Implementation of Gesture-Based Virtual Keyboard and Mouse

[1]Prof. S. A. Nagtilak,[2]Kranti Bhosale,[3]Labhesh Chaudhari,[4]Prathmesh Chaudhari,[5]Mansi Ingale
[1]Assistant Professor, [2,3,4,5]Students
[1,2,3,4,5]Information Technology / Smt. Kashibai Navale College of Engineering
[1,2,3,4,5]Savirtibai Phule Pune University/India

Chapter 1. Abstract

The development of human-computer interaction has seen a major change in the last few years, moving in the direction of more user-friendly and effective interfaces. This tendency has prompted researchers to investigate new input techniques, especially in the field of gesture-based control systems. By utilizing both hand and eye gestures to create a gesture-based virtual mouse and keyboard, this paper introduces a novel method of human-computer interaction. The suggested system offers users a smooth and simple way to interact with computers by combining hand gesture recognition and eye-tracking technology. Users can control cursor movement and keyboard inputs without the use of physical peripherals by using the system to track their hand and eye movements and interpret them as commands.
The incorporation of eye-tracking technology improves the system's accuracy and responsiveness, enabling more precise cursor control and interaction with elements displayed on the screen. Hand gesture recognition, on the other hand, gives users an ergonomic and natural way to input data, allowing them to carry out commands with straightforward motions. The architecture of the system combines real-time processing capabilities to guarantee low latency and high responsiveness with sophisticated machine-learning algorithms for gesture recognition. Additionally, users can define their gesture commands according to their needs and preferences because the system is flexible and customizable. The suggested gesture-based virtual mouse and keyboard system has undergone rigorous testing and evaluation, and the results show promising performance and usability in a range of scenarios and applications. This novel method of interacting with computers has the potential to improve user experience, accessibility, and productivity in a variety of computing contexts.

Keywords: Human-Computer Interaction, Gesture Recognition, Eye-Tracking

Chapter 2. Introduction

The touchless virtual keyboard system represents a cutting-edge innovation in human-computer interaction, offering a revolutionary approach to text input that transcends traditional keyboard input methods. In today's rapidly evolving technological landscape, where the demand for seamless and intuitive user interfaces continues to rise, the development of touchless input solutions has emerged as a promising frontier in enhancing accessibility, convenience, and hygiene in computing environments. This comprehensive introduction delves deep into the intricacies of touchless virtual keyboards, exploring their evolution, underlying technologies, applications across diverse industries, potential impact on society, and prospects. At the core of the touchless virtual keyboard system lies a fusion of advanced computer vision, machine learning, and human-computer interaction techniques, all meticulously orchestrated to enable users to interact with digital interfaces using natural hand gestures and movements. By harnessing the power of real-time hand tracking and gesture recognition algorithms, coupled with sophisticated eye detection mechanisms, touchless virtual keyboards empower users to navigate and manipulate digital content with unprecedented ease and precision. Gone are the days of cumbersome physical keyboards and cumbersome mouse input; instead, users can seamlessly compose text, navigate menus, and execute commands simply by gesturing in the air, ushering in a new era of touchless computing.

The journey towards the realization of touchless virtual keyboards has been characterized by a convergence of technological advancements and interdisciplinary collaboration. From the early experiments in gesture-based computing to the advent of sophisticated deep learning models for hand and eye tracking, each milestone has contributed to the evolution of touchless input solutions. Today, with the advent of high-resolution webcams, powerful processors, and robust software frameworks, touchless virtual keyboards have transitioned from experimental prototypes to practical tools with tangible applications across a myriad of domains.In the realm of accessibility, touchless virtual keyboards hold immense promise for individuals with physical disabilities, providing them with a lifeline to digital communication and interaction. By removing the physical barriers associated with traditional input devices, such as keyboards and mice, touchless virtual keyboards empower users with mobility impairments to express themselves, communicate with others, and access digital resources with unparalleled independence and dignity. Moreover, for individuals with conditions such as arthritis or carpal tunnel syndrome, which may limit their ability to use conventional input devices, touchless virtual keyboards offer a welcome respite, alleviating discomfort and facilitating seamless interaction with digital devices.

Beyond accessibility, touchless virtual keyboards have found a fertile ground for innovation and experimentation in a myriad of industries and sectors. In the realm of healthcare, for instance, touchless virtual keyboards have emerged as invaluable tools for healthcare professionals, enabling them to input patient data, access medical records, and control diagnostic equipment without the need for physical contact, thereby minimizing the risk of cross-contamination and infection transmission. Similarly, in the field of education, touchless virtual keyboards have revolutionized the learning experience, allowing students to participate in interactive lessons, collaborate on projects, and engage with educational content in an immersive and intuitive manner. Moreover, in the realm of entertainment and gaming, touchless virtual keyboards have opened up new vistas of creativity and interactivity,enabling gamers to control characters, navigate virtual environments, and execute complex maneuvers using nothing but their hands and gestures. From virtual reality (VR) experiences that transport users to fantastical realms to augmented reality (AR) applications that blur the lines between the digital and physical worlds, touchless virtual keyboards have become indispensable tools for game developers and enthusiasts alike, offering unparalleled immersion and engagement.

