



An-Najah National University

Faculty of Engineering and Information

Technology Telecommunication Engineering

Department Graduation Project Report 2

***" Deep Learning-Based Real-Time AI Virtual
Mouse System Using Computer Vision "***

Prepared by:

Arwa Ahmad[11744238]

Rawan Aub Ali [11612415]

Supervisor:

Dr. Falah Hassan

**A report Presented in partial fulfillment of the requirements for
Bachelor degree in (Telecommunication Engineering)**

Academic Year: 2020/2021

Acknowledgment

At the outset, we take this opportunity to express our sincere gratitude to all those who helped us complete our project report successfully. We are very grateful to Dr. Falah Hassan for demonstrating this opportunity for us to conduct an independent study on this topic.

Table of Contents:

| | |
|---|----|
| Acknowledgment..... | I |
| List of Figures..... | II |
| List of Abbreviations..... | IV |
| Abstract..... | V |
| Chapter 1: Introduction..... | 1 |
| 1.1 : Overview..... | 2 |
| 1.2 : Objectives..... | 2 |
| 1.3 : Problem Statement..... | 2 |
| 1.4 : Report Organization..... | 3 |
| Chapter 2: Constraints, Standards and Earlier coursework..... | 5 |
| 2.1 : Constraints..... | 6 |
| 2.2 : Standards..... | 6 |
| Chapter 3: Related Works..... | 8 |
| Chapter 4: Methodology..... | 9 |
| 4.1: System Software..... | 10 |
| 4.2: System Flowchart..... | 13 |
| Chapter 5: SWOT Analysis and Feasibility Study..... | 23 |
| 5.1 : SWOT Analysis..... | 24 |
| 5.2 : Feasibility Study..... | 25 |
| Chapter 6: Experimental Results and Evaluation..... | 26 |
| Chapter 7: Conclusion and Recommendations..... | 30 |
| 7.1 : Conclusion..... | 31 |
| 7.2 : Recommendations..... | 31 |
| References..... | 32 |
| Appendix A: Disclaimer statement..... | 35 |

List of Figures:

| | |
|--|----|
| Figure 1: MediaPipe hand recognition graph..... | 11 |
| Figure 2: Co-ordinates or land marks in the hand..... | 12 |
| Figure 3: Flowchart of the real-time AI virtual mouse system..... | 14 |
| Figure 4: Capturing video using the webcam (computer vision)..... | 15 |
| Figure 5: Mouse cursor moving around the computer window..... | 16 |
| Figure 6: Gesture for the computer to perform left button click..... | 17 |
| Figure 7: Gesture for the computer to perform right button click..... | 18 |
| Figure 8: For the Mouse to Perform Scroll up Function | 19 |
| Figure 9: For the Mouse to Perform Scroll down Function..... | 20 |
| Figure 10: Volume up..... | 20 |
| Figure 11: Reduce volume..... | 21 |
| Figure 12: File Transfer..... | 22 |
| Figure 13: Graph of accuracy..... | 28 |
| Figure 14: Graph for comparison between the models..... | 29 |

List of Abbreviations:

AI: Artificial Intelligence

CPU: Central Processing Unit

GPU: Graphics processing unit

HCI : Human-Computer Interaction

MI: Machine Learning.

*SWOT: Strengths, Weaknesses, Opportunities, and
Threats*

List of tables:

Table (1):Experimental results.....27
Table (2):Comparison with existing models.....28

Abstract:

The mouse is one of the coolest inventions of human-computer interaction (HCI). Because it uses a battery for power and a dongle to connect to your computer, a wireless mouse or Bluetooth mouse still uses hardware and isn't completely hardware-free. This problem can be overcome in the proposed AI virtual mouse system by using a webcam or a built-in camera to capture hand movements and recognize hand tips using computer vision. The algorithm of the system is based on the method of machine learning. The computer can be controlled remotely using hand gestures and can perform left-click, right-click, scrolling, and computer cursor tasks without the use of a hardware mouse. To detect hands, the algorithm uses deep learning. As a result, the proposed approach will prevent the spread of COVID-19 by removing human interaction and reliance on remote control of computers.

Chapter 1

Introduction

Chapter 1: Introduction

1.1: Overview

With the advancement of technology in the areas of augmented reality and everyday devices, these devices are becoming more compact in the form of Bluetooth or wireless technologies. This study presents an AI virtual mouse system that uses computer vision to perform mouse activities in the computer utilizing hand motions and hand tip detection. The suggested system's main goal is to employ a web camera or a computer's built-in camera to execute computer mouse cursor and scroll tasks instead of a standard mouse device. Computer vision is used to identify hand gestures and tip detection as an HCI [1] with the computer. We can utilize a built-in camera or a web camera to track the fingertip of a hand gesture and conduct mouse cursor operations and scrolling, as well as move the cursor using the AI virtual mouse system.

When using a wireless mouse or Bluetooth, several items are used to connect to the computer, such as a mouse, a dongle, and a battery to power the mouse, but in this project, the user utilizes his built-in camera or webcam to control the operations of the computer mouse. The webcam in the proposed system captures captured frames, analyses them, identifies various hand gestures and hand movements, and then performs the requested mouse operation.

The AI virtual mouse system was created using the Python programming language, as well as OpenCV, a computer vision library. The model in the proposed AI virtual mouse system makes use of the MediaPipe package for tracking the hands and the tip of the hands, as well as the Pynput, Autopy, and PyAutoGUI packages for moving around the computer's window screen and performing functions like left click, right click, and scrolling. The proposed model's results demonstrated a very high level of accuracy, and the proposed model can function extremely well in real-world applications using only a CPU without the use of a GPU.

1.2: Objectives

The main goal of the proposed AI virtual mouse system is to create an alternative to the regular and traditional mouse system to perform and control the functions of a mouse. This can be achieved by using a webcam that captures hand gestures and then manipulates these frames to perform certain mouse functions such as left click, right click ,scrolling , Double Click and so on .

