



VIT[®]

Vellore Institute of Technology

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

Project Title: Touchless Controls

Course: CSE4015 – Human-Computer Interaction

“J – Component Project Report”

Team Member 1: Preyash 20BPS1022

Team Member 2: Disha Gupta 20BPS1078

Abstract

After the world was hit by the COVID-19 pandemic, we realized how important touchless controls are. Smart devices that could perform wonders like locating information, creating communication, etc. were all around us. But since several users could not share a single device because doing so could spread the virus through touch, they were all rendered useless. Technologies like touchless or gesture controls, which make up a significant and outstanding portion of the HCI area, were highly demanded.

The main goal of our project is to develop an application/ program that allows users to operate fundamental functions with only gestures. The use of the mouse can be reduced by this technology, which also provides a substitute for the mouse in the form of our motions.

Our approach would be little different from the proposed system in the Research Paper as we will be implementing as well as extending it in our project.

- We will be using Python, OpenCV, Media pipes and some other libraries and frameworks of python to implement this idea.
- The result would be that the user would be able to control basic functionalities just by using his gestures by fingers and palm.
- The basic input-output devices will include: Camera (Web cam), Monitor.

Introduction

The project aims on development of a system through which the user can perform tasks just using their gesture, there is no need for a physical touch between the user and the input/ hardware devices. The proposed model can be used in various domains. Contactless controls would bring a new revolution in the world of Human-Computer Interaction. The project would be using Open CV and MediaPipes and multiple modules to develop the program. The user has to just do some predefined gestures like to control the cursor user can use their index figure. For performing tasks like left click the user can use combination of index finger and middle finger. For performing task like right click, the user can use index finger, middle finger and ring finger. We can also perform functions like zooming images by using both of our hands to make a “L- Shaped” gesture using our thumbs and index finger. We can further enhance this model by adding more features to it.

Existing Systems

Recent laptops and computers consist of mouses and keyboards which the user uses to interact with the machine. The mouse was developed in 1960s by SRI International's Douglas Engelbart, while exploring human computer interactions. These methods are in use for a long time. With the forward advancing of the world in technology, machines with touch screen can be seen for real which in middle of last century seemed impossible. Now many machines are integrated with touchless user interface to better the user's experience while communicating with machines. Continuous research in this field is being done to improve and ease the way the user gives the input. Some other technologies which are used to give input to the computer may include:

Wired gloves to give input to the computer about the position and rotation of the hands using magnetic or inertial tracking devices. These gloves can detect finger bending and other gestures of the hands. The input is fed to the software which is translated for the computer to understand. This may seem bulky for normal users but a good option for the gamers who want real-time experiences.

Gesture based controllers are used as an extension to user's body. These record the movements and which are send to the software which can detect the meaning of the gesture.

Problem Statement

- The demand for the touch less interaction with the computer has increased due to the pandemic but not been addressed completely. Touch less form of control of computer is the need of the hour.
- The need to reduce the employment of physical devices (like mouse) to save time and effort.
- These hardware devices are not as ecologically benign as they seem, therefore they do have some restrictions.

Introduction to the Proposed Model

The proposed model uses technologies like OpenCV and MediaPipes for recognizing user images which are being inputted from the camera which is the input sensor. This image is then processed using MediaPipes and OpenCV. The proposed model has features like zooming images, controlling cursor, performing left-click, right-click, scrolling, changing windows and copy-paste. The user can control the zoom-in and the zoom-out feature by making a “L-shaped” gesture using both of his hand. The image can be expanded by distancing your both palms and it can be compressed by bringing your palms closed. The feature of controlling the cursor could be performed by moving the index finger. We can perform left-click by using a combination of index finger and the middle finger. The right-click by using a combination of index finger, middle finger and the ring finger. We can also perform tasks like opening new tabs, scrolling, copy-paste and typing.

Proposed Solutions

- Creating a computer vision-based system for identifying, recording, and comprehending gestures is the aim of this project.
- Using simple web camera and laptop as hardware and human gestures to control the screen to reduce the cost.
- Using python libraries to achieve basic functionalities just by hand gestures.

