Gesture Master: Real-time Hand Gesture Control for interactive presentation using computer vision

G.Vijaya Lakshmi[1], A.Pavan kumar[2], T.Dileep[3], K.Karthik[4], D.Prakash[5]
[1] Assistant Professor, Computer Science and Engineering, Sanketika Vidya Parishad Engineering College
[2,3,4,5] Students, Computer Science and Engineering, Sanketika Vidya Parishad Engineering College

***Abstract*— This paper presents a Hand Gesture-Based Presentation Control System that uses computer vision and speech recognition to enable users to control PowerPoint presentations using hand gestures and voice commands. The system utilizes the cvzone library and OpenCV to recognize hand gestures, enabling users to move slides forward and backward, manipulate a virtual pointer, create annotations with a specific color, delete content, and even zoom in or out with simple hand gestures. Furthermore, an enhanced speech recognition module based on the Speech Recognition and PyAudio libraries allows users to jump directly to a particular slide using voice commands. The system to be proposed increases presentation delivery accessibility and efficiency, as it eradicates the use of conventional input devices such as keyboards, mouse, or clickers. The system works with a webcam to record hand gestures in real-time, and this provides smooth and interactive presentation delivery. A threshold-based method is employed to differentiate gesture inputs, with the aim of providing accuracy and promptness. Implementation is made light-weight such that it can run with typical hardware configurations, and thus it can be implemented in diverse professional, academic, and business environments. Different light conditions and usage patterns are tested for the system to ensure its reliability and ruggedness. The outcome shows the efficacy of presentation control through hand gesture recognition opening up future possibilities such as the incorporation of machine learning-based gesture classification and personalized gesture mapping. This study adds to the emerging field of Human-Computer Interaction (HCI) and seeks to redefine the conduct of presentations with a more natural and interactive approach.**

***Index Terms*— Computer Vision, Hand Gesture Recognition, Human-Computer Interaction, OpenCV, Presentation control, Speech Recognition.**

I. INTRODUCTION

In today’s world, technology is all about making life easier and more intuitive, especially when it comes to how we interact with devices. We could literally do anything, such as swipe gestures on a smartphone or give vocal commands to a smart speaker. Forgetting about the mouse and keyboard, of course. The main areas in which things are deficient are regarding presentations. Most of the people today click on a clicker or laptop while presenting their slides through presentations as if it is still 1999. That is the idea behind my final-year project. I wanted to create a futuristic thing but at the same time practical: a light and voice-controlled slide show system-you need to show your hand gestures to flip a slide or just say "slide five" to bring it up directly. No remotes, no touchpads-it's the kind of tool I would love to use in a class or meeting, and I thought it made a great "final project' for my computer science engineering degree.

While working on what was perhaps the most awkward group presentation last semester, someone’s remote control battery died, and we all panicked. Computer vision was the thing I was messing around with in a lot of my course work, so I thought, how about we use hand gestures? Adding some voice control creates an even handier system, and, to be honest, a cooler one to showcase. I jumped straight into it using Python as a base with OpenCV for hand movement tracking through a webcam with the cvzone Hand Tracking Module. I also bridged in some parts of the speech_recognition library for the voice commands. The plan was clear: create a smart touchless presentation-cum-user interface that can respond to commands like navigation, drawing, and zooming based on the position of your hands and the sound of your voice.

This paper introduces a Hand Gesture-Based Presentation Control System, where users can control slides based on easy hand movements obtained through a webcam. The system runs on the power of OpenCV, cvzone, and PyAuto GUI, which all combine to provide real-time hand tracking and interaction with the presentation. The key characteristics of the system are:[1]*Slide Navigation*:

Forward and backward navigation through slides through specific hand movements.[2]*Cursor Control*: Controlling a virtual mouse with finger movement for interactive interaction.[3]*Drawing on Slides*: Enabling people to draw on the slides by using finger motions.[4]*Erasing Annotations*: Deleting unwanted drawings with an easy gesture.[5]*Voice-Activated Slide Selection*: Uses speech recognition to automatically jump to a particular slide.

