Project Report

Machine Learning

Anukarsh Pratap
Dr. Aruna Tiwari
29 November 2023

Abstract:

In today's world, everyone opts for fast interaction with complex systems that ensure a quick response. Here is where human-computer interaction comes into play. This interaction is unrestricted and challenges the used devices such as the keyboard and mouse for input. Gesture recognition has been gaining much attention. Gestures are instinctive and are frequently used in day-to-day interactions. Therefore, communicating using gestures with computers creates a whole new standard of interaction. In this project, with the help of computer vision and deep learning techniques, user hand movements (gestures) are used in real-time to control the media player. In this project, seven gestures are defined to control the media players using hand gestures. The proposed web application enables the user to use their local device camera to identify their gesture and execute the control over the media player and similar applications. It increases efficiency and makes interaction effortless by letting the user control his/her laptop/desktop from a distance.

Introduction:

Gesture recognition is an active research field that is a kind of non-cognitive computing interface for users that allows devices to catch and interpret human gestures as commands. The usage of gestures comes naturally to all. The study reveals that blind people also use gestures while speaking with others, as gestures easily represent ideas and actions. This motivates the idea of integrating gestures to interact with computer devices and carry out the tasks easily . It has many applications including virtual reality environment control, sign language translation, robot remote control, musical creation, school kids interaction, drowsiness detection of drivers, and also activity monitoring of elderly or disabled people. In recent years, the HCI method has been evolving rapidly. Traditional input devices such as a mouse, keyboard, and joystick are being replaced by these methods by simplifying the interaction with interfaces.

Technology has been using gestures to control various areas of work in many fields, including entertainment, automation, health and medical, electronics, and academics. In healthcare, gestures can provide the sterility needed to prevent any infection from spreading, by using them to interact with medical equipment, control visualisations, and distribute resources too. The entertainment field has been a very propitious and rewarding arena for innovations. End users are always keen to try new paradigms of interfaces.

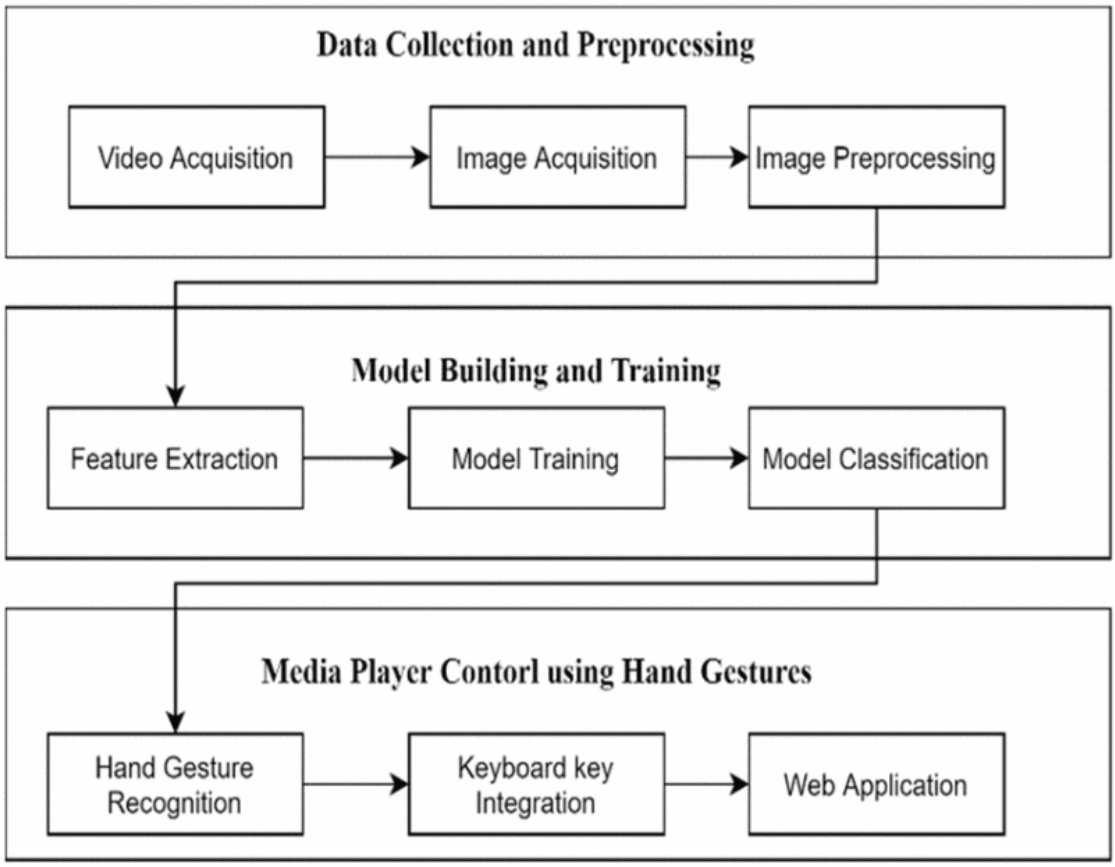
Proposed Methods :

