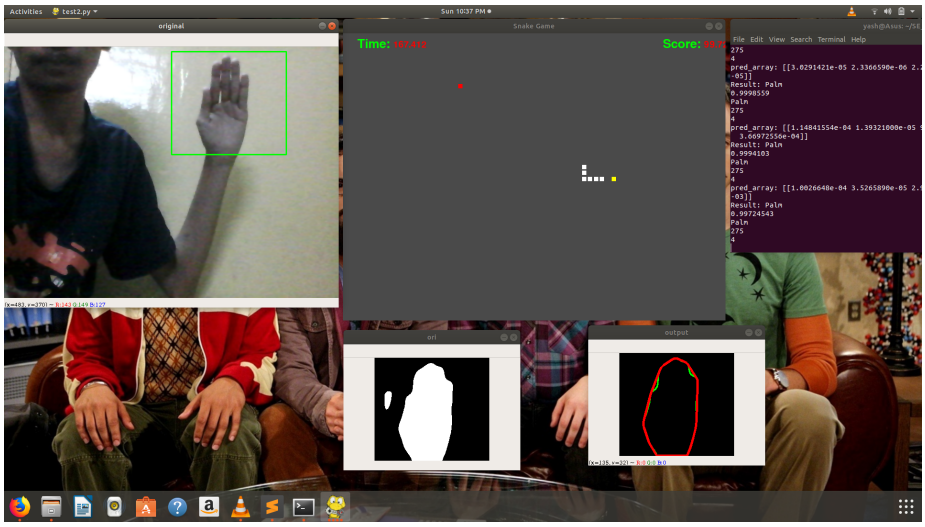
10.1.2 Snake Game

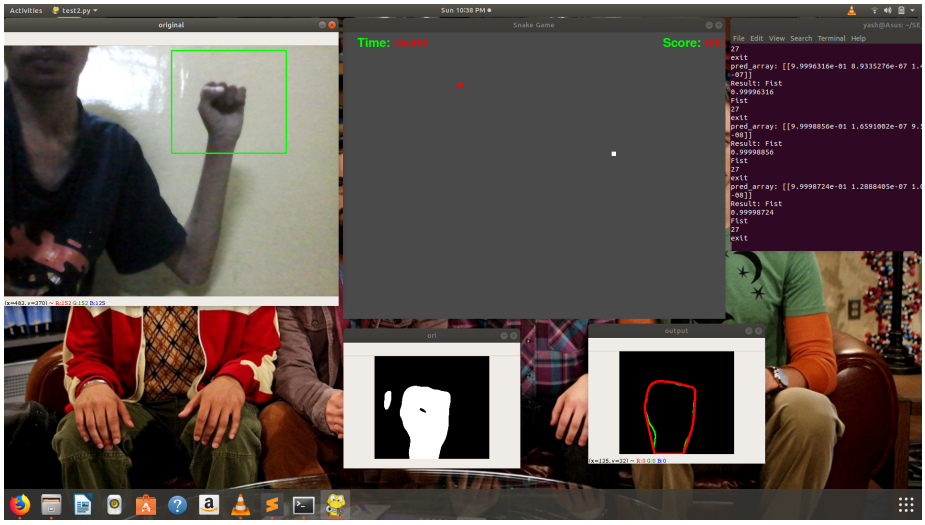
Game can be played using arrow keys left,right,up,down.