Components of the Model

Hardware:

Laptop/Computer

Web camera

Software:

Python

PyCharm

OpenCV

Mediapipes

Description of the components

Open CV

Open CV is an open- source library which is a tool for image processing and performing computer vision tasks. It has around 2500 optimized algorithms. It is used for face detection, objection tracking, landmark detection, etc.

Media Pipe

Media Pipe is a cross-platform pipeline framework to build custom machine learning solutions for live and streaming media. In reference for hand detection, we can use palm detection model of Media Pipe pipeline which will return an oriented hand bounding box from the full image.

Google Vision API vs OpenCV?

- Although Google Cloud Vision API easier to use, OpenCV is easier to set up and administer.
- According to the reviewers OpenCV fulfills the business needs better than Google Cloud Vision API.
- OpenCV's feature updates and roadmaps are more preferred than Google Cloud Vision API.
- Google Cloud Vision API rates 4.1/5 stars with 22 reviews. By contrast, OpenCV rates 4.5/5 stars with 39 reviews.

Block Diagram

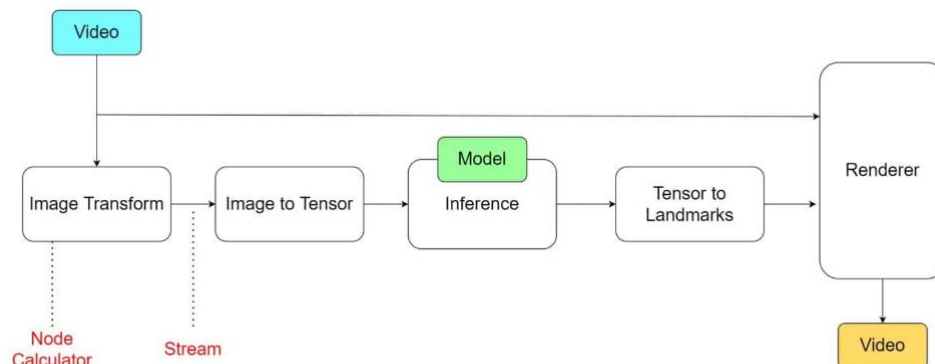


Fig1. MediaPipe Hand graph



Fig. 2 - Before MediaPipe

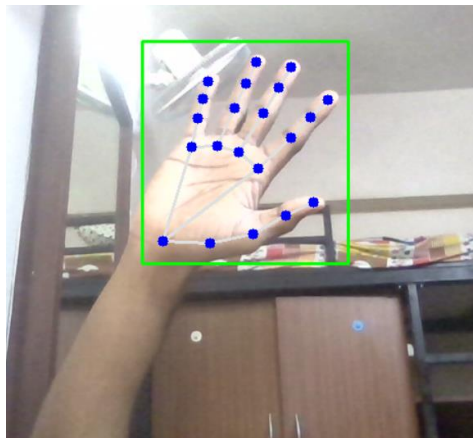


Fig. 3 - After MediaPipe

Working Mechanism

A framework called MediaPipe is used to create machine learning pipelines for processing time-series data, such as audio and video. The desktop/server, Android, iOS, and embedded devices like the Raspberry Pi and Jetson Nano all support this cross-platform framework. A graph is the name of the MediaPipe perception pipeline. Take the first option, Hands, as an example. We input a stream of photographs, and the output includes hand-rendered landmarks on the images. The block diagram above represents the MediaPipe hand graph. A collection of programming functions called OpenCV (Open-Source Computer Vision) is primarily focused on real-time computer vision. It is a library used for image processing, to put it simply. It is mostly used for all operations involving images. Using OpenCV we are performing tasks like reading and writing on the images.