Chapter 3. Literature Survey

J. Shin and C. M. Kim, [1] claimed that a large number of experiments have been conducted on text input systems that use image-based hand gesture detection. Nevertheless, there are certain issues with hand gesture languages being widely used, such as finger alphabets, sign languages, and aerial handwriting, as discussed in the earlier studies. Writing and recognizing aerial handwriting takes a lot of time. The number of people who can use finger alphabets and sign language is limited because they require a significant amount of practice and education to use. This study suggests a new character input method that can be used to improve human-computer interaction. It is based on hand-tapping movements for both English and Japanese hiragana characters. The hand-tapping gestures are motions for hands to tap keys on virtual keypads in the air. Anyone, including those with hearing impairments, can utilise these gestures as an efficient way to use hand alphabets. When writing in hiragana, the consonant portion of the character and the aerial virtual keypad are determined by the hand that is used to press the key and the number of fingers that are stretched. To enter a character, simply tap the key on the virtual keypad that corresponds to the intended vowel. We employ a key layout that is akin to the English and Japanese flick keyboards seen on smartphones, so anyone can use these hand-tapping movements with just a quick explanation. With this non-touch input technology, which uses the Kinect sensor alone—no keyboard, mouse, or body-worn devices—users may interface with computers efficiently. We anticipate that a new avenue for human-computer connection will be opened by this character input method.

L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua [2] stated that Static or dynamic, hand gestures are a hot topic in research and have several applications in real-time systems for human-computer interaction. Basic methods of interacting with computers are through hand gestures, both static and dynamic. This work proposes an approach to the hand-gesture recognition-based text input mechanism. This portable hand-operated text input system is intended for use with virtual reality (VR) and augmented reality (AR) gadgets. A standard camera takes a picture of the hand to identify and categorize hand gestures. Following background subtraction, the hand is segmented, and the segmented hand gesture is fed into the trained neural network for gesture recognition. Lastly, a convex hull algorithm is used to track and record hand movements. A neural network that has been trained is given the matching written character. After testing the suggested architecture and comparing the experimental findings with those from other approaches, it was found that the suggested approach outperformed conventional approaches and achieved an accuracy of 96.12%, which is an improvement over current approaches overall.

C. Li, C. -K. Kim and J. -S. Park [3] This paper describes the indirect interface system in which general users assign computer instructions just through gaze tracing, without a mouse or keyboard. We use the Web camera to replace the computer input system. The face region and the eye region were extracted based on the Haar classifier implemented on the open-source computer vision library (OpenCV). It controls mouse-moving by automatically affecting the position where eyesight focuses on, and simulates mouse-click by affecting blinking action. We use the virtual keyboard displayed on the monitor to simulate the keyboard entry. As a result, more than 95% of the tracing accuracy was achieved when the size of a single key is larger than 25 pixels.

CHAVALI, E Sankar. [4]. Virtual Mouse Using Hand Gesture. Recent improvements in gesture detection and hand tracking have brought about both opportunities and challenges. Try out a few of these options while outlining challenges and exciting future possibilities for virtual reality and user engagement. Given the popularity of COVID-19, the goal of this research is to reduce interactions between people and the reliance on technology to operate computers. These results will motivate additional research and, in the long run, support the use of virtual environments. In the proposed era, there are no such restrictions and gesture recognition may be used as a substitute. It may be possible to click and drag objects during this adventure using a variety of hand actions. The suggested project's input technique will only require a camera. The languages Python and OpenCV.

Y. Zhang, W. Yan and A. Narayanan [5] developed a brand-new virtual keyboard that enables users to text on any surface, on any kind of device. Customized and printed on simple paper, the virtual keyboard can be affixed to a wall or mounted on any oblique plane. Then, leveraging the fingertip location and hand skin tone, the device camera is employed for key recognition. The software will recognize a key as an input if the fingertip stays on it for a set period. The results of the tests demonstrate how several personalized virtual keyboards enable users to enter text without noticeable performance compromises, provided they have a physical keyboard in front of them. The total recognition rate of all inputs is 94.62% (true positives divided by all samples). Under natural illumination, keyboard (a) has the best average input recognition rate of 97.7%, whereas under lamplight, keyboard (b) has the worst average input recognition rate of 90.7%.

S. R. Chowdhury, S. Pathak and M. D. A. Praveena [6] Nowadays computer vision has reached its pinnacle, where a computer can identify its owner using a simple program of image processing. In this stage of development, people are using this vision in many aspects of day-to-day life, like Face Recognition, Color detection, Automatic cars, etc. In this project, computer vision is used in creating an Optical mouse and keyboard using hand gestures. The camera of the computer will read the image of different gestures performed by a person's hand and according to the movement of the gestures the Mouse or the cursor of the computer will move, even perform right and left clicks using different gestures. Similarly, the keyboard functions may be used with some different gestures, like using a finger gesture for alphabet selection and a four-figure gesture to swipe left and right. It will act as a virtual mouse and keyboard with no wire or external devices. The only hardware aspect of the project is a webcam and the coding is done in Python using the Anaconda platform. Here the Convex hull defects are first generated and then using the defect calculations an algorithm is generated and mapping the mouse and keyboard functions with the defects. Mapping a couple of them with the mouse and keyboard, the computer will understand the gesture shown by the user and act accordingly.