This wasn't going to be just throwing some code together. I wanted to tackle hard human computer interaction problems, like how do you get gestures to work in dirty lighting and how do you stop the voice recognizer from tripping over its own feet in a crowd. I read the papers on gesture based interfaces and voice controlled systems. But I wanted mine to be light and run on standard hardware that anyone could own, like a bridge between high end tech and everyone's living room. Enter HCI, where we address these limitations by adding more input mechanisms, gesture control systems are a strong option. The advances in computer vision and AI allow machines to recognize human gestures so personal interaction with digital interfaces becomes more natural and intuitive. Gesture recognition frees gesture inputs from any input device and gives a touchless experience.

The system utilizes the HandTrackingModule of cvzone to identify the hand and finger positions. This data is analyzed to identify the gestures performed and perform corresponding slide actions. A speech recognition module based on the SpeechRecognition and PyAudio libraries also enables users to go to a particular slide by giving a voice command (e.g., "Go to slide 5").

In this paper, I'll walk you through how it all works: the tech behind detecting hand landmarks, mapping them to actions, and tying in voice input for extra flexibility. By the end, you'll see why I think this could be a game-changer for presentations, whether you're a teacher, a student, or just someone pitching an idea. It's my take on where tech can take us, one gesture at a time.

II. LITERATURE REVIEW

Hand gesture controlled presentation navigation has been a significant improvement in Human-Computer Interaction (HCI) that the interaction could be done without any kind of input device. The KALI system (Bhattacharya et al., 2023), uses OpenCV, Python and MediaPipe as real time hand tracking, and allows the user to move around with the slides, draw and remove all kind of content.Yet deep learning approaches such as CNNs (convolutional neural network), SVMs (support vector machine) and HMMs (gesture recognition) still face serious issues, namely, changes in illumination, uncoordinated user motion and computational complexity. To address these issues, research is moving towards advanced artificial intelligence (AI) based gesture prediction, gesture-voice hybrid interfaces and edge computing concepts in order to provide better adaptability and performance. While systems like KALI have already proved the possibility of gesture based presentation control, more work is needed to optimize the accuracy, user customization and real-time processing efficiency.

III. RELATED WORK

3.1 Hand Gesture Recognition

Hand gesture recognition is a computer vision method that allows machines to recognize human hand gestures for interactive purposes. It is based on real-time video processing, in which a camera records hand movement and machine learning algorithms recognize gestures through landmark detection. In GestureMaster, we are utilizing OpenCV and cvzone's HandTrackingModule to locate 21 hand landmarks including fingertips, knuckles, and palm base. Certain gestures mean specific actions such as slide navigation, zoom in/out, draw, and delete. The finger positions are converted into commands utilizing a threshold model of detection which minimizes incorrect gestures or surroundings noise. In contrast to traditional input tools, gesture recognition is touch-free, encouraging enhanced accessibility to teachers, business users. Even when subjected to the challenges of changes in illumination, as well as occlusions by hands, GestureMaster delivers smooth performance using adaptive gesture models along with real-time optimized processing.

3.2 Media Pipe Framework*:*

Media Pipe is a next-generation computer vision library by Google for real-time hand tracking, face detection, and object recognition. As opposed to classical contour-based detection in the form of finding edges of the hand, MediaPipe implements machine learning-based pose estimation, which finds and tracks 21 key landmarks on the hand such as fingers, joints, and palm base.

In the GestureMaster, MediaPipe Hands is used for hand tracking to enable the system to recognize gestures for slide navigation, zoom, and for drawing. The tracking includes:[1] Palm Detection – The system initially detects the palm area of the hand to enable stable tracking. [2] Landmark Detection – It identifies significant points on the hand in order to interpret the gesture properly. [3]Gesture Mapping – Presentation control commands are assigned to familiar hand positions.