The proposed gesture detection system for media player control is broadly divided into two stages. The first stage performs the detection of hand gestures and the keyboard controls are integrated with each gesture in the second stage. The gesture recognition is implemented using computer vision and deep learning techniques with a custom-built dataset. A new dataset is created that contains 7 gestures

The CNN model is trained by feeding the image frames to three convolution layers with a ReLU activation function. Pooling layers are added to perform max pooling which is then flattened and added to dense layers

In the proposed system, the gestures are predicted by the CNN model. The preprocessing and re-sizing of the image are performed before passing it into the model. The trained CNN model is used to predict one of the seven gestures. Each gesture is connected to an individual keyboard control which will further control any media player playing on the system whenever a gesture is detected.

Proposed Workflow :



Caption

I) Image Acquisition and Pre-processing

When the user performs hand gestures in front of the webcam. The image frames are collected from the live video using the OpenCV. These images are converted into black and white images as shown in Fig. 5, to improve accuracy in predicting gestures and then stored in respective directories.

In this project, gestures are collected from 3 different people. The dataset contains 150 images for each gesture. A directory structure is created to store images. Two modes, train, and test are provided for user convenience. The user is given an option to choose either one of the modes. The images collected in train mode are used to train the model and the test mode images are used to test the model accuracy. The images are stored in a specific directory based on the user input.

While the camera is on, two frames are displayed on the screen and the user can capture images frame- by-frame using the read function and the mirror image is simulated. The user has to place the hand in the Region Of Interest (ROI) i.e., the bounding box, and perform the gestures. The frames are extracted from ROI and resized to 120x120x1. The count of the number of images in each directory is printed onto the screen.

II) Feature Extraction

Import Keras models and hidden layers required to build convolutional networks. The CNN model is built using a hidden input layer followed by two convolution layers. An activation operation called ReLU and a pooling layer called MaxPooling are added after every convolution layer. A flattening layer is added along with two fully connected layers, one with RELU activation and another with softmax activation which is used for classification. Fig. 6 shows the architecture of the CNN model used for feature extraction and classification of the gestures.

Lastly, the compilation of the CNN model is performed by implementing adam algorithm as an optimizer, categorical_crossentropy as a loss function method to find error and accuracy as the performance metrics to evaluate the model.

III) Train the model

Next, ImageDataGenerator class is used to generate batches of images to train and validate the model. The fit function is used to train the model with a fixed number of epochs. After training, the trained model is saved in JSON format and the weights are saved directly from the model using the save_weights function.