Flow Chart

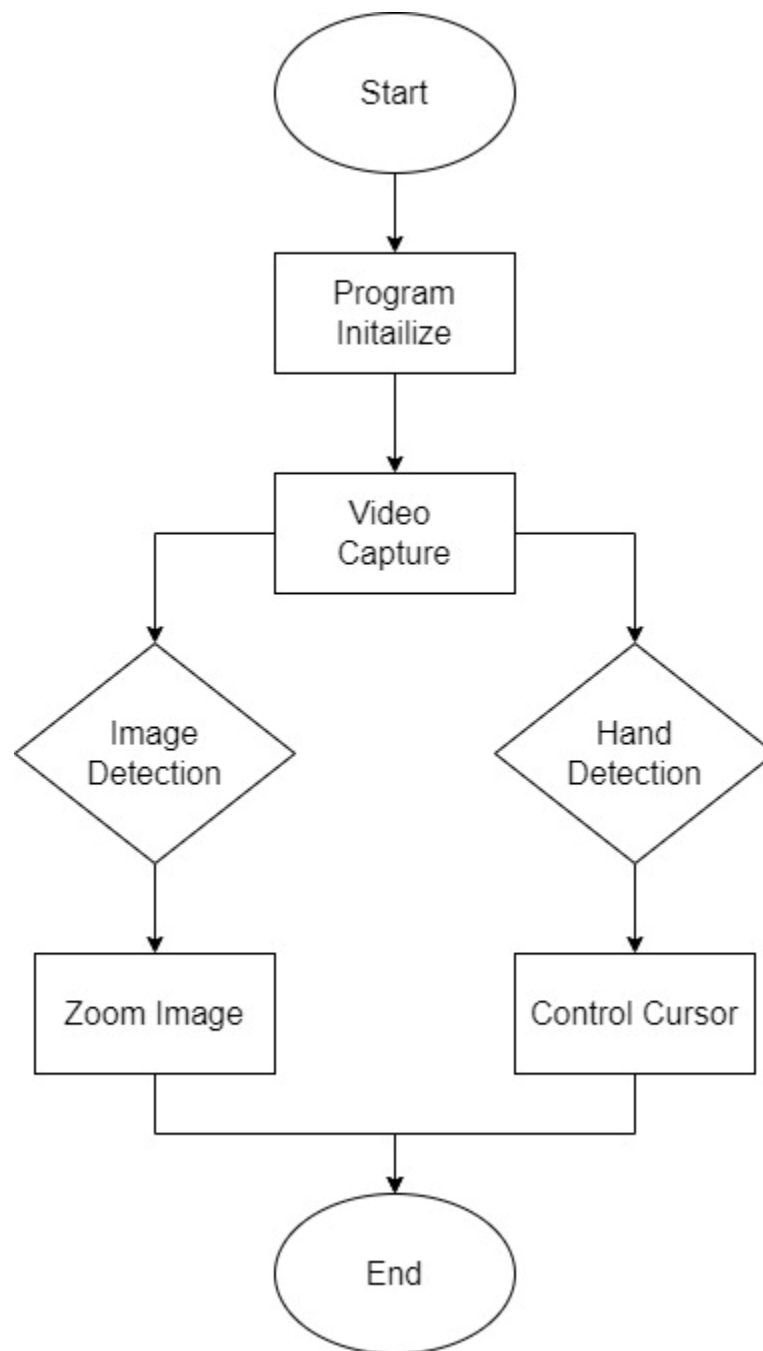


Fig. 4 - Flowchart

Input

The input for the computer would be taken from the camera, the input would be a constant image capturing.