1.3: Problem Statement

The proposed AI virtual mouse system can be used to solve real-world difficulties such as instances where there isn't enough space to utilize a physical mouse and for remote control in meetings. Furthermore, in the midst of the COVID-19 outbreak, it is not safe to operate equipment by touching them, since this could result in viral transmission, hence the proposed artificial intelligence virtual mouse can be utilized to circumvent this. Since using a webcam or built-in camera to operate computer mouse functions via a hand gesture and hand tip detection, these issues have arisen moreover we can get benefits from this project in handling presentations and meetings in such a good way and control the shows it should be controlled.

1.4: Report Organization

This report is organized as follows: The first chapter gives an introduction about the problem and the objectives of the project. Chapter two, talks about the constraints we faced while working on the project, and about the standards and we have used in the system. In the third chapter, we present the previous related works of the project. Chapter four, explained the methodology of the project. In chapter five, talk about SWOT analysis and Feasibility Study. In chapter six, discuss the results and analysis. The final chapter summarized the project in conclusion.

Chapter 2
Constraints,
Standards and
Earlier
coursework

Chapter 2: Constraints, Standards and Earlier coursework

2.1: Constraints

We faced many problems while we are working on this project:

- Difficulty understanding the working principle
- We had difficulty writing the code and making the program execute commands
- Learning a new programming language such as python without having that experience in coding was a very difficult issue to go through.
- Shortness of the time period that we had was a big challenge in which we had to learn new computer technologies like machine learning as well as artificial intelligence

2.2: Standards

Python 3.9.0: Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy-to-learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

What makes 3.9.0 special is the first version of Python to default to the 64-bit installer on Windows. The installer now also actively disallows installation on Windows 7. Python 3.9 is incompatible with this unsupported version of Windows

Chapter 3

Related Works

Chapter 3: Related Works

There have been some analogous virtual mouse works that use hand gesture detection by wearing a glove in the hand and also using color tips in the hands for gesture recognition, but they are not as accurate in mouse functionalities. Because of the gloves, the recognition is not as exact; also, the gloves are not suitable for some users, and in some circumstances, the recognition is not as accurate due to the failure of color tip detection. The hand gesture interface has been detected using a camera in some cases.

Quam first introduced a hardware-based solution in 1990, which required the user to wear a Data Glove [2]. Although Quam's proposed approach produces more accurate results, it is impossible to conduct certain of the gesture commands with it.

A study on "A Real-Time Hand Gesture Recognition System Using Motion History Image" was proposed by Dung-Hua Liou, ChenChiung Hsieh, and David Lee in 2010 [3]. The model's biggest flaw is its inability to handle increasingly complex hand gestures.

A study on "Cursor Control System Using Hand Gesture Recognition" was offered by Monika B. Gandhi, Sneha U. Dudhane, and Ashwini M. Patil in 2013 [4]. The constraint in this study is that saved frames must be processed for hand segmentation and skin pixel recognition.

In the IJCA Journal, Vinay Kr. Pasi, Saurabh Singh, and Pooja Kumari suggested "Cursor Control via Hand Gestures" in 2016 [5]. The system suggests that different bands be used to conduct distinct mouse tasks. The drawback is that mouse functions are dependent on distinct colors.

"Virtual Mouse Using Hand Gesture" was proposed by Chaithanya C, Lisho Thomas, Naveen Wilson, and Abhilash SS in 2018 [6], where the model identification is based on colors. However, just a few mouse functions are used.

Chapter 4

Methodology

Chapter 4: Methodology

4.1 : System Software

4.1.1 Python (programming language)

Python is an interpreted, high-level general-purpose programming language.

Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented, and functional programming. It is often described as a "battery-included" language due to its extensive standard library.

Guido van Rossum began working on Python in the late 1980s, succeeding the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as comprehensive menus and a cycle detection garbage collection system (as well as reference counting). Python 3.0 was released in 2008 and was a major revision to a language that was not fully backward compatible. Python2 was discontinued with version 2.7.18 in 2020. [7]

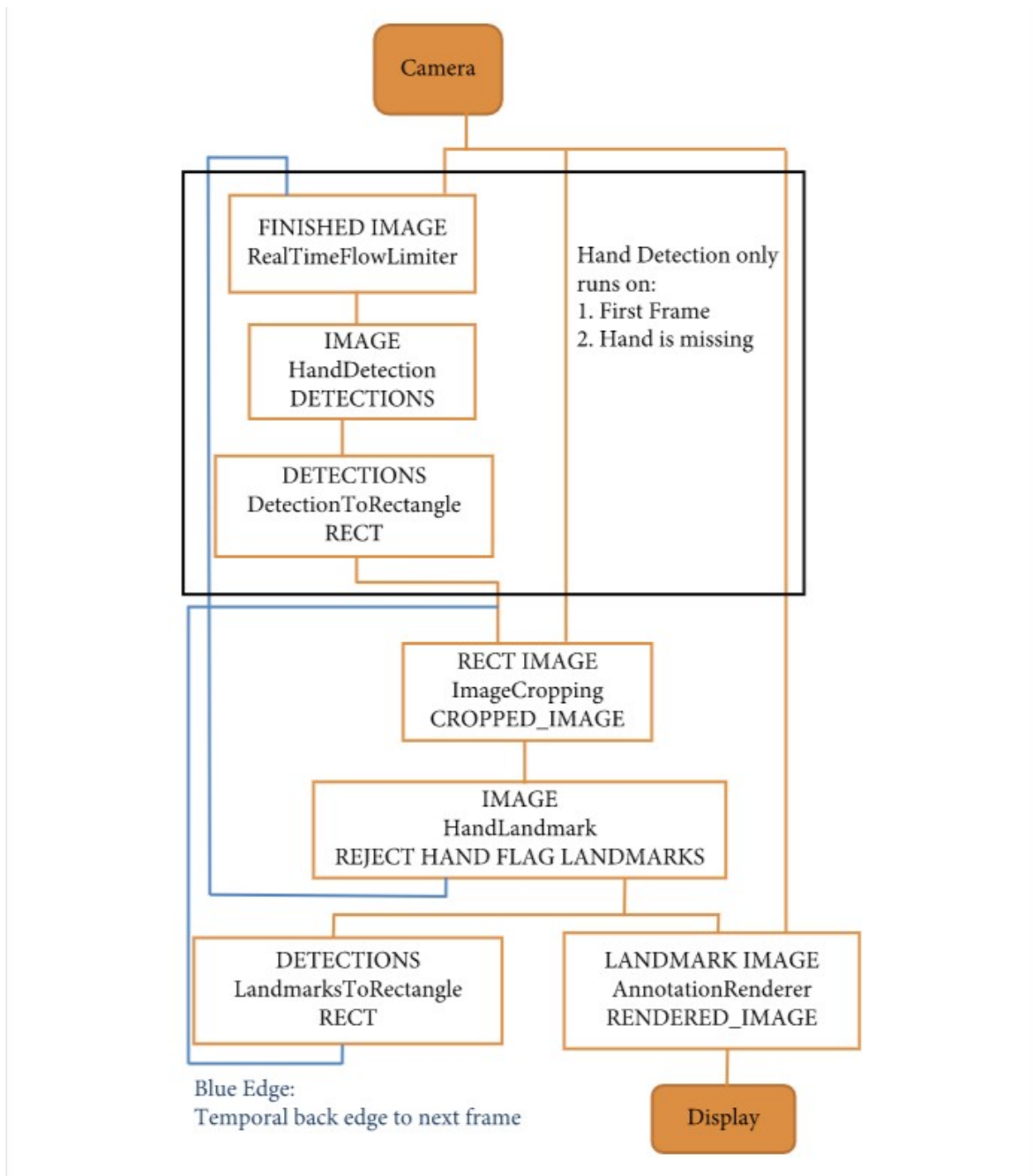
Python can be used as a scripting language for web applications, OpenCV contains Python bindings with a rich set of features for computer vision and image processing.