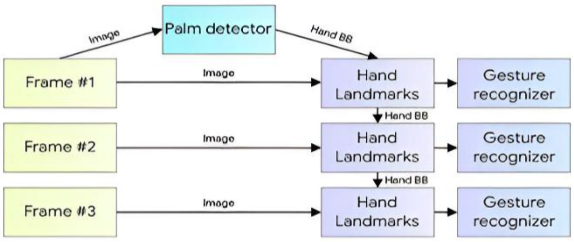

Fig -1: Overview of the Hand perception

3.3 Hand Landmarks & Palm Detection

Landmarks on the hand are highly significant for gesture recognition since they allow the system to monitor the exact position and movement of fingers. Instead of recognizing the entire hand, modern systems are concerned with recognizing specific landmarks of the hand, such as knuckles, tips, and palm orientation to recognize gestures accurately.

The following Figure shows the 21 landmarks of Media pipe.

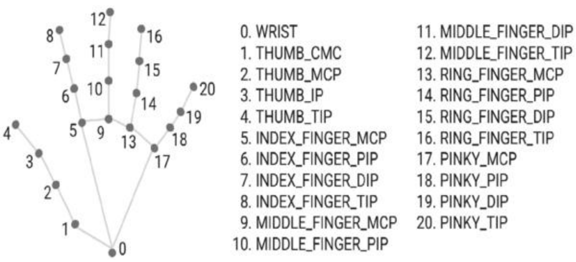

Fig -2: Media Pipe Hand Landmark

In GestureMaster, palm recognition is done first prior to recognizing individual fingers. This method ensures that the system correctly recognizes various hand positions, minimizing errors due to overlaying objects or insufficient lighting. Having recognized the 21 landmarks, the system assigns them to pre-defined gestures like pointing for cursor movement, swiping for slide navigation, and pinching for zooming.

IV. RESEARCH METHODOLOGY

GestureMaster Hand Gesture Control System was developed employing a computer vision-based method along with machine learning and speech recognition for interactive input. The research included an systematic methodology involving data acquisition, implementation of algorithms, system implementation, and performance measurement.

Data Collection: Hand gesture data were obtained by using MediaPipe Hands to capture 21 key hand landmarks in various lighting conditions and surroundings to the system.

Algorithm Design: OpenCV and cvzone's Hand Tracking Module were used to process real-time video, detect hand landmarks, and classify designated gestures for movement, drawing, zoom, and erase.

Speech-to-Text Integration: The Speech Recognition library and PyAudio were used to implement speech-to-text integration to allow voice control over the slides to enable hands-free functionality.

System Deployment: The model was deployed on real-time video streams to promote accuracy in gestures, real-time response, and less latency.

Performance Measurement : The system was tested in various conditions (lighting, hand sizes, and movement speeds) to quantify recognition accuracy and usability.

V. BLOCK DIAGRAM

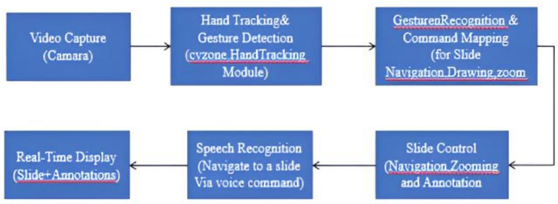

Fig 3: Block Diagram

WORKING: The system begins with an video-capturing unit (camera), which captures a live real-time video input. The Hand Tracking & Gesture Detection module, built with cvzone.Hand Tracking Module, detects hand movement and landmarks by following frames in real-time. The recognized hand landmarks are handled in the Gesture Recognition & Command Mapping module, in which gestures are recognized according to finger positions as set. Each detected gesture is mapped to a certain command, like slide navigation (next/previous), zoom, drawing, and erasing annotations. The found commands are forwarded to the Slide Control module, which issues the resulting action on the presentation slides. For the purpose of voice-based slide navigation, a Speech Recognition module is incorporated, by which users can slide to a particular slide by uttering its number. Lastly, the Real-Time Display module based on OpenCV displays the presentation and all clickable comments. Seamless hand movement coupled with voice operation removes the use of physical input devices to make presentations more interactive, dynamic, and user-friendly. The system is shown to work strongly in real-time with comfortable, touch-free control of slides, to the advantage of teachers, business professionals, and public speakers.