Output Screenshots

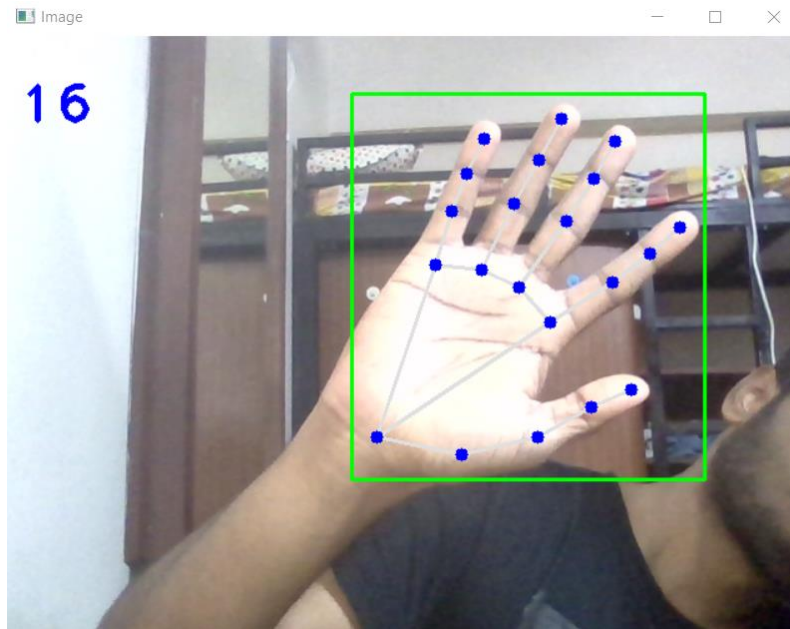


Fig. 5 - Hand Detection

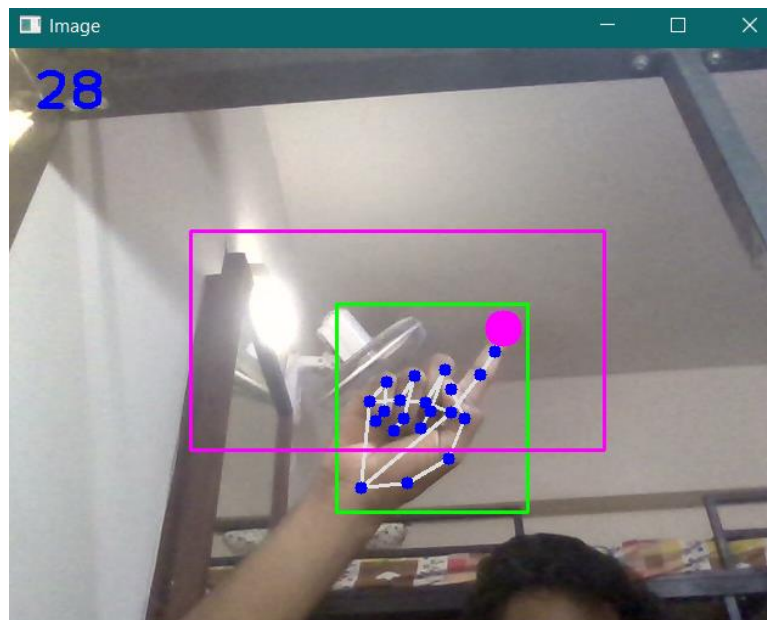


Fig. 6 - Controlling Cursor

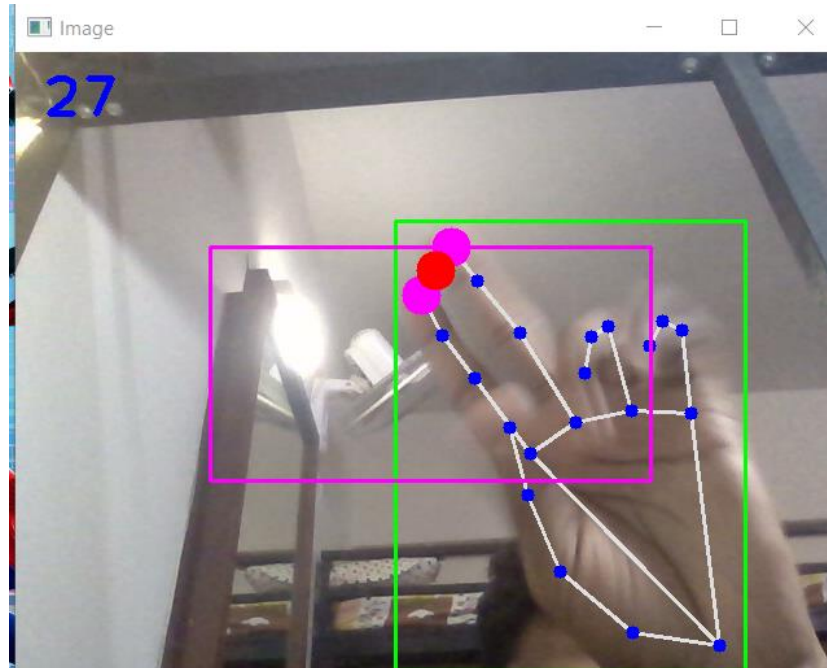


Fig. 7 - Left-Click

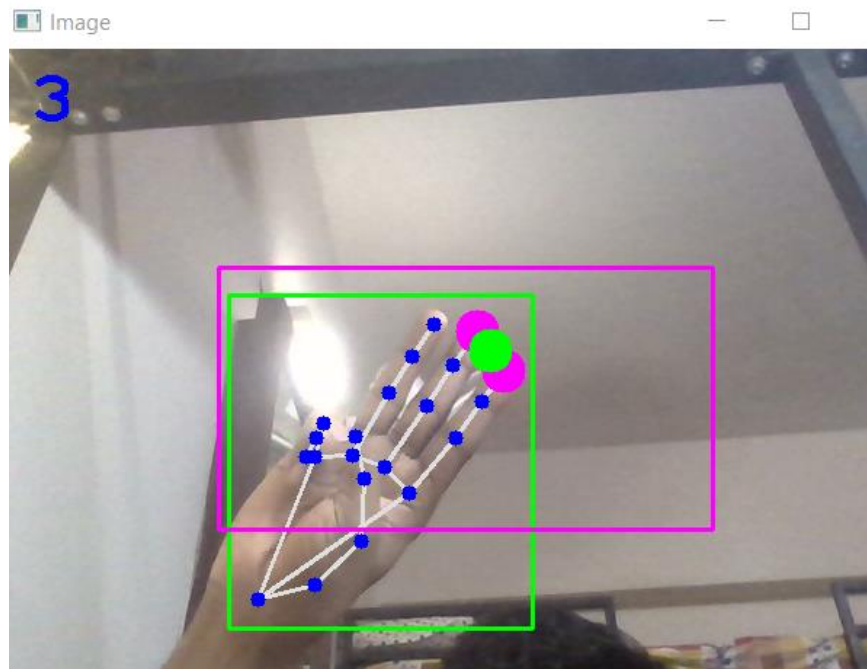


Fig. 8 - Right-Click

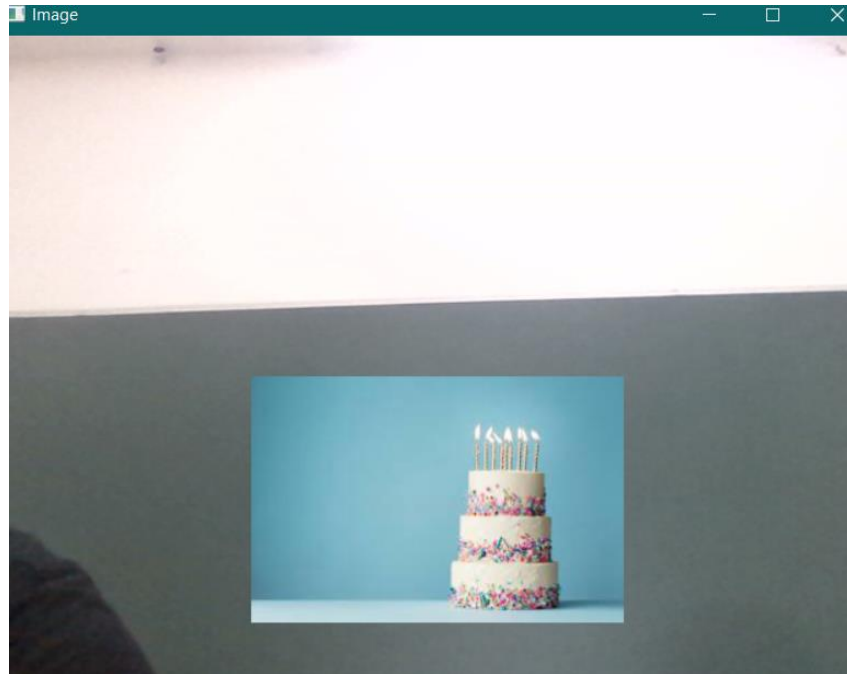


Fig. 9 - Image Detection

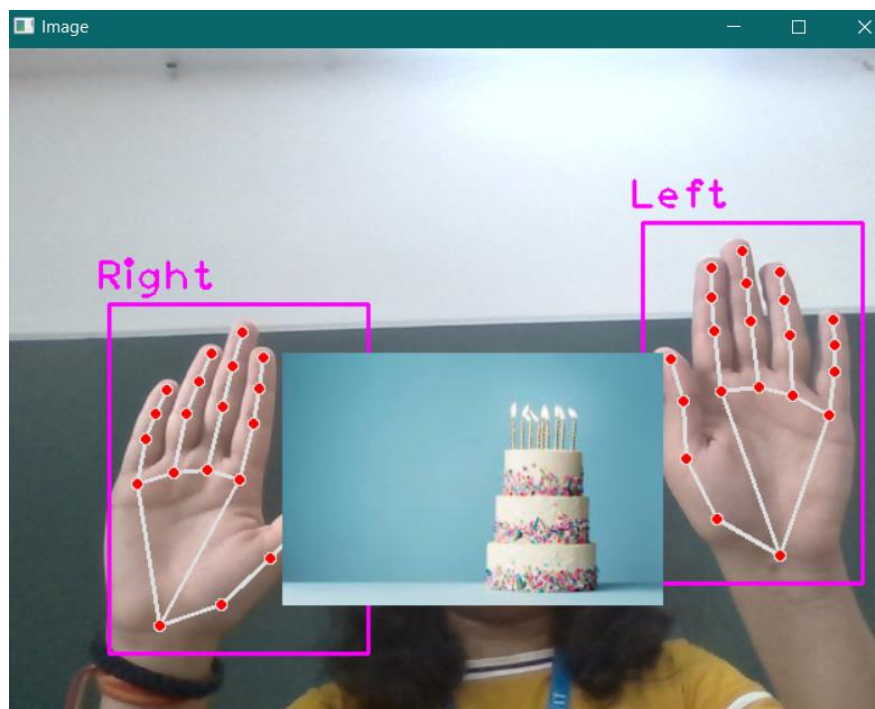