Python is commonly used in Artificial Intelligence(AI) and Machine Learning(ML) projects with the help of libraries such as TensorFlow, Keras, Pytorch, and Scikit-Learn. [8][9][10][11] As a scripting language with a modular syntax, simple syntax and rich text processing tools, Python is often used for natural language processing.

4.1.2 MediaPipe

MediaPipe is a framework which is used for applying in a machine learning pipeline, and it is an open-source framework of Google. The MediaPipe framework is useful for cross-platform development since the framework is built using the time series data. The MediaPipe framework is multimodal, where this framework can be applied to various audios and videos [12]. The MediaPipe framework is used by the developer for building and analyzing the systems through graphs, and it also been used for developing the systems for the application purpose. The steps involved in the system that uses MediaPipe are carried out in the pipeline configuration. The pipeline created can run in various platforms allowing scalability in mobile and desktops. The MediaPipe framework is based on three fundamental parts; they are performance evaluation, framework for retrieving sensor data, and a collection of components which are called calculators [12], and they are reusable. A pipeline is a graph which consists of components called calculators, where each calculator is connected by streams in which the packets of data flow through. Developers are able to replace or define custom calculators anywhere in the graph creating their own application.

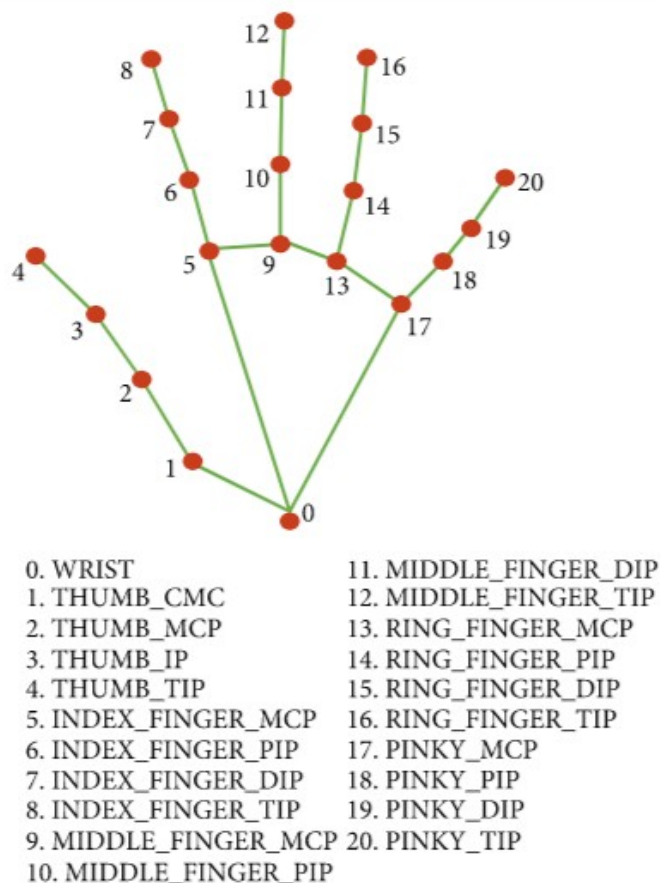
The calculators and streams combined create a data-flow diagram; the graph (**Figure 1**) is created with MediaPipe where each node is a calculator and the nodes are connected by streams [12].



Figure(1) MediaPipe hand recognition graph

In order to identify and recognize a hand or palm in real time, a single-shot detector model is used. The MediaPipe employs the single-shot detector concept. It is first trained for a palm detection model in the hand detection module since palms are easier to train. Furthermore, nonmaximum suppression is far more effective on small items like palms or

fists [13]. As shown in **(Figure 2)**, a model of hand landmark consists of finding 21 joint or knuckle co-ordinates in the hand region.



Figure(2) Co-ordinates or land marks in the hand

4.1.3 OpenCV

OpenCV is a computer vision package that includes object detection picture processing techniques [14]. The computer vision library OpenCV is a python programming language library that may be used to create real-time computer vision applications. The OpenCV library is used to process images and videos, as well as perform analyses such as face and object detection [15].

4.2: System Flowchart

The flowchart of the real-time AI virtual mouse system in)**Figure 3**(explains the many functions and conditions used in the system.