IV) Media control using predicted Hand gestures

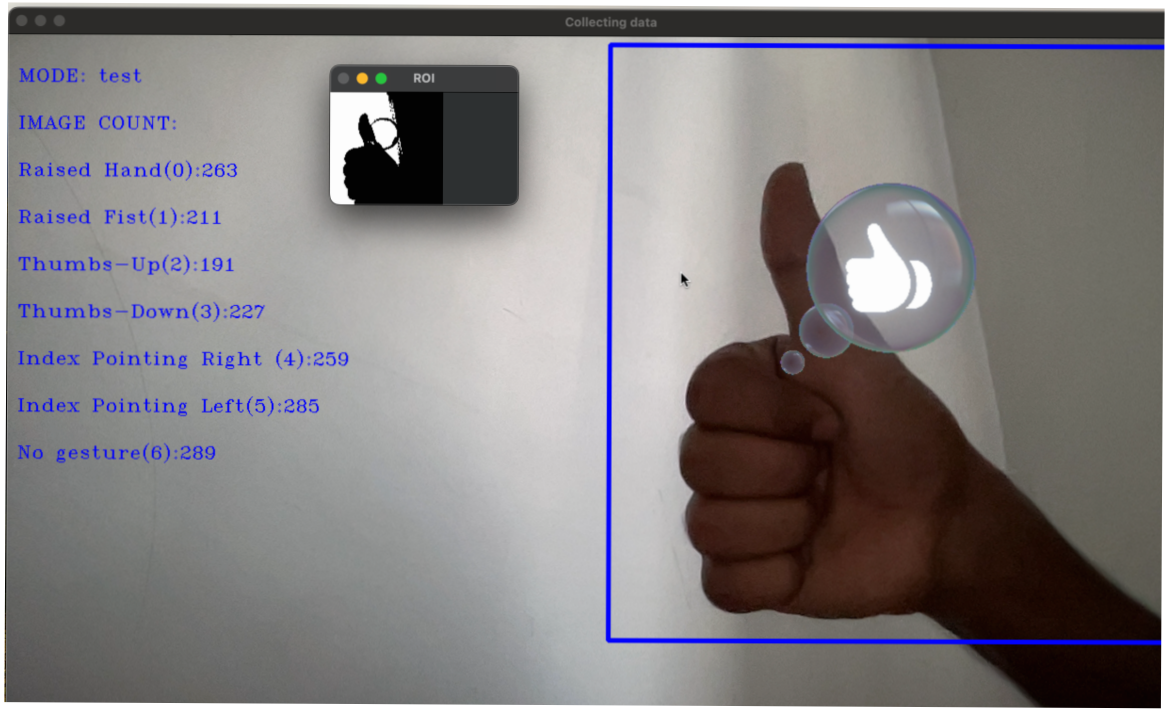
The trained model in JSON format and the model weights are loaded to predict the hand gestures. The PyAutoGUI which is used for Keyboard key integration with hand gestures and Streamlit which is used to create a user interface are also imported.

Three web pages are created using the streamlit web framework. The web pages are designed using the in- built streamlit functions and HTML templates. The first page is an about page with a brief introduction about the project. The second page contains the video demo of the project. The third page is the demo page which is used to predict the hand gestures to control the media player.

