VISVESVARAYA TECHNOLOGICAL UNIVERSITY
Jnana Sangama, Belagavi - 590018

A DIP MINI PROJECT REPORT ON

"FINGER COUNTER DETECTION"

Submitted in the partial fulfillment of the requirement for the sixth semester of

BACHELOR OF ENGINEERING

In

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

By

SIDDARTHA R(1RR20AI029)

Under the guidance of

MAMATHA K R

Associate Professor,
Dept. of AIML,
RRCE

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

RAJARAJESWARI COLLEGE OF ENGINEERING
MYSORE ROAD, BANGALORE-560074
(An ISO 9001:2008 Certified Institute)

(2022-23)

RAJARAJESWARI COLLEGE OF ENGINEERING
MYSORE ROAD, BANGALORE-560074
(An ISO 9001:2008 Certified Institute)
(Affiliated to Visvesvaraya Technological University, Belagavi)
DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

CERTIFICATE

"FINGER COUNTER DETECTION"

Carried out by

SIDDARTHA R(1RR20AI029)

The students of **"Rajarajeswari College of Engineering"** in partial fulfillment for the sixth semester of **Bachelor of Engineering** in **Artificial Intelligence and Machine Learning** of the Visvesvaraya Technological University, Belagavi during the year **2023.** It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini project work prescribed for the sixth semester.

.....	.....
Signature of guide	Signature of HOD
[MAMATHA K R]	**[Dr. K RAJESH]**
Assistant Professor, Dept. of AIML	Prof. & HOD, Dept. of AIML
RRCE, Bangalore	RRCE, Bangalore

External Viva-Voce

Name of the examiners:	Signature with date:
1._____	1. _____
2._____	2. _____

ACKNOWLEDGEMENT

I am truly thankful and convey my sincere gratitude to the principal, **Dr. R Balakrishna** Rajarajeswari College of Engineering, Bangalore.

I convey my sincere gratitude to **Dr. K Rajesh**, HOD, Department of Artificial Intelligence and Machine Learning for his meticulous support, continuous co-operation, valuable suggestion and encouragement during the development of the project. I also extend my thanks for his invaluable guidance to imbibe the requisite knowledge for success of our project.

I convey my sincere gratitude to **Mamatha K R**, Assistant Professor, Department of Artificial Intelligence and Machine Learning, for her meticulous support continuous co-operation, valuable suggestion and encouragement during the development of the project. I also extend my gratitude for her guidance to imbibe the requisite knowledge for success of my project, as an internal guide. I am very much obliged.

I also thank my parents who have encouraged me and supported me in every stage of development of this project.

Last but not the least, my wishes to the entire Artificial Intelligence and Machine Learning for their help and guidance, encouragement, inspiration and co-operation at all stages of the development of this project without which this project wouldn't have been a success.

Finally, I express my heartfelt gratitude to all those who helped me to complete the project work successfully by providing support, suggestions, advise, guidance and much needed encouragement.

SIDDARTHA R(1RR20AI029)

ABSTRACT

The "Finger Counter Detection Using Digital Image Processing with Python" project aims to develop a computer vision system that can accurately count the number of fingers held up by a hand in an image or video stream. This project leverages the power of digital image processing techniques implemented using the Python programming language.

The system utilizes a webcam or any input device capable of capturing images or video frames. The image processing pipeline begins with capturing the input frames, followed by applying various image processing operations to enhance the hand and finger detection. These operations include image threshold, contour detection, and convex hull analysis.

After identifying the hand region in the frame, the system performs finger segmentation to separate individual fingers. This process involves techniques such as morphological operations and finger tracking algorithms to accurately isolate each finger.

The final step of the system is finger counting, where the program uses pattern recognition and geometric analysis to count the number of isolated fingers. The program can then display the finger count in real-time on the output frame or provide the count as a numerical output.