C. Topal, B. Benligiray and C. Akinlar [7] Virtual keyboards are useful tools, that ease the effortless entry of textual data and enable typing with alternative input hardware such as a single switch. Early virtual keyboards are designed similarly to physical keyboards in terms of appearance. Since a physical keyboard is designed for tactile use; the usability of the first virtual keyboards comes up short in utilization with pointing devices. For this reason, more useful virtual keyboards are proposed with improvements in modal and functional properties. In this study, we examine a couple of design issues for virtual keyboards to provide their efficient utilization with pointing devices. We analyze the effects of visual key layouts on the performance of virtual keyboards in connection with the statistical properties of the target language's vocabulary. We also propose a virtual keyboard design for a comfortable text entry experience based on our observations.

D. E. V. Ligarreto and D. L. De Luise [8] Peripherals like keyboard and mouse are part of technological evolution. Over the years there have been new interaction alternates with computers. But those original devices are still an important part of the human-computer interaction. Many papers are focused on the scientific analysis of personality profiling and stress evaluation. By using the keyboard and mouse they can perform good approximations, but there is still some pending work to properly reapply it to measure learning levels. This paper presents a set of improved metrics to assess learning levels for an individual during a learning process in a virtual environment. The main contribution of this work is behavior modeling with no extra metrics than the ones derived from keyboard and mouse usage. The scope of this study covers the comparison and evaluation of the proposal against traditional metrics. Results indicate that original equations can be simplified with no significant loss in precision.

Chapter 4. Methodology

A) Haar cascade algorithm for eye detection:

Paul Viola and Michael Jones presented an efficient object detection technique in their paper "Rapid Object Detection using a Boosted Cascade of Simple Features" (2001) that uses Haar feature-based cascade classifiers [2]. This method is based on machine learning, and it involves training a cascade function with a large number of both positive and negative images. Next, it's applied to identify objects in additional pictures.

To train the classifier, the algorithm first requires a large number of positive images, or images with faces, and negative images, or images without faces. After that, we must extract its features, Haar features, as seen in the image. They resemble the convolutional kernel exactly. Each feature is a single value obtained by subtracting the sum of pixels under the white rectangle from the sum of pixels under the black rectangle.

Currently, each kernel's potential sizes and locations are utilized to compute a large number of features—roughly 16000 features. We must ascertain the total of the pixels beneath the white and black rectangles for every feature computation. They introduced the integral image as a solution. However, the majority of these features—of all the ones we calculated—are unimportant. The Adaboost algorithm is used to solve this problem. It eliminates up to 6000 features. We do this by applying every feature to every training image. It determines the optimal threshold for each feature to categorize the faces as positive or negative. According to the paper, 95% accuracy in detection is achieved with just 200 features. They used about 6000 features in their final setup. So now you take an image. Take each 24x24 window. Apply 6000 features to it. Check if it is face or not.

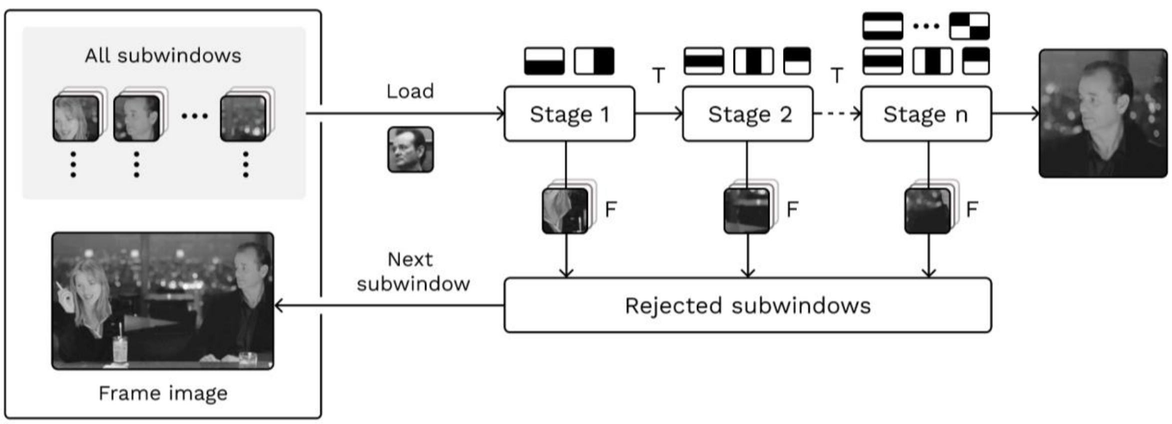

Fig 4.1 Harr Cascade Classifier [2]

