HAND AND FACE LANDMARKS DETECTION USING MEDIA PIPE AND AI

K. V. Bharath Kumar
KV SubbaReddy Engineering College, Kurnool, A.P, India

B Purushotham, B. Sushanth Sai, P. Sathish, M. Shoaib
KV SubbaReddy Engineering College, Kurnool, A.P, India

Abstract

This project presents a real-time face mesh and hand gesture detection system using Media Pipe and OpenCV. The system accurately detects 468 facial landmarks and 21 hand landmarks through a webcam stream. The detected facial and hand features are drawn on a black canvas to highlight the mesh clearly, facilitating use cases such as expression analysis, sign language interpretation, and gesture-controlled applications. The project demonstrates the power of lightweight, real-time machine learning pipelines for computer vision on standard hardware.

Keywords: *Real-time Detection, Face Mesh, Hand Gesture Recognition, Media Pipe, OpenCV, Facial Landmarks, Hand Landmarks, Computer Vision, Human-Computer Interaction, Gesture-Controlled Applications, Expression Analysis, Sign Language Interpretation, Lightweight ML Pipelines.*

I.INTRODUCTION

Hand and face landmarks detection are fundamental tasks in computer vision and human-computer interaction, enabling various applications such as gesture recognition, facial expression analysis, and augmented reality. Traditional methods often lack accuracy and speed for real-time applications, while existing deep learning approaches focus on either hands or faces separately. This project introduces a novel system that combines the power of Mediapipe and Artificial Intelligence to achieve accurate and real-time detection of landmarks on both hands and faces. The proposed system aims to provide robust and precise landmark detection for diverse applications, contributing to advancements in human-computer interaction and computer vision.

II. LITERATURE SURVEY

Current landmark detection systems often rely on handcrafted features and traditional machine learning models, limiting their accuracy and efficiency in real-time scenarios. While some deep learning-based methods exist, they typically focus on either hand or face landmarks individually, overlooking the need for a unified solution. The proposed system bridges this gap by leveraging Mediapipe and state-of-the-art deep learning techniques to detect landmarks on both hands and faces simultaneously. By combining these technologies, the system aims to overcome the limitations of existing methods and offer a unified solution for landmark detection in real-time applications.

III.SYSTEM ANALYSIS

The Systems Development Life Cycle (SDLC), or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. In software engineering the SDLC concept underpins many kinds of software development methodologies.

EXISTING SYSTEM

Traditional landmark detection systems often rely on handcrafted features and machine learning models, which may not provide the required accuracy and speed for real-time applications. While some deep learning-based methods have been introduced, they often focus on either face or hand landmarks separately, leading to inefficiencies in multi-modal applications.

DISADVANTAGES

Traditional landmark detection systems suffer from several disadvantages that limit their effectiveness in real-time applications. They often rely on handcrafted features and conventional machine learning models, which generally lack the precision and speed required for responsive performance. Although some deep learning-based methods have emerged, these typically focus on detecting either hand or face landmarks in isolation. This separation leads to inefficiencies when such systems are used in multi-modal scenarios that require simultaneous detection of both. Furthermore, these approaches may demand higher computational resources or experience latency issues, making them unsuitable for seamless, real-time applications on standard hardware.

PROPOSED SYSTEM

Our proposed system combines the strengths of the Mediapipe framework with state- of-the-art deep learning techniques to achieve accurate and real-time detection of landmarks on both hands and faces. The system is designed to address the limitations of existing methods and provide a unified solution for landmark detection. Key components include

ADVANTAGES

Enhanced Understanding of Consumer Behavior: The proposed system seeks to uncover detailed insights into how consumers shop for groceries online, contributing to academic research and managerial knowledge.
Convenience and Accessibility: Offering a user-friendly interface accessible via various devices, the system aims to eliminate physical store visits, providing doorstep deliveries.
Secure Transactions and Varied Payment Options: Secure online payment methods, diverse product catalog, and personalized user accounts ensure a safe and smooth shopping experience.
Admin Control and Business Analytics: The system's administrative dashboard enables efficient inventory management, order tracking, customer support, and generates insightful business reports.

IV.REQUIREMENTS SPECIFICATION

Requirement Specification provides a high secure storage to the web server efficiently. Software requirements deal with software and hardware resources that need to be installed on a serve which provides optimal functioning for the application. These software and hardware requirements need to be installed before the packages are installed. These are the most common set of requirements defined by any operation system. These software and hardware requirements provide a compatible support to the operation system in developing an application.