The developed system demonstrates the potential of digital image processing techniques implemented in Python for real-time finger counting applications, with possible applications in human-computer interaction, gesture recognition, and virtual reality interfaces.

TABLE OF CONTENTS

INTRODUCTION

The project "Finger Counter Detection" Using Digital Image Processing with Python focuses on developing a computer vision system that can detect and count the number of fingers in real-time. The system utilizes the power of digital image processing techniques implemented using the Python programming language. By leveraging the capabilities of Python libraries such as OpenCV, NumPy, and scikit-image, the project aims to provide an efficient and reliable solution for finger counting.

The main goal of this project is to create a system that can accurately count fingers even in challenging scenarios, including varying hand poses, different lighting conditions, and complex backgrounds. The system will be able to process live video streams from a webcam or any other input device and display the finger count in real-time.

The Python programming language is chosen for its ease of use, extensive library support, and wide adoption in the computer vision community. With Python, developers can implement complex image processing algorithms in a concise and efficient manner, enabling the development of a robust finger counting system.

PROBLEM DEFINITION

The specific objectives of the project include:

1. Developing a robust image processing pipeline to enhance hand and finger detection.
2. Implementing finger segmentation techniques to isolate individual fingers.
3. Designing a finger counting algorithm based on pattern recognition and geometric analysis.
4. Real-time processing of video frames from a webcam or other input devices.
5. Displaying the finger count in real-time on the output frame or providing a numerical output.

The project aims to overcome the following challenges:

1. Hand detection: Developing techniques to accurately detect and localize the hand region in the input images or video frames, despite variations in hand pose and background clutter.
2. Finger segmentation: Designing algorithms to separate individual fingers from the hand region, considering variations in finger orientation, occlusion, and complex backgrounds.
3. Robust counting: Creating an algorithm that can reliably count the number of fingers, even in challenging scenarios with varying lighting conditions and hand poses.
4. Real-time performance: Ensuring that the finger counting system operates in real-time, processing video frames at a sufficiently high frame rate to provide immediate feedback.

The general flow of the finger counting system for your reference:

1. Capture input frame from the webcam or video source.
2. Preprocess the frame by applying image enhancement techniques, such as denoising and contrast adjustment.
3. Perform hand detection using techniques like color-based segmentation, skin detection, or machine learning-based approaches to localize the hand region.

4. Apply image processing techniques like thresholding, contour detection, and convex hull analysis to extract the contours and convex hull of the hand.
5. Segment individual fingers by applying morphological operations, such as erosion and dilation, to separate them from the hand region.
6. Implement finger tracking algorithms to ensure continuity and accuracy during finger segmentation.
7. Analyze the segmented fingers using pattern recognition and geometric analysis to count the number of fingers.
8. Display the finger count in real-time on the output frame or provide a numerical output.
9. Repeat the process for subsequent frames to achieve real-time finger counting.

LITERATURE STUDY

As per various literature surveys it is found that for implementing this project the basic concepts involved are:

1. Hand Detection and Localization
2. Image Preprocessing Techniques
3. Image Thresholding
4. Contour Detection
5. Convex Hull Analysis
6. Finger Segmentation
7. Morphological Operations
8. Finger Tracking Algorithms
9. Pattern Recognition for Finger Counting
10. Real-time Video Processing
11. Python Libraries for Image Processing (OpenCV, NumPy, scikit-image)

SOFTWARE REQUIREMENTS

ANACONDA

It is a free and open source distribution of the Python for data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system. The Anaconda distribution is used by over 6 million users, and it includes more than 250 popular data science packages suitable for Windows, Linux, and MacOS.

JUPYTER NOTEBOOK

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations and text. Jupyter notebook is maintained by the people at project Jupyter. Jupyter notebooks are a spin of project from the IPython project, which used to have an IPython notebook project itself.