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

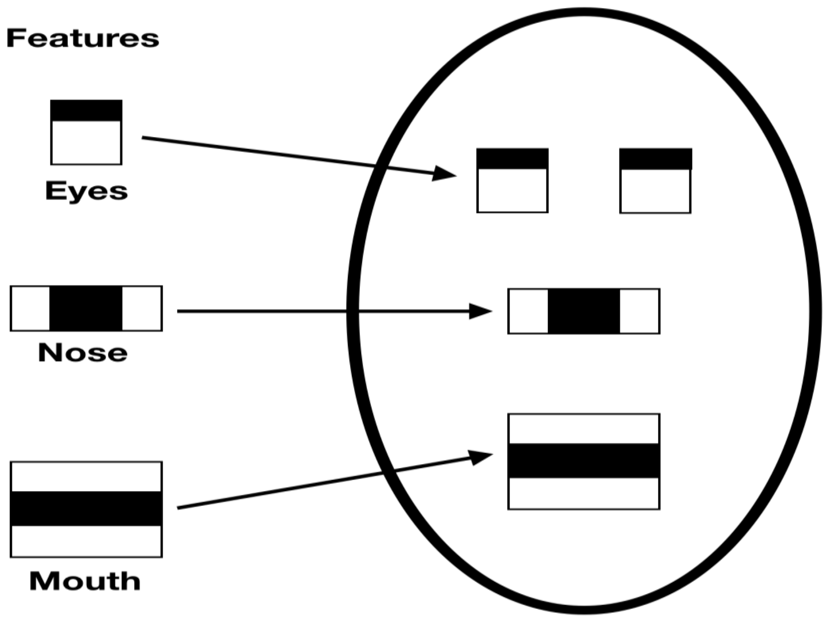

Fig 4.2 General Model of Face Features Extraction [2]

B) Hand gesture recognition for hand detection

Techniques for recognizing hand gestures usually entails multiple crucial phases. First, information is obtained using cameras or depth sensors that record hand movements in pictures or videos. After that, preprocessing methods such as segmentation to isolate the hand and noise reduction are used to improve the quality of the data. Then, pertinent features that include both temporal and spatial aspects of hand movements are extracted from the preprocessed data. These characteristics accurately depict various hand gestures.

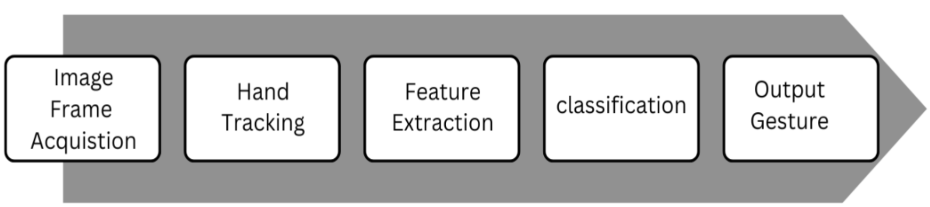

Fig 4.3 Block Diagram For Hand Gesture Recognition System

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

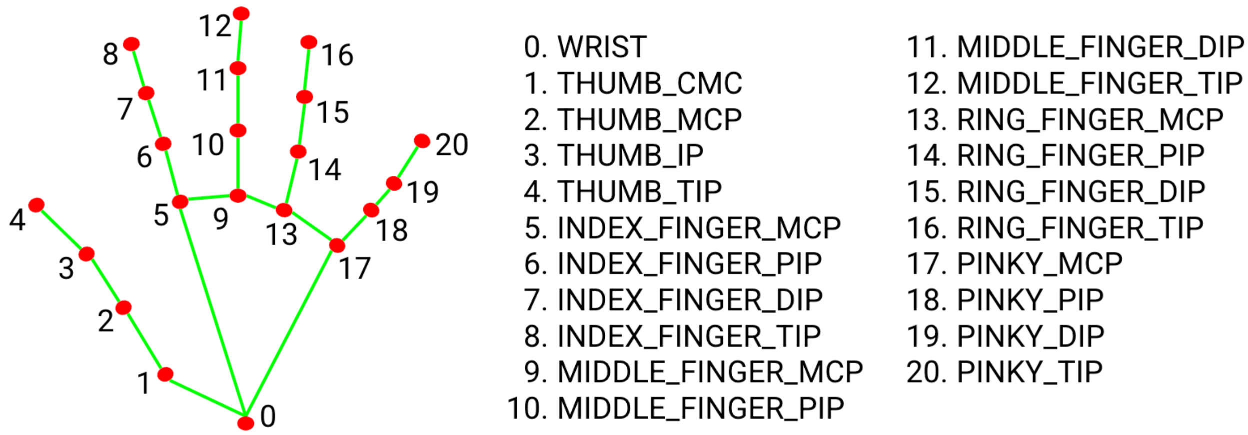

Fig 4.4 Hand Landmarks Detection Guide

Chapter 5. Implementation

A) Eye detection:

The process by which a camera detects eyeballs involves a series of sophisticated algorithms and techniques designed to analyze image data comprehensively. At its core, the camera relies on specialized eye detection algorithms that are adept at identifying and locating eyes within an image. One prevalent method is through Haar cascade classifiers, which are pre-trained machine learning models finely tuned to recognise specific features like eyes based on patterns of pixel intensity. These classifiers undergo rigorous training on extensive datasets containing positive and negative examples to learn the distinguishing characteristics of eye regions. Once the algorithm begins its analysis, it scans the image in various scales and positions, meticulously searching for patterns that resemble eyes. It identifies regions exhibiting similar characteristics, such as dark centers and lighter surroundings, and delineates them as potential eye locations. Subsequently, bounding dots are drawn around these identified regions to isolate the eye areas for further scrutiny. To minimize false positives, the algorithm conducts additional validation steps. It examines geometric constraints, ensuring that the detected regions align symmetrically and match typical eye sizes. Furthermore, contextual information, like proximity to other facial features, is taken into account to validate eye detection. Following this validation phase, the camera can engage in post-processing activities, leveraging the detected eye regions for various tasks. These tasks may include continuous eye tracking for gaze analysis or attention monitoring, biometric authentication through iris or retina scans, and image enhancement techniques tailored specifically for the eyes, such as adjusting exposure or focus. In summary, the camera's eye detection process is a multifaceted endeavor, combining advanced algorithms, validation procedures, and post-processing functionalities to accurately identify and utilize eye regions within images for a variety of applications.

