International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Gesture-Driven Automation for Dynamic Presentations

Prof. Kiran Itagi [a], *Shivam Patil* [b], *Shreya Gawade* [c], *Veda Patil* [d], *Sanjana Shinde* [e]

[a,b,c,d,e] *Department of Computer Science and Engineering, Angadi Institute of Technology and Management, Belagavi-590009, India*

ABSTRACT

This research paper explores the development and implementation of an automated system for digital presentation control using hand gesture techniques. The aim of this study is to investigate the feasibility and effectiveness of utilizing hand gestures as a natural and intuitive interface for controlling presentations. The proposed system leverages computer vision and machine learning algorithms to recognize and interpret hand gestures, providing a hands-free and interactive solution for presenters. The research involves the design, development, and evaluation of the system, comparing its performance with traditional presentation control methods.

Keywords: Automated Presentation Control, Hand Gesture Techniques, Computer Vision, Machine Learning, Human-Computer Interaction.

1. Introduction

Over the past few decades, hand gesture recognition has surfaced as a ground breaking technique within the domain of human-computer interaction (HCI), signifying a notable shift away from traditional input methods like keyboards and mice. This inventive approach is distinguished by its automatic, intuitive, and user-friendly features, providing a seamless channel of communication between individuals and computing systems.

In the context of presentation software, gesture recognition introduces exciting possibilities for users to navigate slides, execute commands, and engage with content effortlessly, all through natural hand movements. This advancement not only improves the user experience by offering a hands-free and responsive interaction method but also reflects a broader trend of utilizing human gestures as a powerful input method across different software applications. The integration of gesture recognition into presentation software showcases the ongoing evolution of technology to align with more natural and user-friendly interaction styles, contributing to a more enjoyable and efficient computing experience.

Presenters traditionally navigate PowerPoint slides using input devices such as a mouse, keyboard, or remote control. While these methods have been standard, they come with inherent inconveniences and discomforts. The use of a mouse may tether presenters to a specific location, limiting their mobility during a presentation. Keyboards can be cumbersome and may disrupt the flow of the presentation, requiring presenters to divert attention from their audience. Remote controls, though offering mobility, can be prone to technical glitches or misplacement.

To tackle these issues, people are looking into easier and less disruptive options. One exciting solution is gesture recognition technology. This cool tech lets

presenters more freedom to move around we interact with presentation software much smoother and user-friendly.and connect with the audience. This shift to using gestures instead of clunky devices is making the way.

2. Methodology

The Methodology for Gesture Driven Automation For Dynamic Presentations is Depend on specific algorithms that process the captured gestures and performs the actions. Here are a few general steps that might be involved in this process:

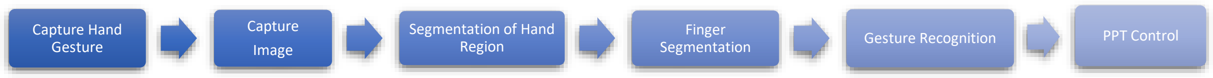

Fig. 1 – Methodology Flow

The development of the system was executed using the Python programming language, incorporating key libraries such as OpenCV, Mediapipe Hands, and PyWin32. OpenCV played a crucial role in frame capture from the camera system and implemented additional functionalities to optimize the utilization of MediaPipe for directional capture. MediaPipe Hands, in turn, captured hand movements from the frames provided by OpenCV, recognizing user-generated gestures as the basis for slide changes. The hand images were processed as a mesh of 21 points, facilitating hand gesture recognition. Following gesture recognition, PyWin32 took control of the system and executed the corresponding tasks.

In contrast to existing systems, the gesture recognition system in this work deviates significantly. Rather than relying on machine learning algorithms, it employs a mathematical logic derived from the Finger Up function within the Hand Tracking Module developed using OpenCV and Media Pipe. This unique approach eliminates the need for extensive data collection and training sets, achieving gesture recognition without dependence on machine learning techniques.

A. Computer Vision

OpenCV, an open-source computer vision library, played a pivotal role in the project by providing pre-defined algorithms for various computer vision tasks. This included frame capturing from the camera for input, facilitating essential functions such as object detection, face detection, and biometric recognition.

B. Media Pipe

Media Pipe, Google's open-source framework for real-time multimedia processing, offered pre-built machine learning models and a flexible architecture for tasks like image processing, object detection, and facial recognition. Its modular design and emphasis on real-time performance simplified the development of computer vision applications for mobile devices and embedded systems.

C. Media Pipe Hands

Media Pipe's Hand Tracking solution utilizes machine learning for real-time hand detection and tracking in video or image input. It can accurately detect and track 21 points on the hand, enabling precise representation of hand movements. This functionality supports applications like hand gesture recognition, virtual try-on, and augmented reality interactions.

1. Palm Detection Model

Media Pipe introduced a specialized palm detection model that employs a unique approach by estimating bounding boxes for rigid objects like palms and fists. This simplifies hand detection by focusing on palm detection and utilizes advanced feature extraction and loss optimization methods. The model achieved impressive results with an average precision of 95.7% in palm detection.

2. Palm key points model

The Palm Key Points Model anticipates the coordinates of 21 crucial positions on the palm, employing regression methods for precise localization. Trained on a varied dataset that encompasses genuine hand images and synthetic palm models, the model guarantees precise localization even in demanding situations such as self-occlusions or hands that are only partially visible.