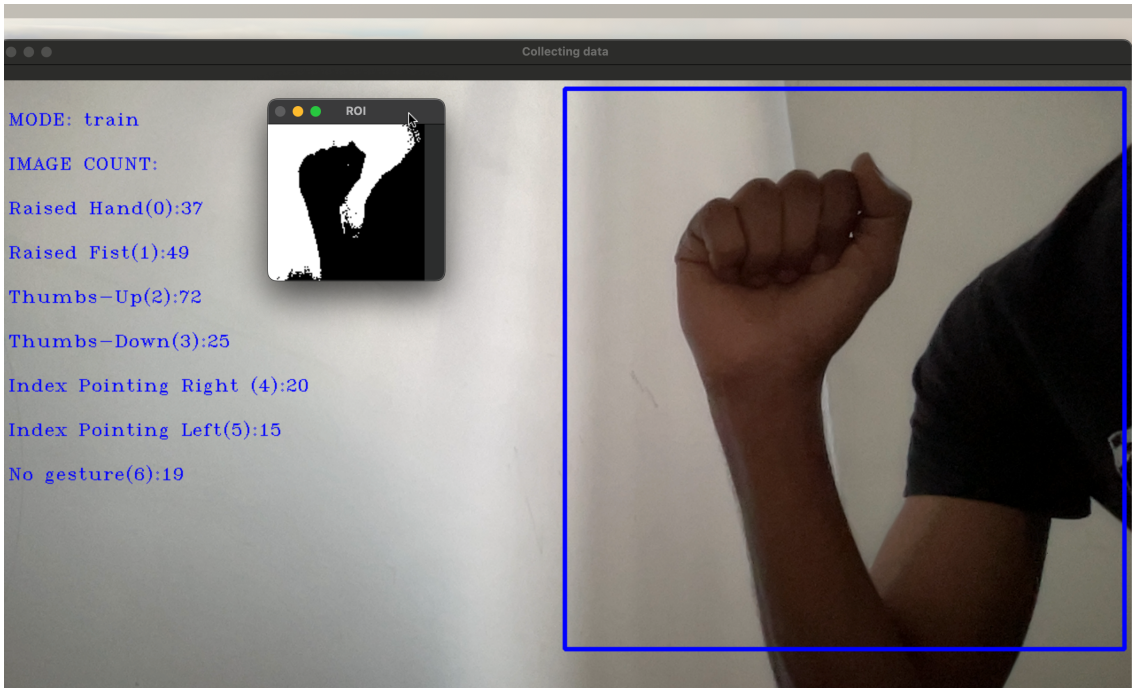
When the user clicks the start button on the web page, the web camera starts running. The user can perform the hand gestures in the Region Of Interest and the trained CNN model will predict the gesture. Each hand gesture is integrated with a Keyboard key using PyAutoGUI with help of conditional if-else statements to call predicted gestures. Each gesture is mapped with a keyboard key control and a label. The number of presses is assigned as 1, so every time a gesture is predicted the integrated control function is

performed once. The user can exit the system by pressing the escape keyboard key. The video frame will display the gesture predicted and the action being performed whenever the user is using the system to control the media player.

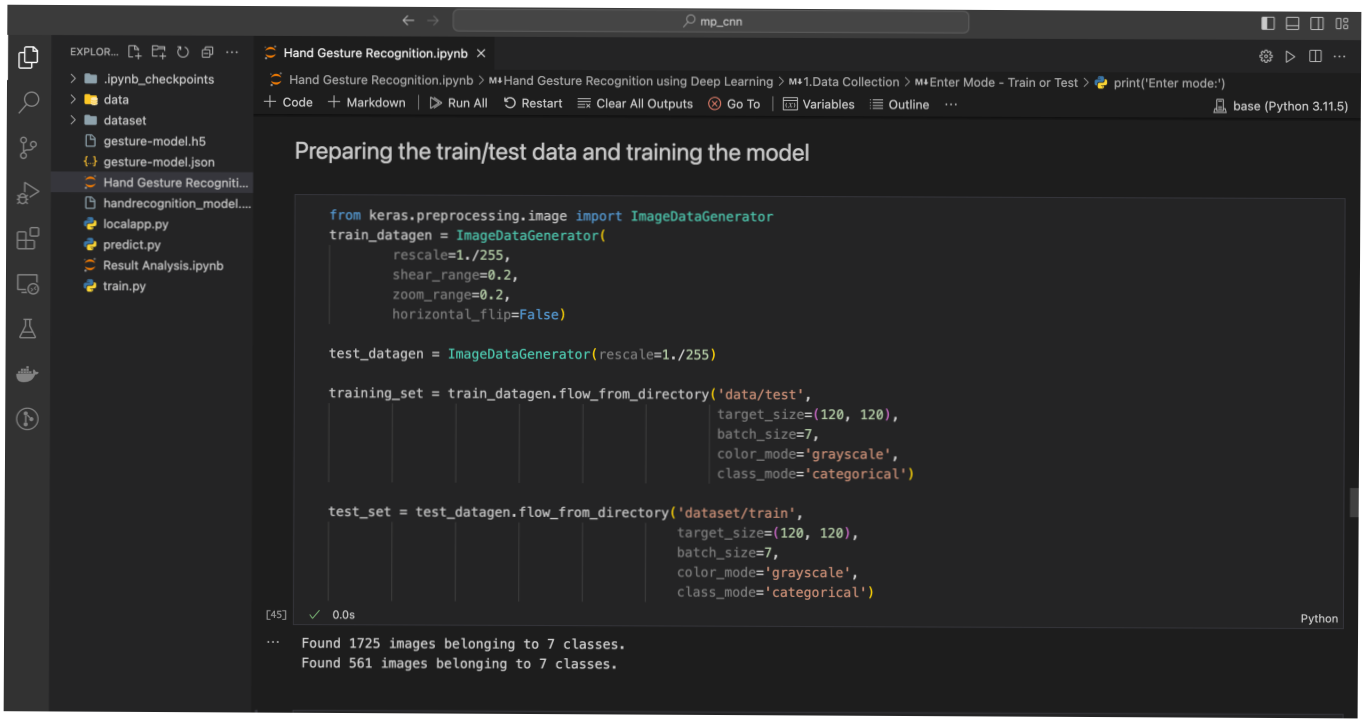
Output:



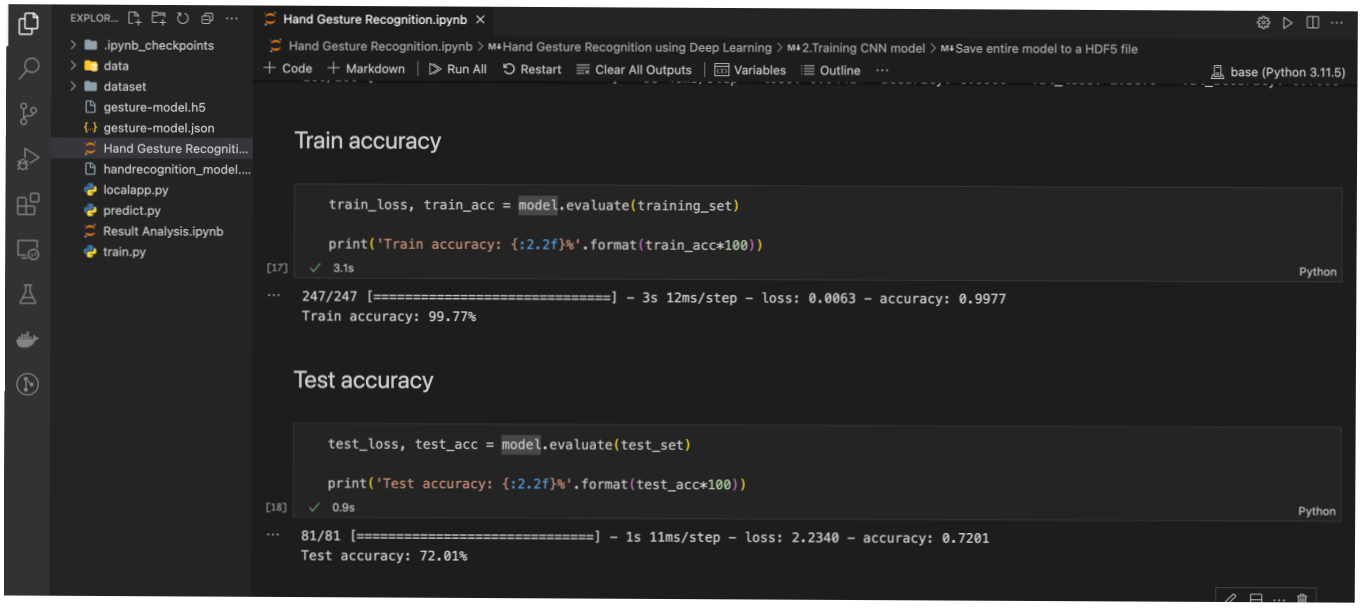
Testing mode



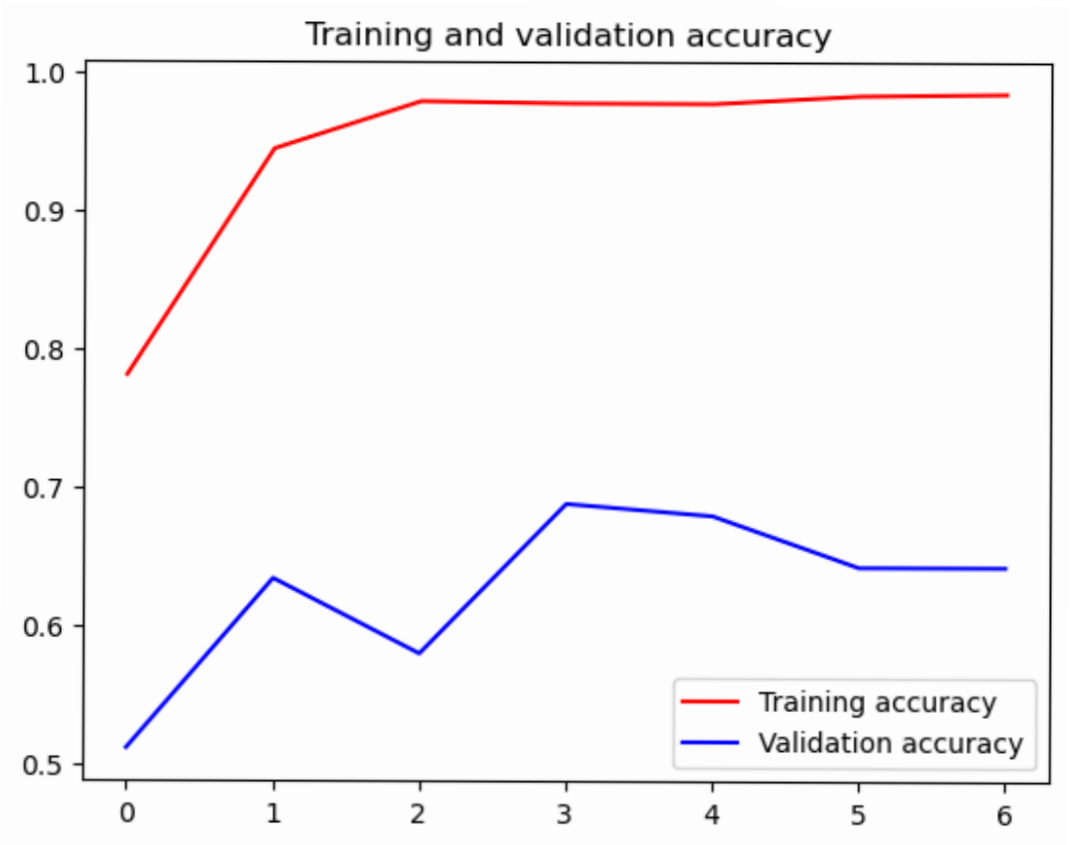
Training mode

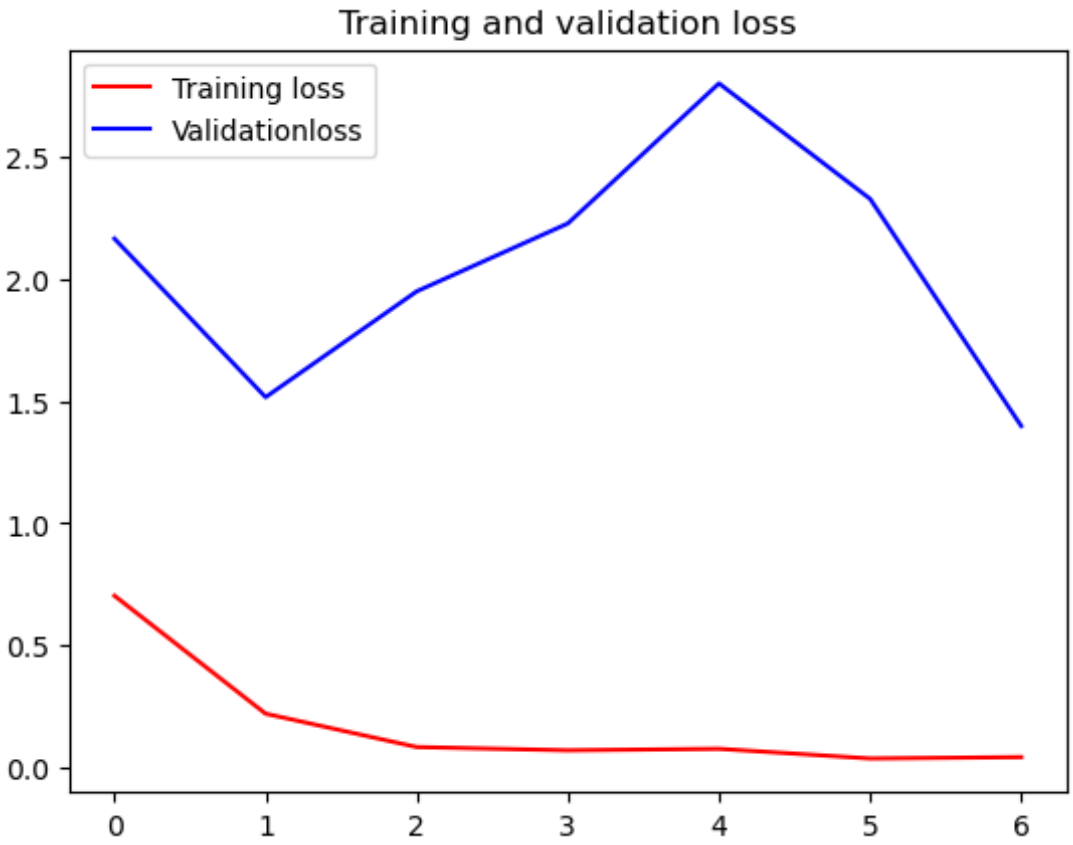


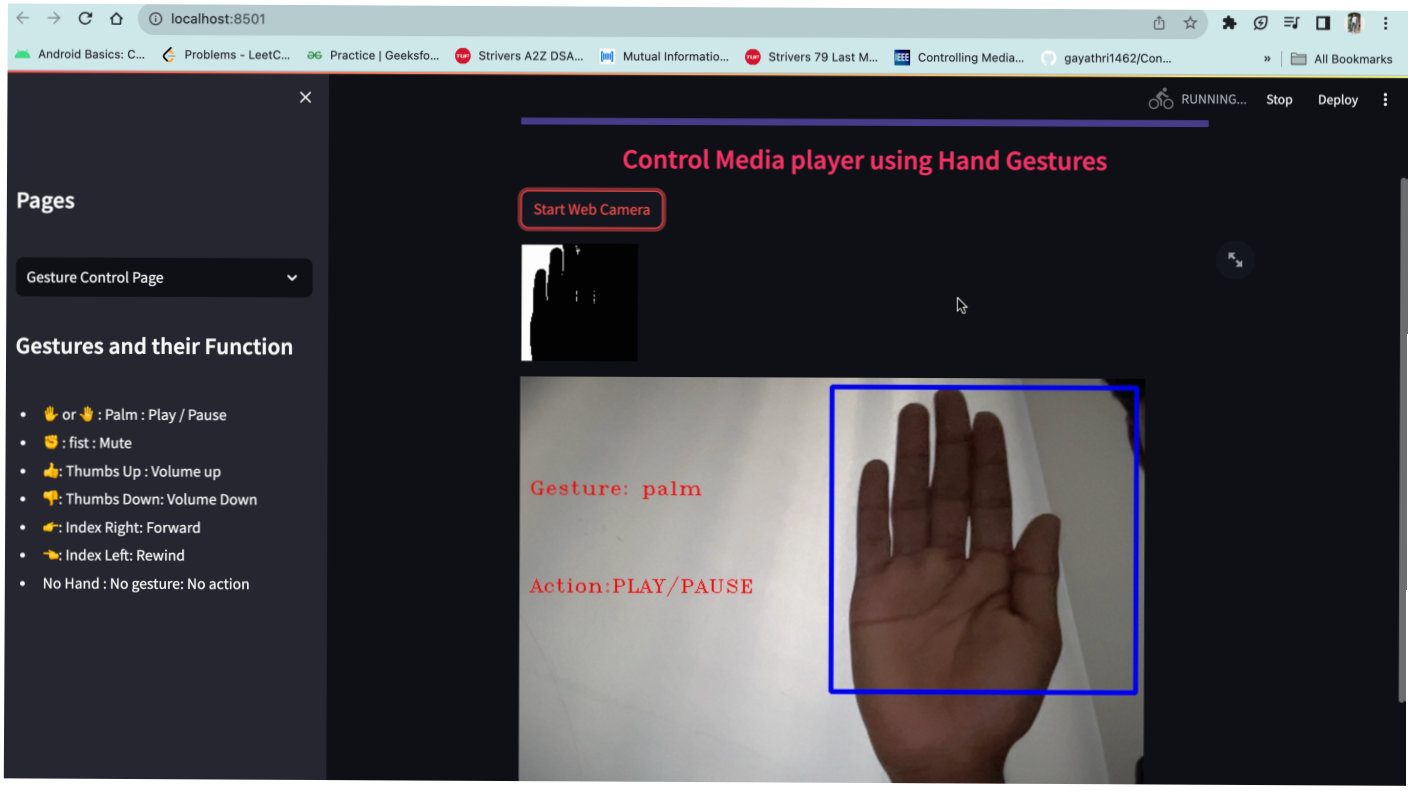
Training/Testing Model



Accuracy


Training and validation accuracy
1.0
0.9
0.8
0.7
0.6
0.5
0
1
2
3
4
5
6
Training accuracy
Validation accuracy


Training and validation loss
Training loss
Validationloss
2.5
2.0
1.5
1.0
0.5
0.0
0
1
2
3
4
5
6



Web Application

CONCLUSIONS :

The OpenCV techniques are used to capture the images, 2-Dimensional Convolutional Neural Network is used to extract features and predict the gestures, PyAutoGUI