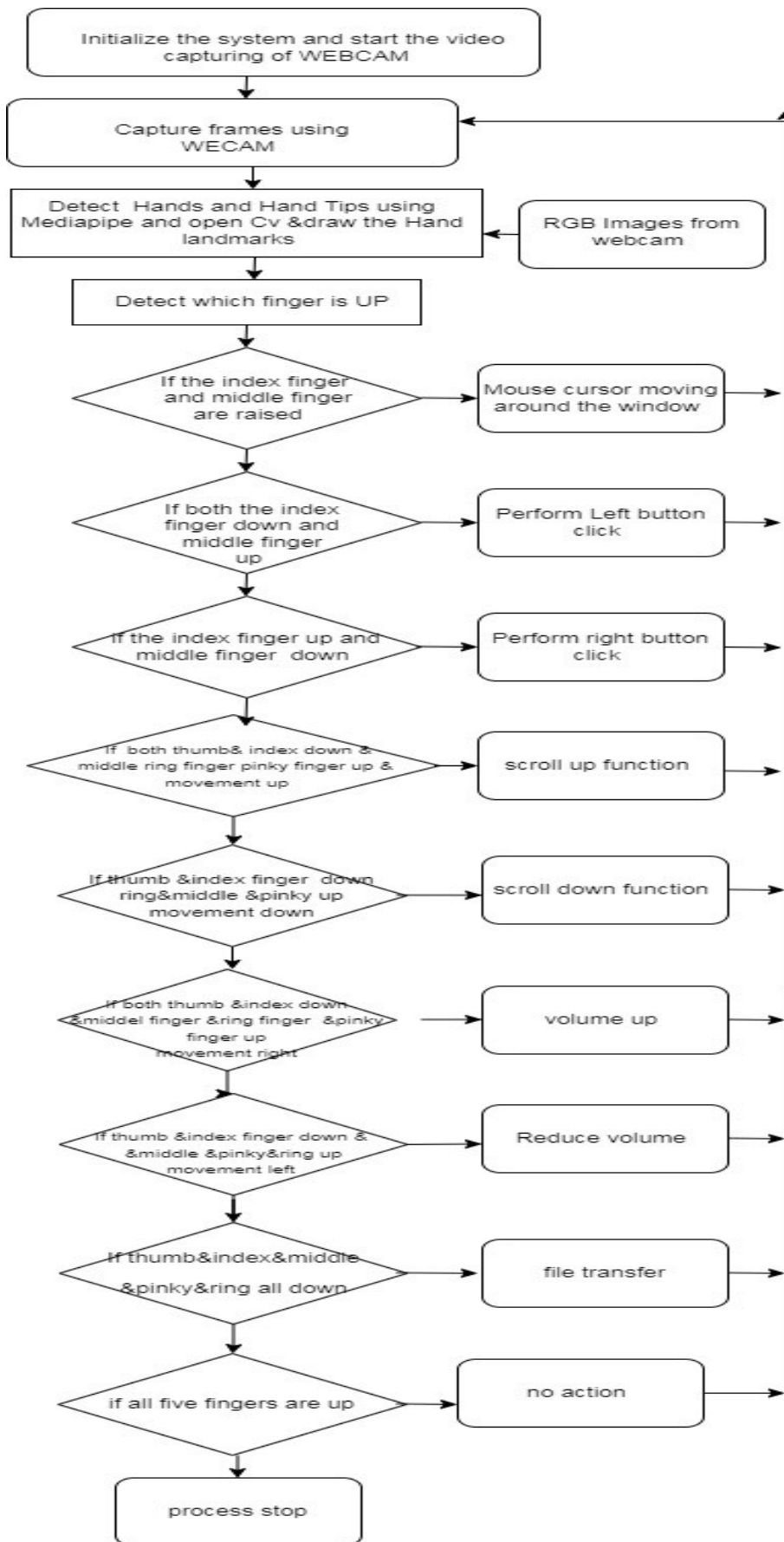


Figure (3): Flowchart of the real-time AI virtual mouse system.

- 1. The Camera Used in the AI Virtual Mouse System** The proposed AI virtual mouse technology is based on the frames acquired by a laptop or computer's webcam. The video capture object is constructed using the Python computer vision library OpenCV, and the web camera will begin capturing video, as illustrated in **(Figure 4)**. The frames are captured by the web camera and sent to the AI virtual system.

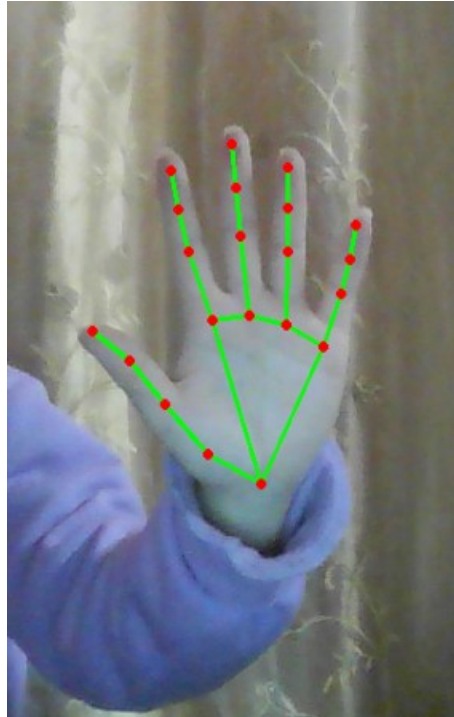


Figure (4) :Capturing video using the webcam (computer vision).

The video frames are processed from BGR to RGB color space to find the hands in the video frame by frame as shown in the following code:

```
def findHands(self, img, draw = True):  
imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  
self.results = self.hands.process(imgRGB)
```

- 2. Detecting Which Finger Is Up and Performing the Particular Mouse Function.** In this stage, we are detecting which finger is up using the tip Id of the respective finger that we found using the MediaPipe and the respective coordinates of the fingers that are up , and according to that, the particular mouse function is performed.

3. In order to move the mouse pointer around the computer window. If the index finger is raised with the tip Id = 8 and the middle finger is with the tip Id = 12 up, then the mouse pointer is made to move around the computer window using the AutoPy package from Python, as shown in)**Figure 5**).