HARDWARE REQUIREMENTS:

The hardware requirement specifies each interface of the software elements and the hardware elements of the system. These hardware requirements include configuration characteristics.

- System : Intel core i5.
- Hard Disk : 500 GB.
- Monitor : 15’ HD Color.
- Mouse : Logitech.
- RAM : 8 GB.

SOFTWARE REQUIREMENTS:

The software requirements specify the use of all required software products like data management system. The required software product specifies the numbers and version. Each interface specifies the purpose of the interfacing software as related to this software product.

- Operating system : Windows 10/11
- Coding Language: Python 3.8

FUNCTIONAL REQUIREMENTS

The functional requirement refers to the system needs in an exceedingly computer code engineering method.

The key goal of determinant "functional requirements" in an exceedingly product style and implementation is to capture the desired behavior of a software package in terms of practicality and also the technology implementation of the business processes.

NON FUNCTIONAL REQUIREMENTS

All the other requirements which do not form a part of the above specification are categorized as Non-Functional needs. A system perhaps needed to gift the user with a show of the quantity of records during info. If the quantity must be updated in real time, the system architects should make sure that the system is capable of change the displayed record count at intervals associate tolerably short interval of the quantity of records dynamic. Comfortable network information measure may additionally be a non-functional requirement of a system.

The following are the features:

- Accessibility
- Availability
- Backup
- Certification
- Compliance
- Configuration Management
- Documentation
- Disaster Recovery
- Efficiency (resource consumption for given load)
- Interoperability

V.METHODOLOGY

SDLC (Software Development Life Cycle) – Umbrella Model

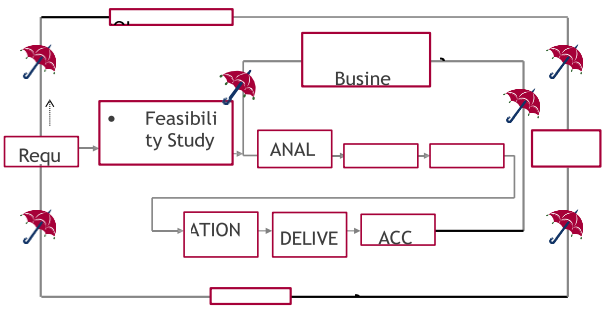


Fig No. 6.1 umbrella model

SDLC is nothing but Software Development Life Cycle. It is a standard which is used by software industry to develop good software.

Requirements Gathering Stage

The requirements gathering process takes as its input the goals identified in the high- level requirements section of the project plan. Each goal will be refined into a set of one or more requirements. These requirements define the major functions of the intended application, define operational data areas and reference data areas, and define the initial data entities. Major functions include critical processes to be managed, as well as mission critical inputs, outputs and reports. A user class hierarchy is developed and associated with these major functions, data areas, and data entities. Each of these definitions is termed a Requirement. Requirements are identified by unique requirement identifiers and, at minimum, contain a requirement title and textual description.

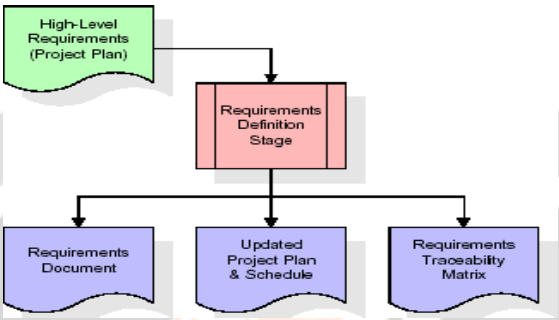


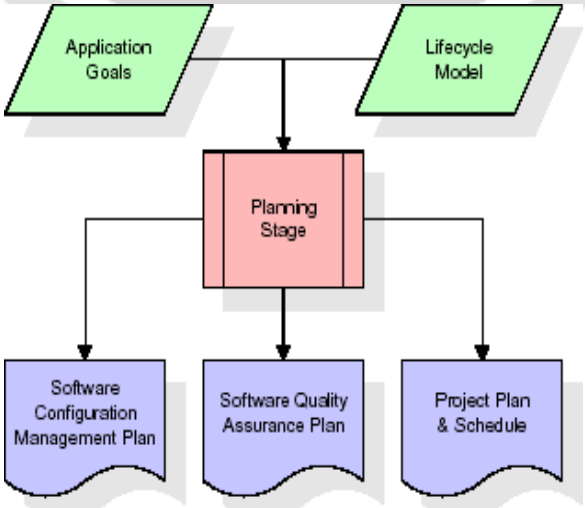
Fig No. 6.2 Requirements Gathering stage

These requirements are fully described in the primary deliverables for this stage: the Requirements Document and the Requirements Traceability Matrix (RTM). The requirements document contains complete descriptions of each requirement, including diagrams and references to external documents as necessary. Note that detailed listings of database tables and fields are not included in the requirements document.