1. **Eye Detection Algorithms:** Cameras use specialized algorithms to identify and locate eyes within an image. These algorithms analyze patterns, shapes, and colour variations to distinguish eye regions from the rest of the face.

2. **Haar Cascade Classifiers**: One common technique is using **Haar cascade classifiers**. These are pre-trained machine learning models that can recognise specific features (such as eyes) based on patterns of pixel intensity. The Haar cascade classifier is trained on a large dataset of positive and negative examples to learn what eye regions typically look like.

3. **Feature Extraction:** The algorithm scans the image in different scales and positions, looking for patterns that resemble eyes. It identifies regions with similar characteristics (e.g., dark centers and lighter surroundings) and marks them as potential eye locations.

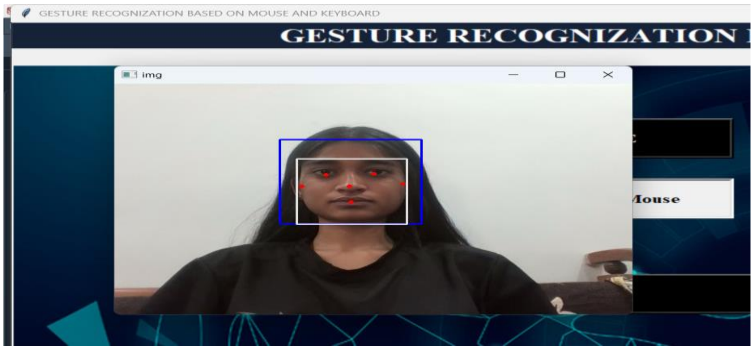

Fig 5.1 Face Detection And Eyes Extraction

4. **Validation and Refinement:** To reduce false positives, additional checks are performed. For instance:

-**Geometric Constraints:** Eyes are usually symmetrically positioned on the face, so the algorithm checks if the detected regions align correctly.

- **Contextual Information**: The algorithm considers the context (e.g., proximity to other facial features) to validate eye detections.

B) Hand detection:

The implementation of a touchless virtual keyboard system entails several key steps to ensure its functionality and user-friendliness. Firstly, upon successful login, users are presented with two primary options: accessing either the Virtual Mouse or Keyboard functionalities. Choosing the Virtual Keyboard option triggers the display of a keyboard interface directly on the desktop screen. This interface is meticulously designed to include only the necessary characters and letters, ensuring simplicity and ease of use. To interact with the virtual keyboard, the system leverages the webcam of the user's laptop or computer. Employing computer vision techniques, such as hand tracking and gesture recognition, the system detects the user's hand movements and interprets them as inputs for letter selection. As users point to each letter with their fingers, the system accurately determines the corresponding key and highlights it for visual confirmation. This real-time feedback mechanism enhances the user experience by providing immediate feedback on their selections. Selected letters are dynamically displayed in a small rectangular field below the virtual keyboard, allowing users to track their input progress effortlessly. Once the desired characters are selected and displayed, users have the flexibility to utilize the text for various purposes. This may include copying and pasting the text into a document, entering text into a text input field, or any other relevant action based on the system's functionality. The touchless nature of typing on the virtual keyboard, facilitated by hand gestures, offers a convenient and hygienic alternative to traditional keyboard input methods. This method of interaction is especially beneficial for individuals with physical disabilities and in environments where physical contact with devices may pose health risks, such as during a pandemic. Overall, implementing a touchless virtual keyboard system enhances accessibility and user experience, paving the way for seamless and intuitive computer interactions.