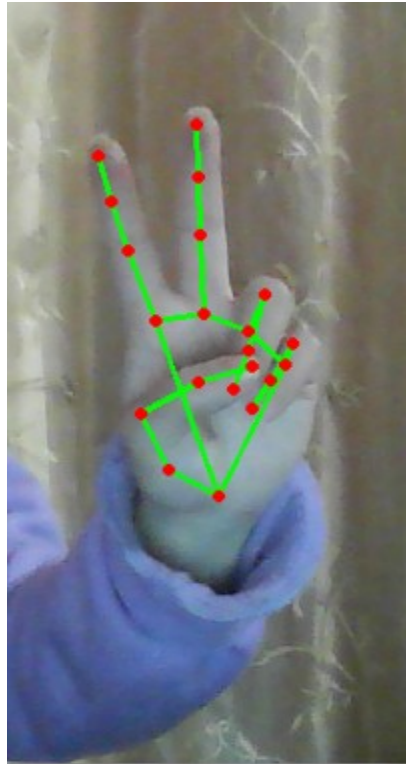


Figure (5): Mouse cursor moving around the computer window.

4. For the mouse to perform click the left click. If both the index finger with the tip Id = 8 down and the middle finger with the tip Id = 12 up and the distance between the two fingers is less than 30 pixels, then the computer is designed to perform a left-click with pynput.



)Figure 6 :(Gesture for the computer to perform left button click.

5. For mouse to perform right click. If the index finger with tip Id = 8 up and middle finger with tip Id = 12 down and the distance between the two fingers is less than 40 pixels, then the computer is designed to perform a right-click using the Python pynput package, as shown in (Figure 7).

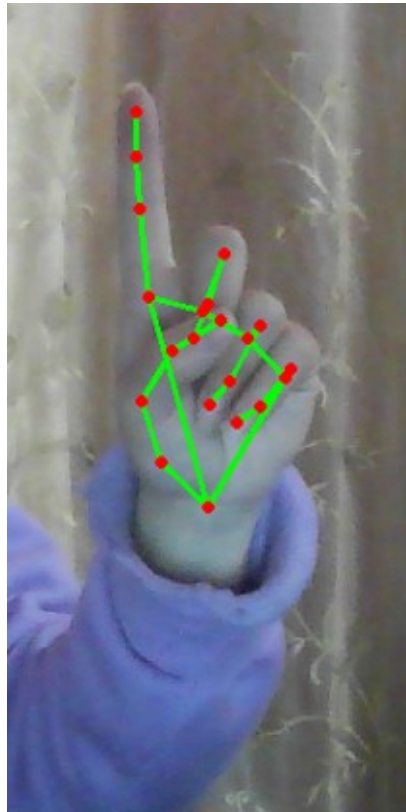


Figure (7): Gesture for the computer to perform right button click.

6. For the Mouse to Perform Scroll up Function If thumb with Id tip = 0, index finger with Id tip = 8 down, middle finger with Id = 12, ring finger with Id = 16, pinky finger with Id = 20 up and the movement is up. as shown in **(Figure8)**.

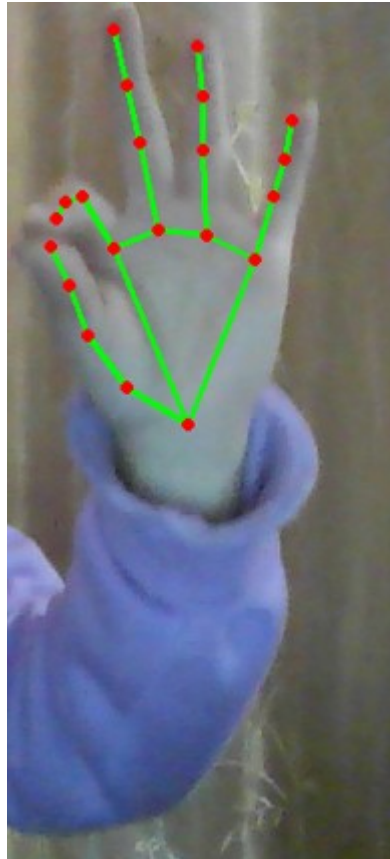


Figure (8): For the Mouse to Perform Scroll up Function

7. For the Mouse to Perform Scroll down Function If thumb with ID tip = 0, index finger with ID tip = 8 down, middle finger with ID = 12, ring finger with ID = 16, pinky finger with ID = 20 up and movement down. . as shown in **(Figure9)**.

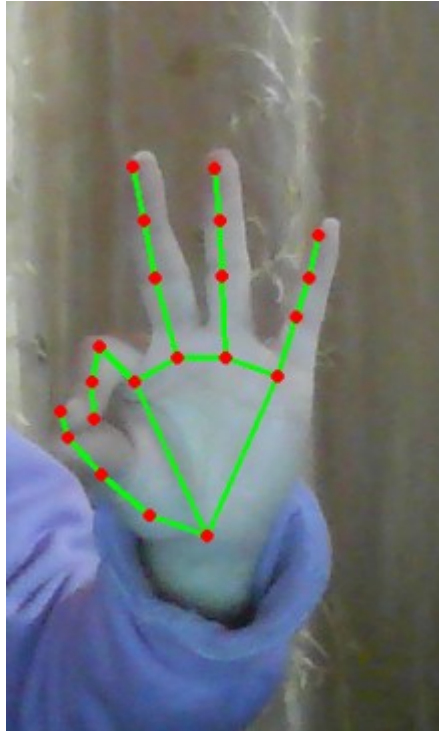


Figure (9): For the Mouse to Perform Scroll down Function

8. Volume up If thumb with ID tip = 0, index finger with ID tip = 8 down, middle finger with ID = 12, ring finger with ID = 16, pinky finger with ID = 20 up and right movement. as shown in **(Figure10)**.