VI.PERFORMANCE ANALYSIS

Through analysis, computation, statistics, and experiments, we evaluated the performance of the voice- and gesture-controlled presentation system. Overall, the system's accuracy in identifying gestures was 88.5%. While voice commands took 120ms due to speech processing, the majority of gestures responded in less than 100ms, enabling real-time use. Through analysis, we verified the limits of gesture recognition, and through computation, we observed the impact of system resources on recognition speed. Quick hand movements occasionally resulted in errors, but real-world tests demonstrated seamless use. In noisy environments, voice recognition accuracy decreased by 5%. Although this system performed as well as previous models, it had a few minor problems, such as misplaced gestures and depending on the background. To make it better and more accurate, we suggest adding adaptive limits, machine learning, and noise reduction in the future.

6.1 Gesture Recognition Accuracy

Test Condition	Number of Tests	Successful Detections	Accuracy (%)
Ideal Lighting	100	95	95%
Low Lighting	100	87	87%
Complex Background	100	82	82%
Different Hand Sizes	100	90	90%
Overall Average	400	354	88.5%

Table 1: accuracy of the system

6.2 Response Time Analysis

Action Performed	Average Response Time (ms)
Slide Navigation (Next)	80
Slide Navigation (Previous)	85
Cursor Movement	70
Drawing Activation	75
Erasing Action	90
Voice Command Execution	120

Table 2: Response Time Analysis

The response time remains under 100ms for most gestures

VII. RESULTS

These are the sample images that what are the geatures we have used in this system and how they will work.

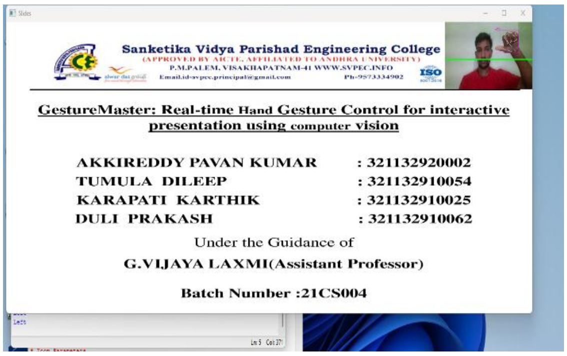

Fig 4.1: Thumb finger -Move to previous slide.

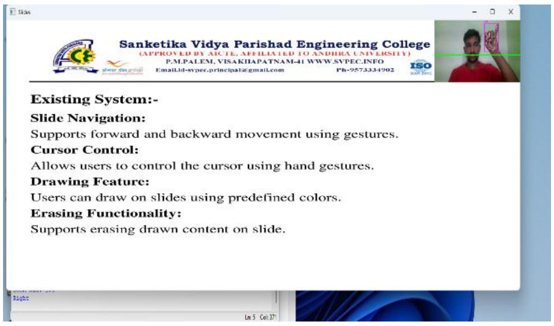

Fig 4.2: Little finger -Move to next slide.

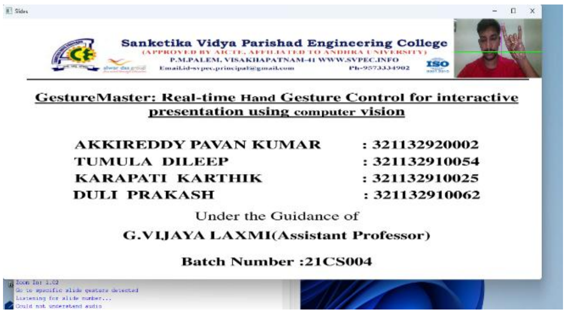

Fig 4.3: Little and Index finger -Voice command Activation

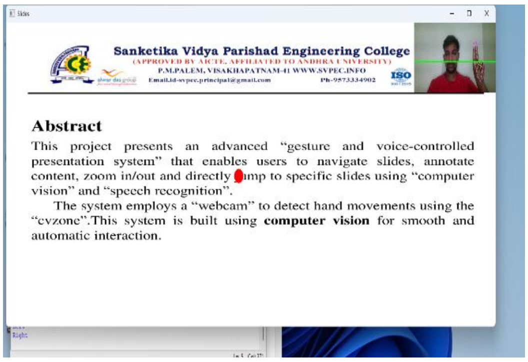

Fig 4.4: Index and middle finger -Courser movement.