The title of each requirement is also placed into the first version of the RTM, along with the title of each goal from the project plan. The purpose of the RTM is to show that the product components developed during each stage of the software development lifecycle are formally connected to the components developed in prior stages.

In the requirements stage, the RTM consists of a list of high-level requirements, or goals, by title, with a listing of associated requirements for each goal, listed by requirement title. In this hierarchical listing, the RTM shows that each requirement developed during this stage is formally linked to a specific product goal. In this format, each requirement can be traced to a specific product goal, hence the term requirements traceability.

The outputs of the requirements definition stage include the requirements document, the RTM, and an updated project plan.



VI.RESULT

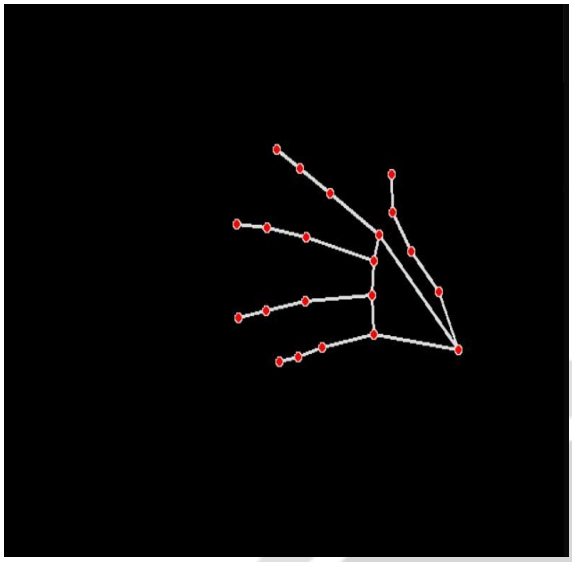


Fig NO:9.1 Hand Detection Screenshot

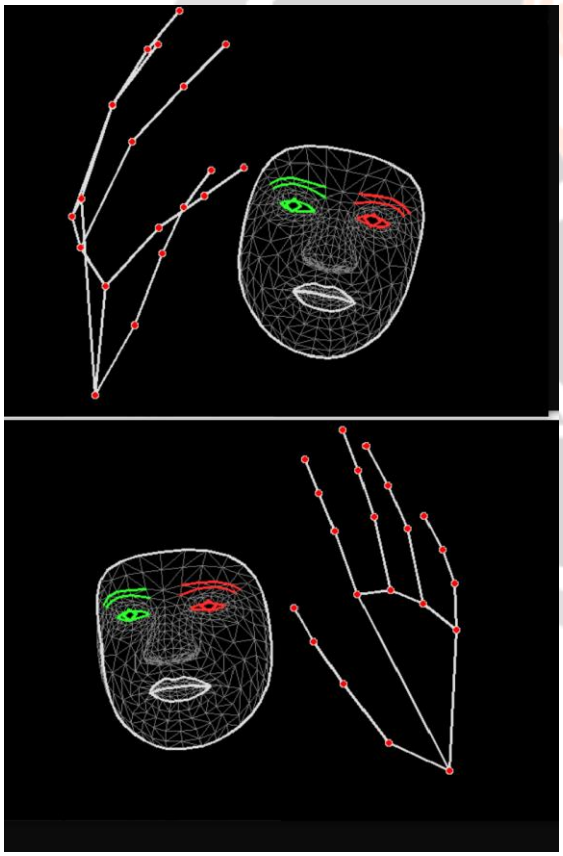


Fig No. 9.2 Hand and face Detection Screenshot

VII.CONCLUSION

The proposed system represents a significant advancement in hand and face landmarks detection by leveraging the capabilities of the Mediapipe framework and deep learning. By providing a unified solution for both modalities, it opens up new possibilities for interactive and immersive applications. The real- time capabilities of the system make it suitable for a wide range of use cases, from enhancing user experiences in gaming to assisting in medical diagnostics and beyond. The successful implementation of this system can lead to breakthroughs in human-computer interaction and computer vision applications..