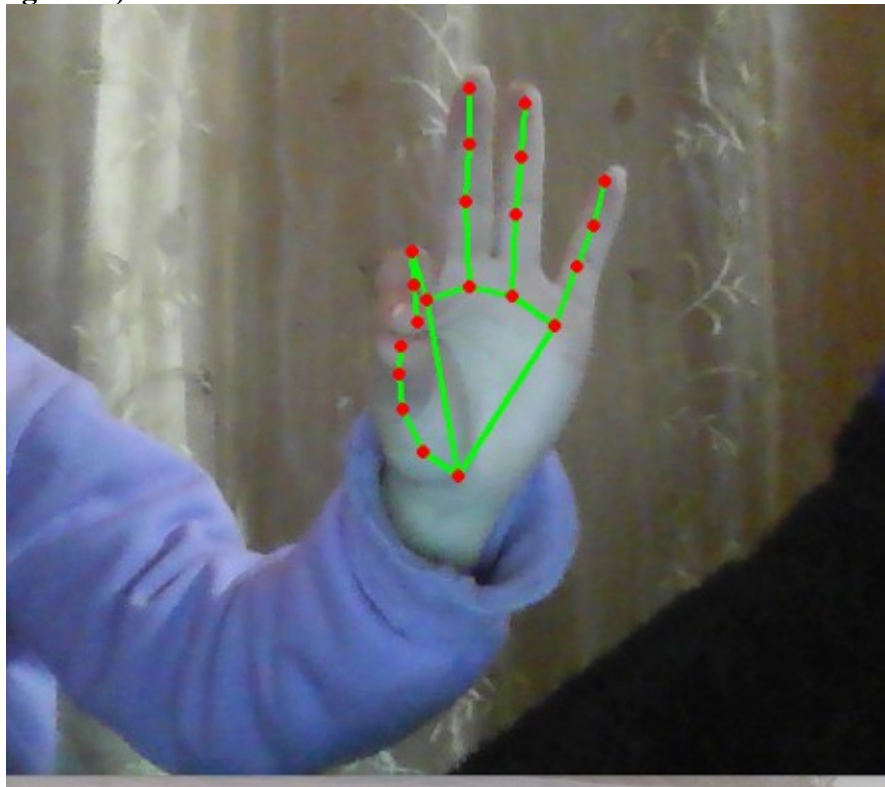


Figure (10): Volume up

9. Reduce volume If thumb with ID tip = 0, index finger with ID tip = 8 down, middle finger with ID = 12, ring finger with ID = 16, pinky finger with ID = 20 up and left movement. as shown in **(Figure11)**.

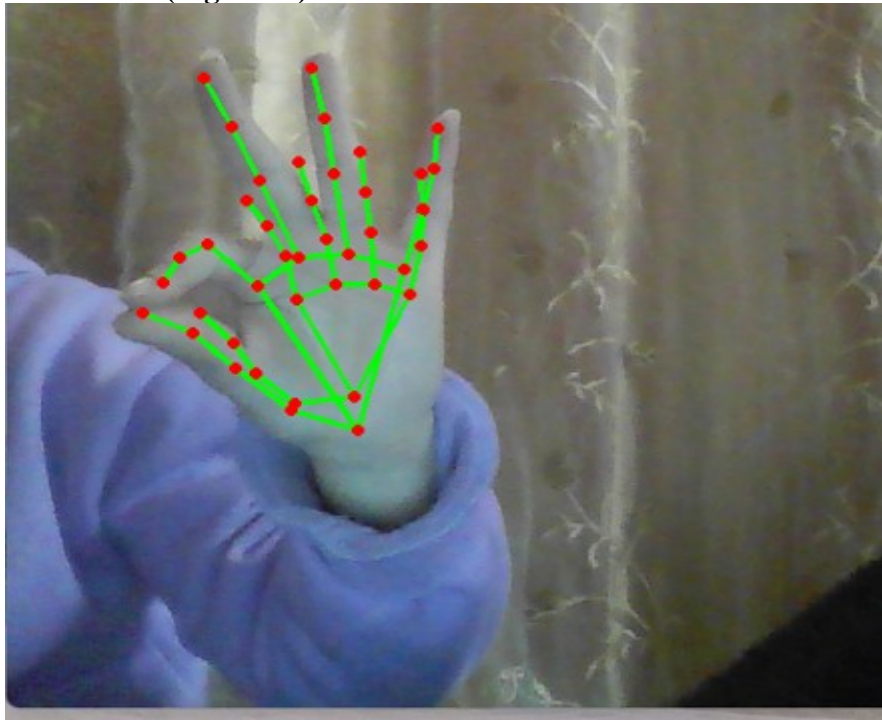


Figure (11): Reduce volume

10. File Transfer If thumb with ID tip = 0, index finger with ID tip = 8 Middle finger with ID = 12, ring finger with ID = 16, pinky finger with ID = 20 down moves the file. as shown in **(Figure12)**.



Figure (12): File Transfer

Chapter 5

SWOT Analysis
and Feasibility
Study

5.1: SWOT Analysis

A SWOT analysis is a strategy technique simple, but powerful tool is acronym for Strengths, Weaknesses, Opportunities, and Threats, that can used it to understanding all sorts of situations in projects or organizations for strategic planning.

By estimating the data for internal factors (strengths, weakness) and external factors (opportunities, threats) this project is also assessed and knowing how is much positive. As shown:

Strengths:

- ✓ There is no need of external hardware or software to use this application for user, user only need a laptop or computer which has a webcam
- ✓ This application is free for user.

Weakness:

- × its not as accurate as the physical mouse
- × there is some lag in capturing the frames and it need a powerful CPU to operate well

Opportunities:

- ✓ This application provide a wonderful set of features that facilitate users' lives, in addition to making them more comfortable, so you can take advantage of this application and add other features to it to make life more felixible and able to do further morethings.

Threats:

- × No internet access.

5.2: Feasibility Study

In the project, no budget needed because we used a software program.

Chapter 6

Experimental

Results and

Evaluation

The concept of enhancing human-computer interaction using computer vision is presented in the suggested AI virtual mouse system. Because there are just a few datasets available, cross-comparing the testing of the AI virtual mouse system is problematic. Hand gestures and finger tip detection have been tested in a variety of lighting conditions, as well as at varied distances from the webcam for tracking. To describe the outcomes in Table, an experimental test was undertaken.

The test was performed 25 times by 4 persons resulting in 600 gestures with manual labelling, and this test has been made in different light conditions and at different distances from the screen, and each person tested the AI virtual mouse system 10 times in normal light conditions, 5 times in faint light conditions, 5 times in close distance from the webcam, and 5 times in long distance from the webcam, and the experimental results are tabulated in (**Table 1**).

| Mouse function performed | Success | Failure | Accuracy(%) |
|--------------------------|---------|---------|-------------|
| Mouse movement | 100 | 0 | 100% |
| Left button click | 98 | 2 | 98% |
| Right button click | 99 | 1 | 99% |
| Scroll function | 93 | 7 | 93% |
| Brightness control | 95 | 5 | 95% |
| Volume control | 96 | 4 | 96% |
| No action performed | 100 | 0 | 100% |
| Result | 681 | 19 | 97.28% |

Table (1): Experimental results.