is used to control the keyboard keys whenever a gesture is integrated with it is predicted. A custom dataset of 7 gestures is collected to test the proposed model. This model is also tested with these gestures in real-time to examine the accuracy of the proposed system. The proposed CNN model achieved a high accuracy of 98%, providing a user-friendly, cost-effective approach to interaction with computer systems. Thus, the proposed system is a true real-time model with low to negligible latency. The future scope is to work on improving the gesture recognition capabilities in varied environments such as illumination levels. Also, the integration of more functions is proposed with new hand gestures to use with other applications such as typing in word documents and web browsers.

REFERENCES :

- [1] Zhuang H, Yang M, Cui Z, Zheng Q. A method for static hand gesture recognition based on non-negative matrix factorization and compressive sensing. IAENG International Journal of Computer Science. 2017 Mar 1;44(1):52-9.

- [2] Paul PK, Kumar A, Ghosh M. Human-Computer Interaction and its Types: A Types. International Conference on Advancements in Computer Applications and Software Engineering (CASE 2012), At Chittorgarh, India. –2012 2012 Dec 21.

- [3] Ritupriya G.Andurkar, Human–computer interaction. International Research Journal of Engineering and Technology, (ISSN: 2395-0072 (P), 2395-0056 (O)). 2015; 2(6):744-72.

- [4] [4] Shanthakumar VA, Peng C, Hansberger J, Cao L, Meacham S, Blakely V. Design and evaluation of a hand gesture recognition approach for real-time interactions. Multimedia Tools and Applications. 2020 Feb 21:1-24.

- [5] Bashir A, Malik F, Haider F, Ehatisham-ul-Haq M, Raheel A, Arsalan A. A smart sensor-based gesture recognition system for media player control. In2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) 2020 Jan 29 (pp. 1-6). IEEE