Fig. 10 - Palm Detection

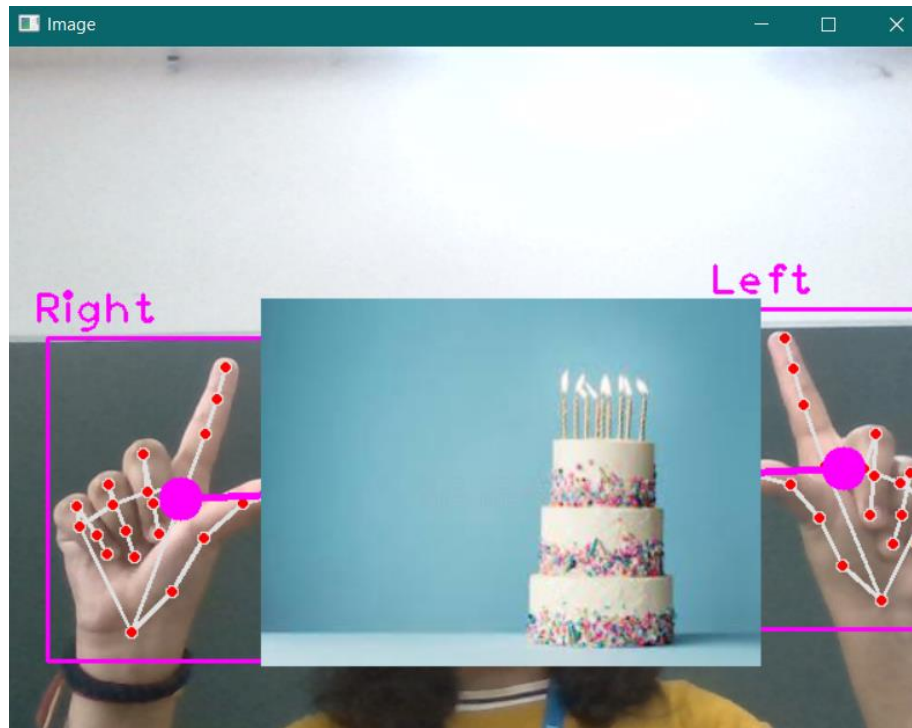


Fig. 11 – Zoom-in

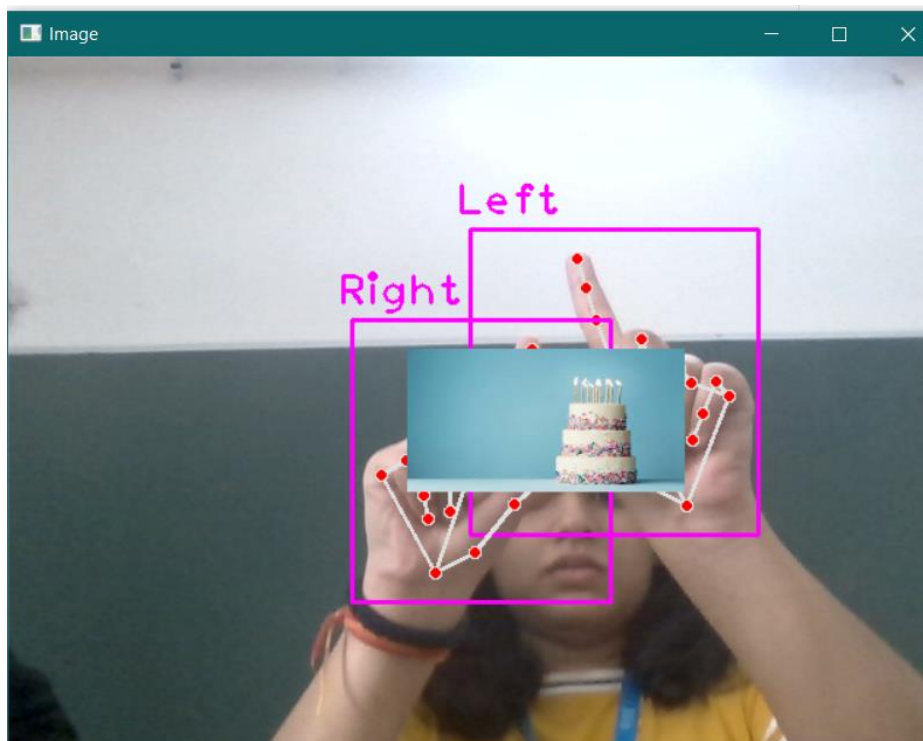


Fig. 12 – Zoom-out

Conclusion

The tasks like controlling mouse cursor, performing left-click, right-click, scrolling, changing tabs, opening applications were performed using the proposed model.

References

- <https://www.mdpi.com/2076-3417/11/6/2646/htm>
- <https://assets.researchsquare.com/files/rs-1616375/v1/5d27011b-ea68-416f-bb6f-f69c00be9b28.pdf?c=1653195556>
- <https://sci-hub.se/10.1109/ICSPS.2010.5555462>
- <https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html>
- <https://link.springer.com/article/10.1007/s11042-020-10156-5>
- <https://www.sciencedirect.com/science/article/pii/S1877050917319130>
- <https://sci-hub.se/10.1109/ICASERT.2019.8934612>
- <https://sci-hub.se/10.1109/ICSPS.2010.5555462>
- <https://www.g2.com/compare/google-cloud-vision-api-vs-opencv>
- <https://stackshare.io/stackups/google-cloud-vision-api-vs-opencv#:~:text=Google%20Cloud%20Vision%20API%20belongs,Insight%20From%20Your%20Images>
- <https://www.geeksforgeeks.org/right-and-left-hand-detection-using-python/>
- <https://www.analyticsvidhya.com/blog/2021/09/some-amazing-applications-of-opencv-library/>