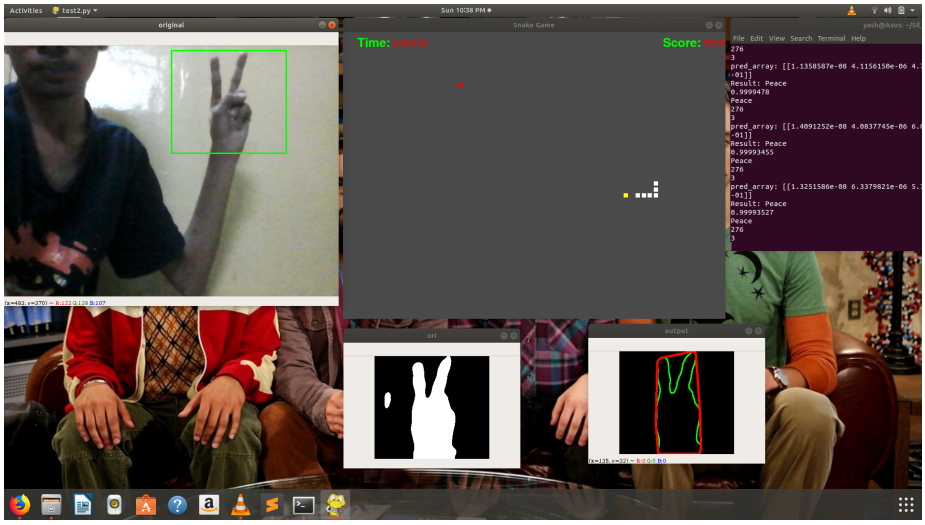


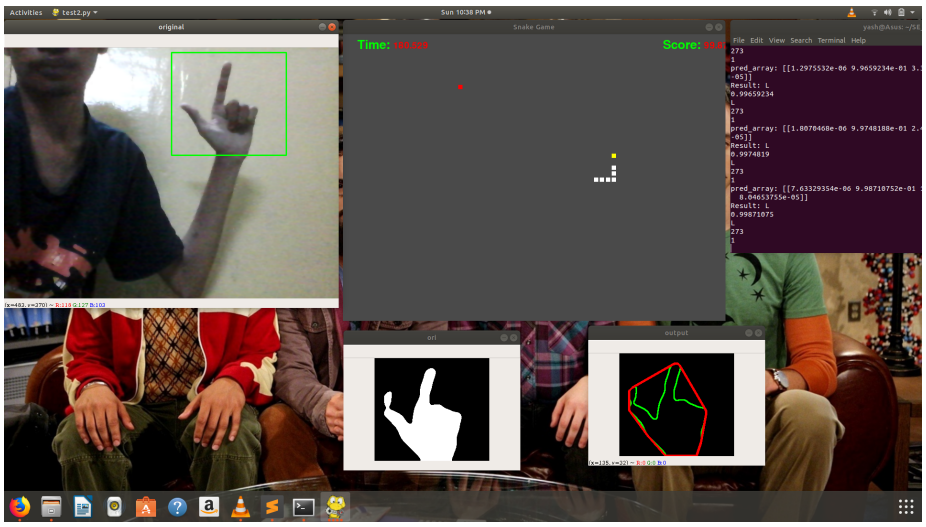
10.2 Integration Test

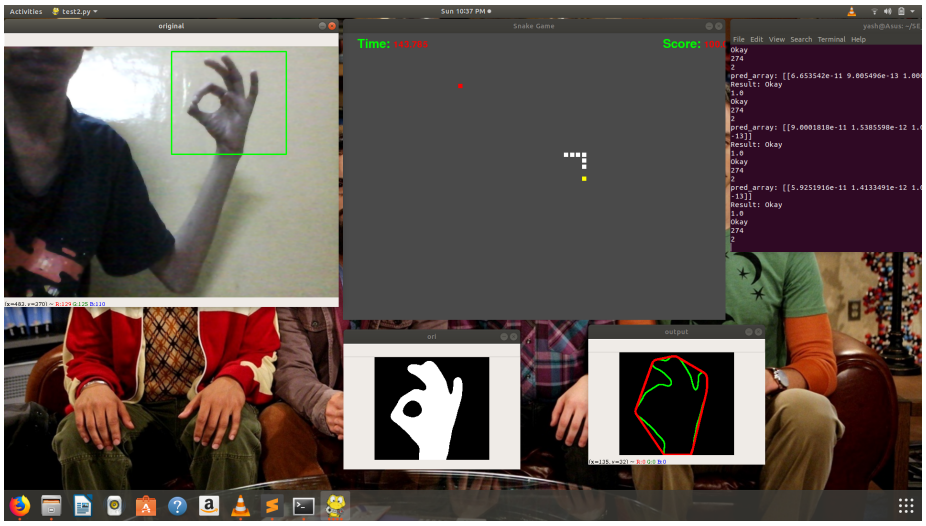
Finally, we integrated both models, recognition of gestures and snake gake. So that we can play the game using different hand gestures. We assigned game controls to different gestures:
1.Palm - Right
2.Fist - Exit
3.Peace - Left
4.L - Up
5.Ok - Down




Activities
test2.py
original
Snake Game
Time:
Score:
File Edit View Search Terminal Help
ori
output


Activities
test2.py
original
Snake Game
Time:
Score:
File Edit View Search Terminal Help
ori
output


original
Snake Game
Time:
Score:
ori
output


original
Snake Game
Time:
Score:
ori
output

11 References

1.Research papers of IV – Labs VNIT.

2. Video Reference https://youtu.be/TwYUm7i61JM

3. IEEE Research Papers

https://ieeexplore.ieee.org/document/6179213/
https://ieeexplore.ieee.org/document/7176405/
4. Hand Gesture Recognition Using Deep Learning -Soeb Hussain and Rupal Saxena Xie Han, Jameel Ahmed Khan, Prof. Hyunchul Shin.
5. Gesture Recognition using Open-CV - Mohd. Baqir Khan, Kavya Mishra, Mohammed Abdul Qadeer.
6. A Remote Conversation Support System for Deaf- mute Persons Based on Bimanual Gestures Recognition Using Finger-worn Devices - Kengo Kuroki, Yiming Zhou, Zixue Cheng, Zixian Lu, Yinghui Zhou, Lei Jing
7. Sign Language Recognition Using Image Based Hand Gesture Recognition Techniques - Ashish S. Nikam , Aarti G. Ambekar
8. Real-time Object Detection and Tracking in an Unknown Environment -Shashank Prasad, Shubhra Sinha 9. Object Detection Algorithms for Video
Surveillance Applications - Apoorva Raghunandan, Mohana, Pakala Raghav and H. V. Ravish Aradhya
01. Head Gesture Recognition System Using Gesture Cam - Rushikesh T. Bankar , Dr. Suresh S. Salankar
11. Gesture Recognition Method Based On Deep Learning - Tong Du, Xuemei Ren, Huichao Li