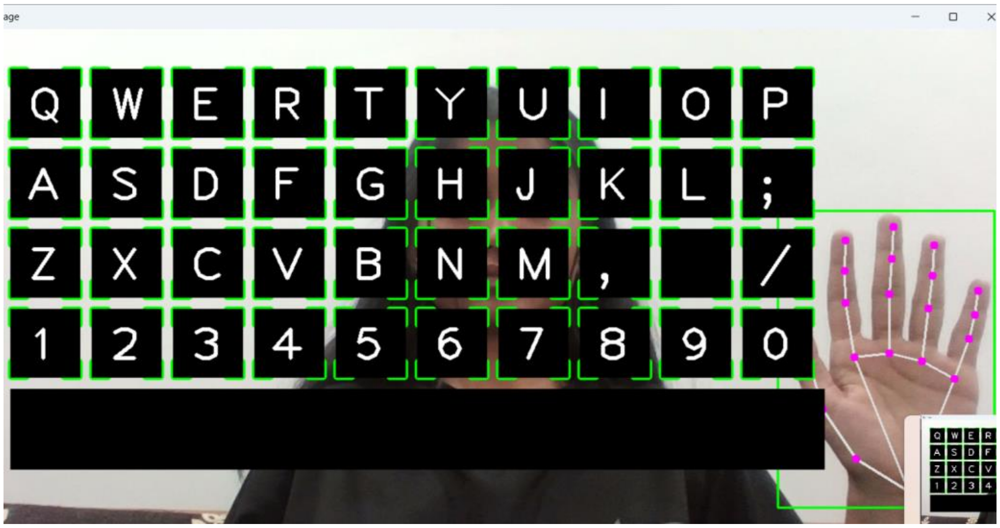

Fig 5.2 Hand And Landmarks Detection

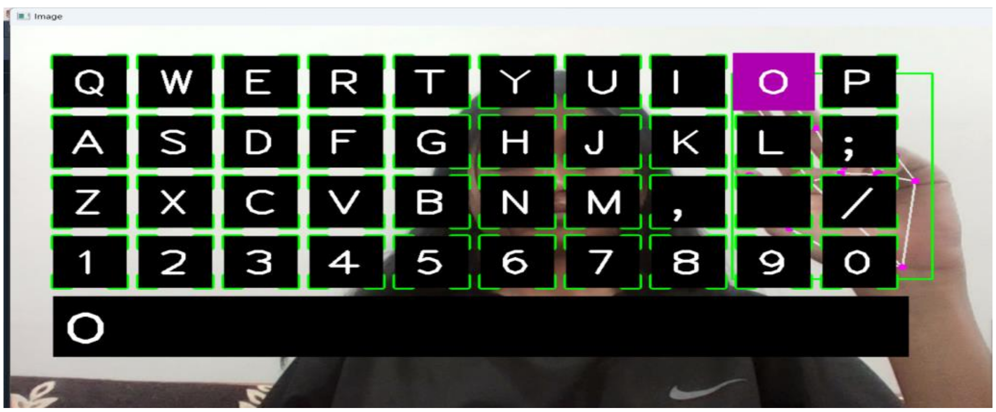

Fig 5.3 Detecting The Hand Movement And Recognizing The Character

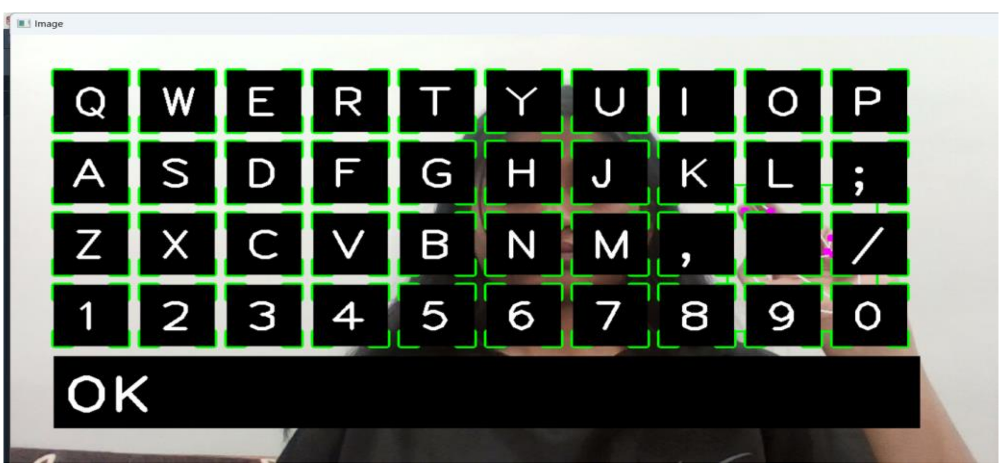

Fig 5.4 Displaying Selected Characters

Working of Keyboard

1) Initialization: Begin system setup and present login options to the user.
2) User Input Selection: Await user selection of the Virtual Keyboard option.
3) Virtual Keyboard Rendering: Display the virtual keyboard interface on the screen.
4) Hand Gesture Recognition: Activate webcam for detecting hand movements.
5) Letter Selection Process: Map fingertip position to corresponding keys on the virtual keyboard.
6) Display Selected Characters: Update the display to reflect the chosen letters.
7) Text Utilization: Offer options for using the selected text.
8) End of Interaction: Continue monitoring gestures until the user exits the interface.

These steps outline the systematic approach to implementing a touchless virtual keyboard system, focusing on user interaction, feedback mechanisms, and practical applications.

Chapter 6. Result

The touchless virtual keyboard system implementation is progressing smoothly, particularly in its functionality for hand and eye detection. Upon successful login, users are seamlessly guided to select the Virtual Keyboard option, which promptly displays the keyboard interface on the screen. Utilizing sophisticated hand gesture recognition, the system accurately tracks the user's hand movements in real-time, allowing them to effortlessly select letters and characters by pointing to them with their finger. The selected letters are promptly displayed below the virtual keyboard, providing users with immediate visual feedback on their input. This interactive process ensures a seamless and intuitive typing experience, enhancing accessibility for individuals with physical disabilities and offering a hygienic alternative to traditional keyboard input methods, especially in situations where physical contact with devices may pose health risks, such as during a pandemic. Overall, the integration of hand and eye detection functionalities into the touchless virtual keyboard system demonstrates its versatility and potential to revolutionize computer interaction paradigms, paving the way for enhanced accessibility and user experience across various industries and environments.