Table shows that the planned AI virtual mouse system had attained an accuracy of around 97 %. We may conclude that the proposed AI virtual mouse system functioned well based on its 97 % accuracy. The accuracy for "Scroll function" is low, as shown in Table, because this is the most difficult motion for the computer to grasp. Because the gesture required to conduct the specific mouse action is more difficult, the accuracy of the scroll function is low. For all of the other gestures, the accuracy is also very good and high. In comparison to earlier virtual mouse approaches, our model performed admirably, with a 97 % accuracy rate.)**Figure13** (shows a graph of accuracy.

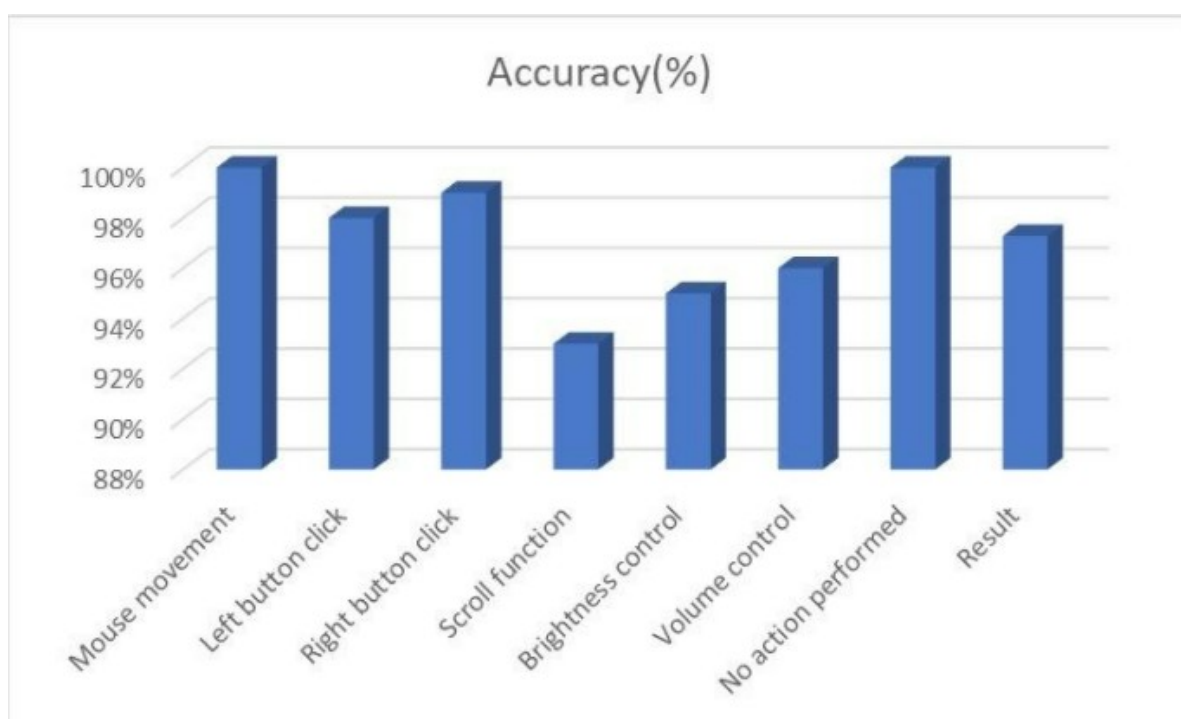


Figure)13: (Graph of accuracy.

In terms of accuracy, (**Table 2**) compares the existing models with the proposed AI virtual mouse model.

| Existing models | Accuracy(%) |
|--|-------------|
| Virtual mouse system using RGB-D images and fingertip detection [17] | 96.13 |
| Palm and finger recognition based [18] | 78 |
| Hand gesture-based virtual mouse [19] | 78 |
| The proposed AI virtual mouse system | 99 |

Table (2):Comparison with existing models

(Table 2) shows that when compared to previous virtual mouse models, the suggested AI virtual mouse performed exceptionally well in terms of accuracy. The proposed model is unique in that it can execute most mouse tasks, such as left and right clicks, scroll up and down, and mouse pointer movement, utilizing finger tip recognition, and it can also control the PC in a virtual mode like a hardware mouse. (Figure14) depicts a comparison graph comparing the models.