D. PyWin32

PyWin32, a Python library, provides access to the PowerPoint COM object model, allowing interaction with presentation elements such as slides and shapes. The win32com.client module within PyWin32 enables the creation of COM objects and facilitates actions like opening presentations and manipulating slide content.

3. Comparison Table

The presented papers encompass a diverse range of hand gesture recognition applications, each with unique methodologies, techniques, and focuses.In [1] A comprehensive survey on hand gesture recognition is conducted, offering an extensive overview of various recognition techniques for both static and dynamic gestures. While it provides a broad understanding of the field and highlights potential applications, it lacks specific implementation details.

The next [2] focus on practical applications of hand gestures, particularly in the context of PowerPoint presentations. The introduces a novel approach for navigation using distance transform and skin color segmentation, offering a hands-free interaction during presentations. However it may be limited to specific hand positions or conditions.

The [3] concentrates on finger-pointing gestures for slide presentations, potentially enhancing engagement and interactivity, but it may lack diversity in the types of gestures considered.

The [4] proposes real-time hand gesture recognition for human-computer interaction, addressing the need for efficiency but lacks specific algorithmic details. Lastly, the [5] integrates real-time object detection with TensorFlow and OpenCV, showcasing the potential to enhance user interaction in real-time scenarios, albeit with performance variations influenced by the complexity of detected objects, contributing to the evolving field of gesture recognition. Overall, the papers collectively contribute to the evolving field of gesture recognition, with strengths in surveying, practical applications, and real-time interaction. Yet, certain limitations, such as the lack of detailed implementations or specificity in gesture types, should be considered for future research.

SL No.	Title	Authors	Advantages	Disadvantages	Accuracy
1	A survey on hand gesture recognition	Lingchen Chen, Feng Wang, Hui Deng, Kaifan Ji	Offers an extensive overview of various hand gesture recognition method	Lacks specific details on the implementation of techniques.	92.6%
2	Navigation of PowerPoint Using Hand Gestures	Dnyanada R Jadhav, L. M. R. J Lobo	Provides a novel approach for hand gesture-based navigation in PowerPoint	May be limited to specific hand positions or conditions	89.7%
3	Finger-Pointing Gesture Analysis for Slide Presentation	Maisevli Harika, Ary Setijadi P, Hilwadi Hindersah, Bong-Kee Sin	Concentrates on a specific type of gesture, potentially enhancing accuracy.	May lack diversity in the types of gestures considered.	95.4%
4	Real Time Hand Gesture Recognition for Human Computer Interaction	Rishabh Agrawal, Nikita Gupta	Addresses the need for efficient and responsive gesture recognition.	Lack of details on the specific algorithms or technologies used.	90%
5	Detection of Real Time Objects Using TensorFlow and OpenCV	Ajay Talele, Aseem Patil, Bhushan Barse	Real-time object detection	Dependent on lighting conditions	97%

Fig 2. Comparison Table

4. Conclusion

In conclusion, the presented papers collectively contribute to the burgeoning field of hand gesture recognition with varied methodologies and applications. The comprehensive survey provides a broad understanding of recognition techniques, while the practical applications in PowerPoint navigation and slide presentations offer innovative ways to enhance user interaction. The emphasis on real-time gesture recognition and object detection reflects the growing demand for responsive human-computer interfaces. However, the limitations, such as a lack of detailed implementations in some cases and potential specificity in gesture types, underscore the need for further research to refine and advance the effectiveness of these approaches. Overall, these papers serve as valuable stepping stones in the pursuit of more sophisticated and inclusive hand gesture recognition systems.

References

[1] D. Jadhav, Prof. L.M.R.J. Lobo, Hand Gesture Recognition System To Control Slide Show Navigation IJAIEM, Vol. 3, No. 4 (2014)

[2] M. Harika, A. Setijadi P, H. Hindersah, FingerPointing Gesture Analysis for Slide Presentation, Bong-Kee Sin Journal Of Korea Multimedia Society, Vol. 19, No. 8, August (2016)

[3] Md.F. Wahid, R. Tafreshi, M. Al-Sowaidi, R. Langari, An Efficient Approach to Recognize Hand Gestures Using Machine-Learning Algorithms, IEEE 4th MECBME, (2018)

[4] D.O. Lawrence, and Dr. M.J. Ashleigh, Impact Of Human-Computer Interaction (Hci) on Users in Higher Educational System: Southampton University As A Case Study, Vol.6, No 3, pp. 1-12, September (2019)

[5] I. Dhall, S. Vashisth, G. Aggarwal, Automated Hand Gesture Recognition using a Deep Convolutional Neural Network, 10th International Conference on Cloud Computing, Data Science & Engineering(Confluence), (2020)

[6] Ren, Zhou, et al. Robust part-based hand gesture recognition using kinect sensor, IEEE transactions on multimedia 15.5, pp.1110-1120, (2013)

[7] Ajay Talele, Aseem Patil, Bhushan Barse on the Detection of Real Time Objects Using TensorFlow and OpenCV, Asian Journa of Convergence in Technology, Vol 5, (2019)

[8] 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)

[9] Sebastian Raschka, Joshua Patterson and Corey Nolet, Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence, (2020)

[10] Xuesong Zhai, Xiaoyan Chu, Ching Sing chai, Morris Siu Yung Jong, Andreja Istenic, Michael Spector, Jia Bao Liu, Jing Yuan, Yan Li, A Review of Artificial Intelligence (AI) in Education from 2010 to 2020, (2021)