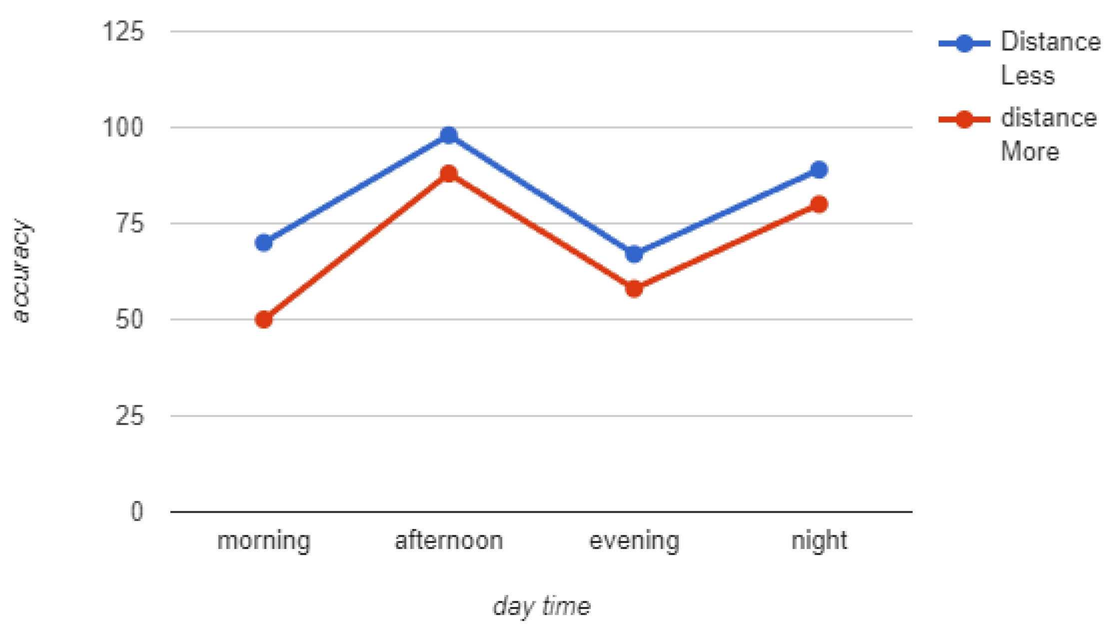

Fig 6.1 Day Time vs Accuracy Graph

The accuracy of the system practically should have been 100 percent, but external impacts such as intensity of light, an average camera, etc were the reason the accuracy little low, the accuracy of the system is about 70-85 percent after all the trials and error, in which the maximum average being 85 at broad daylight, whereas at night time the average accuracy decreases to about 75 percent. The output which is determined using this experiment, is how the cursor moves in that specific direction the cursor moves down, left, and right respectively. Accuracy of the system in different lighting conditions

Chapter 7. Future scope

With the use of the Haar cascade algorithm, it is possible to expand and improve the project of creating a virtual mouse and keyboard in the future. Continually improving the system's accuracy and precision is one way to make it better. To do this, the Haar cascade algorithm must be continuously improved to recognize better and track hand gestures and movements. Furthermore, increasing the number of gestures that are recognized can improve usability, which makes it necessary to incorporate machine learning techniques to identify a larger variety of hand gestures for controlling keyboard and mouse functions. Furthermore, a more intuitive interaction system can be produced by incorporating various modalities, such as voice commands, facial expressions, and hand gestures. To do this, the Haar cascade must be combined with additional computer vision algorithms or methods for natural language processing. To increase user satisfaction, customization features can also be added so that users can adjust mappings and gestures to suit their preferences. Furthermore, it's critical to guarantee accessibility for people with disabilities by integrating assistive technologies or using alternate input methods. Its functionality and usability can be increased by integrating the system with other software programs or platforms and optimizing it for real-time performance. Additionally, modifying the system for wearables or mobile devices creates new markets and use cases. Strong authentication and encryption protocols are also necessary to address security and privacy concerns. To sum up, user input and testing are crucial for pinpointing problem areas and directing further development work. Ultimately, the project's future scope calls for a comprehensive strategy that includes usability upgrades, technical advancements, and system integration to produce a flexible and approachable virtual mouse and keyboard.

Although deep learning techniques can address limitations like accuracy and robustness, virtual mouse and keyboard systems utilizing Haar cascades show potential. Future developments will focus on enabling multimodal control, such as voice and eye gaze commands, expanding applications in hygienic environments, VR/AR, and accessibility, and enabling advanced interactions. Virtual interfaces can be made more flexible and user-friendly by overcoming obstacles and investigating these opportunities.

Chapter 8. Conclusion

In this study, a virtual mouse application based on object tracking was developed and implemented using a webcam and the Python programming environment with OpenCV libraries. This technology has numerous applications in areas such as augmented reality, computer graphics, gaming, prosthetics, and biomedical engineering.

We created a system that takes in inputs from eyes on the screen to control the mouse cursor, captured in real-time through a camera. The development and implementation of a Gesture-Based Virtual Mouse and Keyboard system utilizing eye and hand gestures represent a significant advancement in human-computer interaction. Through the integration of eye-tracking technology and hand gesture recognition, the system offers users a more intuitive and efficient means of interacting with computers, free from the constraints of traditional input devices. All standard mouse functions such as left and right clicks, double clicks, and scrolling were integrated into the system. The results indicate that if the vision algorithms can perform well in a range of environments, our system will function more effectively, potentially improving presentation experiences and reducing workspace. We aimed to create this technology as affordably as possible, while also ensuring compatibility with a standardized operating system, with the potential to aid patients who lack mobility in their limbs. The overarching goal was to develop a virtual mouse and keyboard using hand gesture recognition and image processing to control the movement of the mouse pointer according to hand gestures.