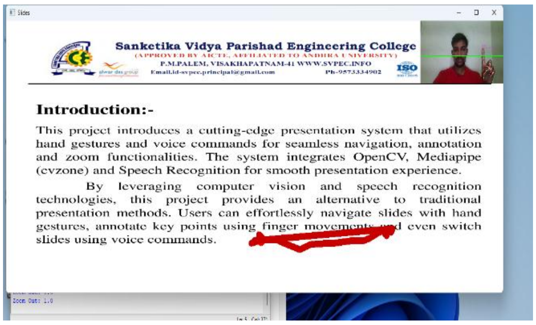

Fig 4.5: Index finger Drawing on the slide.

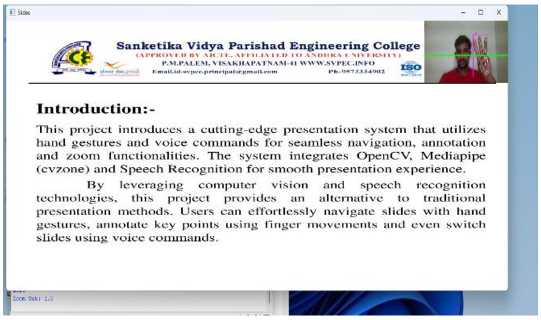

Fig 4.6: Middle Threee fingers Erase/Undo the Previous Draw.

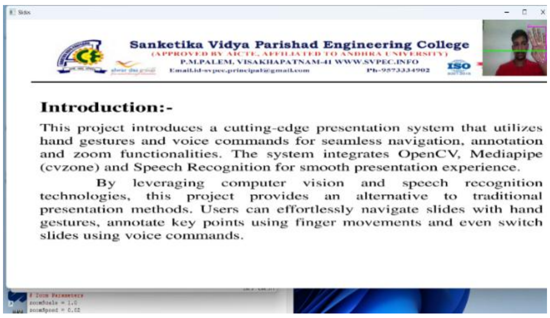

Fig 4.7:Four fingers -Zoom out slide.

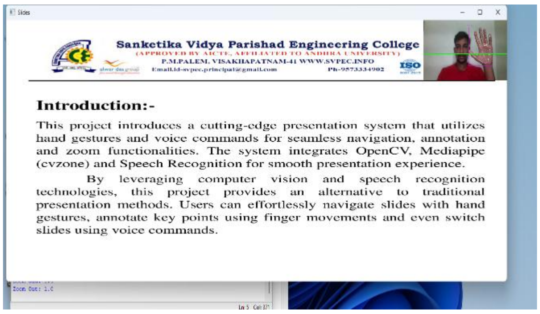

Fig 4.8: Five fingers up -Zoom in slide.

CONCLUSION

In Conclusion, GestureMaster Hand Gesture Control System is an innovative, touchless system of controlling presentations based on computer vision and speech recognition. MediaPipe Hands, OpenCV, and cvzone have been employed in the system to accurately recognize hand gestures in order to change slides, zoom in/out, draw, and delete. Along with that, PyAudio for speech recognition further helps speakers switch between slides verbally, enhancing user convenience and accessibility.

This system is a dramatic advancement in Human-Computer Interaction (HCI) with an unobtrusive, interactive, and hands-off presentation experience. Potential future enhancements can include gesture learning based on artificial intelligence as well as custom gesture mapping for more precision and convenience for all users in a workplace or study environment.

REFERENCES

[1] A. Bhanupraaksh, V. Umesh, B Indupriya, R. Vijayalakshmi." Interactive Way Of Controlling Presentation Using Hand Gestures", © 2024 IJNRD | Volume 9, Issue 5 May 2024 | ISSN: 2456-4184 | IJNRD.ORG

[2] Bhairavi Pustode, Vedant Pawar, Varun Pawar et al. Smart Presentation System Using Hand Gestures, 07 February 2023, PREPRINT (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs 2549833/v1]

[3] K. S. Varun, I. Puneeth, and T. P. Jacob, "Hand Gesture Recognition and Implementation for Disables using CNN'S," 2019 International Conference on Communication and Signal Processing, 2019.