Future Enhancements:

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment. Based on the future security issues, security can be improved using emerging technologies like single sign-on.

VIII.REFERENCES

1. Dongyue Chen, Qiusheng Chen, Jianjun Wu, Xiaosheng Yu and Tong Jia, "Face Swapping: Realistic Image Synthesis Based on Facial Landmarks Alignment", Mathematical Problems in Engineering, vol. 2019, 2019.
2. Y. Zhang, L. Zheng and V. L. L. Thing, "Automated face swapping and its detection", 2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP), pp. 15-19, 2017.
3. I. Korshunova, W. Shi, J. Dambre and L. Theis, "Fast Face-Swap Using Convolutional Neural Networks", 2017 IEEE International Conference on Computer Vision (ICCV), pp. 3697-3705, 2017.
4. Luming Ma and Zhigang Deng, "Real-time Face Video Swapping From A Single Portrait", Symposium on Interactive 3D Graphics and Games (I3D 20). Association for Computing Machinery, pp. 1-10, 2020.
5. W. Zhang and C. Zhao, "Exposing Face-Swap Images Based on Deep Learning and ELADetection", Proceedings, vol. 46, pp. 29, 2020.
6. S. Sountharrajan et al., "Automatic classification on bio medical prognosisof invasive breast cancer", Asian Pacific Journal of Cancer Prevention: AP JCP, vol. 18.9, pp. 2541, 2017.
7. S. Sountharrajan et al., "Automatic glioblastoma multiforme detection using hybrid-SVM with improved particle swarm optimisation", International Journal of Biomedical Engineering and Technology, vol. 26.3–4, pp. 353-364, 2018.
E. Suganya et al., "Mobile cancer prophecy system to assist patients: Big data analysis and design", Journal of Computational and Theoretical Nanoscience, vol. 16.8, pp. 3623-3628, 2019.
1. M. Karthiga et al., "Machine Learning Based Diagnosis of Alzheimers Disease", International Conference on Image Processing and Capsule Networks, 2020.
2. M. Karthiga et al., "Malevolent Melanoma diagnosis using Deep Convolution Neural Network", Research Journal of Pharmacy and Technology, vol. 13.3, pp. 1248-1252, 2020.
3. S. Mahajan, L. Chen and T. Tsai, "SwapItUp: A Face Swap Application for Privacy Protection", 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), pp. 46-50, 2017.
4. Qian Zhang, Hao Zheng, Tao Yan and Jiehui Li, "3D Large-Pose Face Alignment Method Based on the Truncated Alexnet Cascade Network", Advances in Condensed Matter Physics, vol. 2020, 2020.
5. Mohammed Alghaili, Li Zhiyong and Hamdi A. R. Ali, "FaceFilter: Face Identification with Deep Learning and Filter Algorithm", Scientific Programming, vol. 2020, 2020.
6. Qian Zhang, Hao Zheng, Tao Yan and Jiehui Li, "3D Large-Pose Face Alignment Method Based on the Truncated Alexnet Cascade Network", Advances in Condensed Matter Physics, vol. 2020, 2020.
7. Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt and Matthias Niessner, "FaceVR: Real-Time Gaze-Aware Facial Reenactment in Virtual Reality", ACM Trans. Graph., vol. 37, no. 2, June 2018.
8. Thibaut Weise, Sofien Bouaziz, Hao Li and Mark Pauly, "Realtime Performance based Facial Animation", ACM Trans. Graph., vol. 30, no. 4, pp. 77: 1-77: 10, July 2011.
Chenglei Wu, Takaaki Shiratori and Yaser Sheikh, "Deep Incremental Learning for Efficient High-fidelity Face Tracking", ACM Trans. Graph., vol. 37, no. 6, Dec. 2018.
1. Feng Xu, Jinxiang Chai, Yilong Liu and Xin Tong, "Controllable High-fidelity Facial Performance Transfer",

ACM Trans. Graph., vol. 33, no. 4, July 2014.

2. Li Zhang, Noah Snavely, Brian Curless and Steven M. Seitz, "Spacetime Faces: High Resolution Capture for Modeling and Animation", ACM Trans. Graph., vol. 23, no. 3, pp. 548- 558, Aug. 2004.

3. Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, et al., "Real-time Non-rigid Reconstruction Using an RGB-D Camera", ACM Trans. Graph., vol. 33, no. 4, pp. 156: 1-156: 12, July 2014.