Acknowledge

We are very thankful to all the teachers who have provided us with valuable guidance towards the completion of this project work on Gesture Recognition Based Virtual Mouse and Keyboard. We express our sincere gratitude towards the cooperative department which has provided us with valuable assistance and requirements for the project work.
We are very grateful and want to express our thanks to Prof. S. A. Nagtilak for guiding us in the right manner, correcting our doubts by giving her time whenever we required, and providing her knowledge and experience in making this project work. We are also thankful to the HOD of our Information Technology department Dr. M. L. Bangare for his moral support and motivation which has encouraged us to make this project work. We are also thankful to our Vice Principal Dr. K. R. Borole and our Principal Dr. A.V. Deshpande, who provided his constant support and motivation that made a significant contribution to the success of this project.

References

[1] J. Shin and C. M. Kim, "Non-Touch Character Input System Based on Hand Tapping Gestures Using Kinect Sensor," in IEEE Access, vol. 5, pp. 10496-10505, 2017, doi: 10.1109/ACCESS.2017.2703783. [IRJASET 2023].

[2] L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 2017, pp. 483-487, doi: 10.1109/ICEMI.2017.8265863.

[3] C. Li, C. -K. Kim and J. -S. Park, "The Indirect Keyboard Control System by Using the Gaze Tracing Based on Haar Classifier in OpenCV," 2009 International Forum on Information Technology and Applications, Chengdu, China, 2009, pp. 362-366, doi: 10.1109/IFITA.2009.276.

[4] CHAVALI, E Sankar. (2023). Virtual Mouse Using Hand Gesture. 7. 7. 10.55041/IJSREM21501

[5] Y. Zhang, W. Yan and A. Narayanan ”A Virtual Keyboard Implementation Based on Finger Recognition” Auckland University of Technology, DOI: 10.1109/IVCNZ.2017.8402452ISSN: 2151-2205, vol(17)[2017 IEEE].

[6] S. R. Chowdhury, S. Pathak and M. D. A. Praveena, "Gesture Recognition Based Virtual Mouse and Keyboard," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 585-589, doi: 10.1109/ICOEI48184.2020.9143016.

[7] C. Topal, B. Benligiray and C. Akinlar, "On the efficiency issues of virtual keyboard design," 2012 IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS) Proceedings, Tianjin, China, 2012, pp. 38-42, doi: 10.1109/VECIMS.2012.6273205.
[8] D. E. V. Ligarreto and D. L. De Luise, "Metrics design for keyboard and mouse: Assessing learning levels," 2017 IEEE URUCON, Montevideo, Uruguay, 2017, pp. 1-4, doi: 10.1109/URUCON.2017.8171846.
[9] N. Nooruddin, R. Dembani and N. Maitlo, "HGR: Hand-Gesture-Recognition Based Text Input Method for AR/VR Wearable Devices," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 2020, pp. 744-751, doi: 10.1109/SMC42975.2020.9283348.
[10] C. Lian et al., "Towards a Virtual Keyboard Scheme Based on Wearing One Motion Sensor Ring on Each Hand," in IEEE Sensors Journal, vol. 21, no. 3, pp. 3379-3387, 1 Feb.1, 2021, doi: 10.1109/JSEN.2020.3023964.
[11] Pratiksha Kadam, Prof. Minal Junagre, Sakshi Khalate, Vaishnavi Jadhav, Pragati Shewale, “Gesture Recognition based Virtual Mouse and Keyboard” Department of Computer Engineering, Indira College of Engineering and Management, Pune, India.vol(11), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538,[2023 IRJASET].
[12] Dinh-Son Tran, Ngoc-Huynh Ho, Hyung-Jeong Yang, Guee Sang Lee, “Real-time virtual mouse system using RGB-D images and fingertip detection”, Chonnam National University.
[13] Multimedia Tools and Applications 80(7):1-18, DOI:10.1007/s11042-020-10156-5 (Vol 8)[2020 Springer].
[14] Mishaha MK, Manjusha MS PG Scholar,” Virtual Mouse And Keyboard For Computer Interaction“, Dept. Computer Science & Engineering, M.Dasan Institute Technology Ulliyeri.
[15] Koyilandi, Kozhikoode,673620, DOI:10.46647/ijetms.2022.v06i04.0036 ISSN: 2581-4621 ,vol(6)[2022 IJETMS].
[16] Eichelberger, C., Wagner, D., & Schmalstieg, D. "Real-time detection of hand and finger poses in a head-mounted display for 3D interaction.", Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2(4), 168. [IMWUT 2018.].
[17] Zhang, X., Sugano, Y., Fritz, M., & Bulling, A., "Gaze estimation using deep learning and its applications in driver monitoring." In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4117-4126.[2015]
[18] Rahim, M. S. I., & Molla, M. M. M., "A survey of hand pose recognition with Microsoft Kinect.", Computer Vision and Image Understanding, 131, 18-39.,[2015].