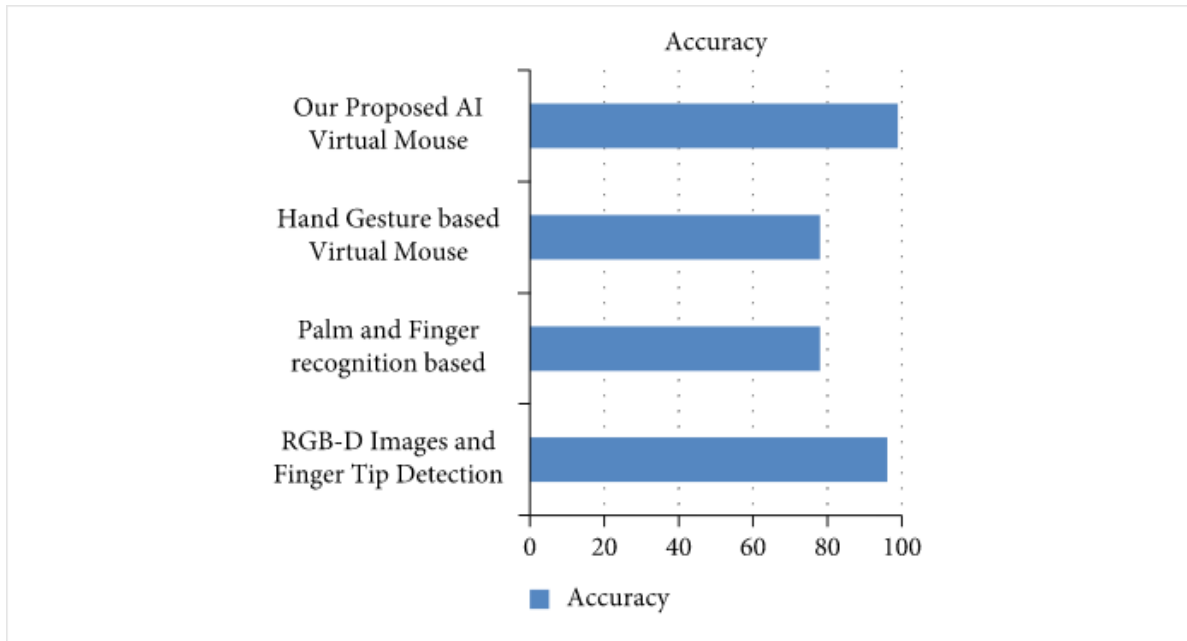


Figure (14) :- Graph for comparison between the models.

Chapter 7

Conclusion and Recommendations

7.1: Conclusion

A few strategies have to be devised because accuracy and efficiency play a key role in making the application as helpful as an actual physical mouse. Following the implementation of such an application, the physical mouse is mostly replaced, and no actual mouse is required. This motion tracking mouse performs all of the physical mouse's movements (virtual mouse).

There are several features and improvements needed in order for the program to be more user friendly, accurate, and flexible in various environments. The following describes the improvements and the features required:

- a) **Smart Movement:** Because the existing recognition process is limited to a 25cm radius, adaptive zoom in/out functions are necessary to increase the covered distance. These functions can automatically modify the focus rate dependent on the distance between the users and the webcam.
- b) **Better Accuracy & Performance:** The response time is significantly reliant on the machine's hardware, which includes the processor's processing speed, the size of the available RAM, and the webcam's available capabilities. As a result, when the software is performed on a respectable system with a webcam that performs well in a variety of lighting conditions, the program may operate better.
- c) **Mobile Application:** In future this web application also able to use on Android devices, where touchscreen concept is replaced by hand gestures.

7.2: Recommendations

There are several ways to improve our project, for instance:

1. We should improve the quality of code and libraries that we are using so we can get a better results

References

References:

- [1]: J. Katona, "A review of human–computer interaction and virtual reality research fields in cognitive Info Communications," *Applied Sciences*, vol. 11, no. 6, p. 2646, 2021.
- [2]. D. L. Quam, "Gesture recognition with a DataGlove," *IEEE Conference on Aerospace and Electronics*, vol. 2, pp. 755–760, 1990.
- [3]. D.-H. Liou, D. Lee, and C.-C. Hsieh, "A real time hand gesture recognition system using motion history image," in *Proceedings of the 2010 2nd International Conference on Signal Processing Systems*, IEEE, Dalian, China, July 2010.
- [4]. S. U. Dudhane, "Cursor control system using hand gesture recognition," *IJARCCCE*, vol. 2, no. 5, 2013.
- [5]. K. P. Vinay, "Cursor control using hand gestures," *International Journal of Critical Accounting*, vol. 0975–8887, 2016.
- [6]. L. Thomas, "Virtual mouse using hand gesture," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 4, 2018.
- [7]. Peterson, Benjamin (20 April 2020). "Python Insider: Python 2.7.18, the last release of Python 2". *Python Insider*. Archived from the original on 26 April 2020. Retrieved 27 April 2020.
- [8]. Dean, Jeff; Monga, Rajat; et al. (9 November 2015). "TensorFlow: Large-scale machine learning on heterogeneous
- [9]. Piatetsky, Gregory. "Python eats away at R: Top Software for Analytics, Data Science, Machine Learning in 2018: Trends and Analysis". *KDnuggets*. *KDnuggets*. Archived from the original on 15 November 2019. Retrieved 30 May 2018.
- [10]. "Who is using scikit-learn? — scikit-learn 0.20.1 documentation". *scikit-learn.org*. Archived from the original on 6 May 2020. Retrieved 30 November 2018.
- [11]. Jouppi, Norm. "Google supercharges machine learning tasks with TPU custom chip". *Google Cloud Platform Blog*. Archived from the original on 18 May 2016. Retrieved 19 May 2016.
- [12]. J. T. Camillo Lugaresi, "MediaPipe: A Framework for Building Perception Pipelines," 2019, <https://arxiv.org/abs/1906.08172>.
- [13]. V. Bazarevsky and G. R. Fan Zhang. *On-Device, MediaPipe for Real-Time Hand Tracking*.
- [14]. K. Pulli, A. Baksheev, K. Korniyakov, and V. Eruhimov, "Realtime computer vision with openCV," *Queue*, vol. 10, no. 4, pp. 40–56, 2012.

[15]. Pulli, Kari; Baksheev, Anatoly; Korniyakov, Kirill; Eruhimov, Victor (1 April 2012). "Realtime Computer Vision with OpenCV". *Queue*. 10 (4): 40:40–40:56. doi:10.1145/2181796.2206309.

[16]. UNWOMEN (November 2020) Facts and figures: Ending violence against women, Available at: (Accessed: 13 March 2021).

[17]. D.-S. Tran, N.-H. Ho, H.-J. Yang, S.-H. Kim, and G. S. Lee, "Real-time virtual mouse system using RGB-D images and fingertip detection," *Multimedia Tools and Applications*, vol. 80, no. 7, pp. 10473–10490, 2021.

View at: [Publisher Site](#) | [Google Scholar](#)

[18]. A. Haria, A. Subramanian, N. Asokkumar, S. Poddar, and J. S. Nayak, "Hand gesture recognition for human computer interaction," *Procedia Computer Science*, vol. 115, pp. 367–374, 2017.

View at: [Publisher Site](#) | [Google Scholar](#)

[19]. K. H. Shibly, S. Kumar Dey, M. A. Islam, and S. Iftekhar Showrav, "Design and development of hand gesture based virtual mouse," in *Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, pp. 1–5, Dhaka, Bangladesh, May 2019.

View at: [Publisher Site](#) | [Google Scholar](#)

Appendix A: Disclaimer statement

This report was written by students at the Telecommunication Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.