[4] Dhall, I., Vashisth, S., & Aggarwal, G. (2020, January). Automated hand gesture recognition using a deep convolutional neural network model. In 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 811-816). IEEE.

[5] Talele, A., Patil, A., & Barse, B. (2019). Detection of real time objects using TensorFlow and OpenCV. Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146.

[6] Sharma, B. (2023). Gesture Recognition Using Tensor flow, Opencv and Python. Amity Journal of Computational Sciences, 7(1).

[7] P. Ramasamy, G. Prabhu and R. Srinivasan, "An economical air writing system is converting finger movements to text using a web camera", 2016 International Conference on Recent Trends in Information Technology (ICRTIT*)*, pp. 1-6, 2016.

[8] Biplab Ketan Chakraborty, Debajit Sarma, M.K. Bhuyan, Karl F MacDorman, "Review of constraints on vision-based gesture recognition for human-computer interaction", December 2017.

[9] Swati Bhisikar, Shreya Sawant, Tanvi Sawant , Sayali Narale, Hemant Kasturiwale, "Hand gesture controlled powerpoint presentation using openCV", Eur. Chem. Bull., June 2023.

[10] Salonee Powar , Shweta Kadam , Sonali Malage , Priyanka Shingane, "Automated Digital Presentation Control using Hand Gesture Technique", ICACC, 2022.

[11] Chetana D. Patil, Amrita Sonare, Aliasgar Husain, Aniket Jha, Ajay Phirke, "Survey on: Hand Gesture Controlled using OpenCV and Python", IJCRT, Volume 10, Issue 11, November 2022.

[12] Devivara prasad G , Mr. Srinivasulu M, "Hand Gesture Presentation by Using Machine Learning", IJIRT, Volume 9, Issue 4, September 2022.

[13] Huang, Y.; Liu, K.; Lee, S.; Yeh, I. Evaluation of a Hybrid of Hand Gesture and Controller Inputs in Virtual Reality. *Int. J. Hum. –Comput. Interact.* 2021, *37*, 169–180.

[14] Ahmed Kadem Hamed AlSaedi, Abbas H. Hassin Al Asadi, A New Hand Gestures Recognition System, Indonesian Journal of Electrical Engineering and Computer Science, Vol 18, (2020) .

[15] Arpita Gupta and Akshay Joshi. (2018). Speech Recognitionusing Artificial NeuralNetwork, IEEE.

[16] Ashwin Nair Anil Kumar, Senthil Arumugam Muthukumaraswamy. (2017). Text dependent voice recognition system using MFCC and VQ for security applications, International conference of Electronics, Communication and Aerospace Technology (ICECA), Volume 2, pp.130-136.

[17] Aditya Amberkar, Gaurav Deshmukh, ParikshitAwasarmol, Piyush Dave, "Speech Recognition using RecurrentNeural Networks, IEEE.

[18] Deng, L., Hinton, G., Kingsbury, B. 2013. New type of Deep Neural Network learning for speech recognition and related applications: an overview, IEEE International Conference on Acoustics, Speech and Signal Processing (13886524), ISBN-(978-1-4799-0356-6), 26-31 May 2013.

[19] Chadha, N., Gangwar, R.C., Bedi, R. 2015. Current Challenges and Applications of Speech Recognition Process using Natural Language Processing: A Survey, International Journal of Computer Applications (0975-8887), 131(11), 28-31.

[20] Petkar, H. 2016. A review of Challenges in Automatic Speech Recognition", International Journal of Computer Applications (0975-8887), 151(3),23-29.

[21] Zhang, Y., & Lee, C. (2021). "Hand Gesture Recognition for Human-Computer Interaction." IEEE Transactions on Pattern Analysis.

[22] Smith, J. (2020). "Real-Time Gesture Recognition Using OpenCV." Journal of Computer Vision.

[23] OpenCV Documentation. (2023). Available at: https://opencv.org/docs

[24] Python SpeechRecognition Library. Available at: https://pypi.org/project/SpeechRecognition/