HARDWARE INTERFACES:

1. **Processor:** Intel CORE i7 processor with minimum 2.9 GHz speed.
2. **RAM:** Minimum 4 GB.
3. **Hard Disk:** Minimum 500 GB

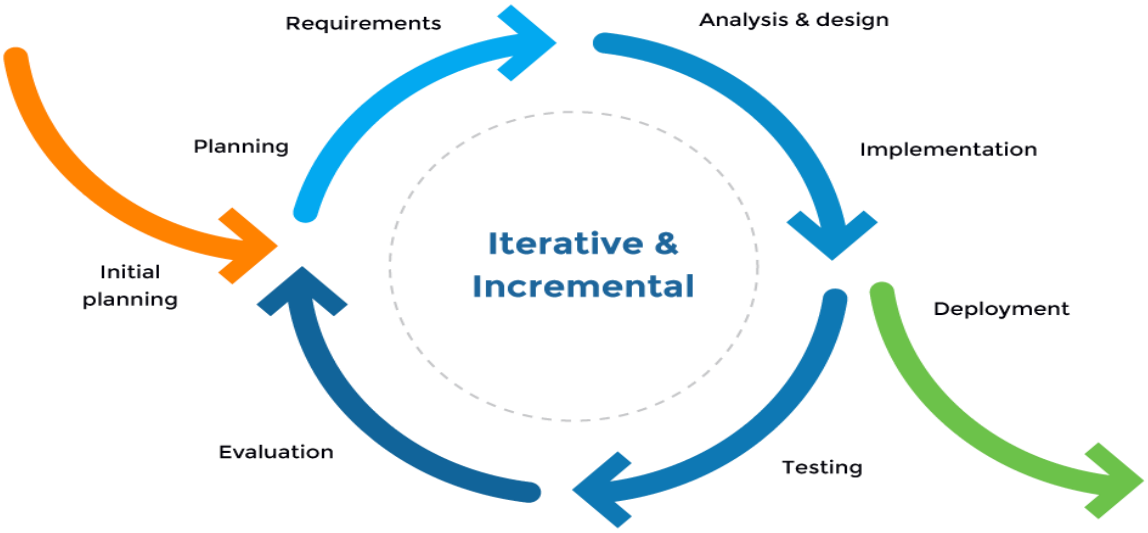
SOFTWARE INTERFACES:

1. Python Programming Language
2. OpenCV framework
3. Windows 11
4. MediaPipe library
5. Time library

PLANNING

The steps we followed while developing this project are-:

1. Analysis of the problem statement.
2. Gathering of the requirement specification
3. Analyzation of the feasibility of the project.
4. Development of a general layout.
5. Going by the journals regarding the previous related works on this field.
6. Choosing the method for developing the algorithm.
7. Analyzing the various pros and cons.
8. Starting the development of the project.
9. Installation of software like ANACONDA.
10. Developing an algorithm.
11. Analyzation of algorithm by guide.
12. Coding as per the developed algorithm.



ALGORITHM

STEP 1: Import required libraries.

STEP 2: Declare the Hand Detector and open the default camera to capture the hand.

STEP 3: Count the number of fingers.
Capture the frames continuously and detect the hand from the frame then detect how many fingers are up and count them. On the counter make appropriate conditions and place an image of fingers.

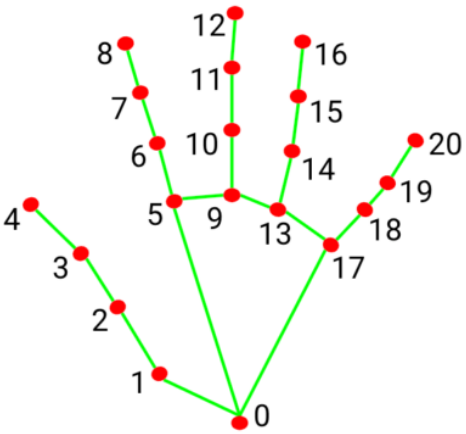
STEP 4: Terminate the loop.

THE LIBRARY AND PACKAGES

1.MEDIAPIPE

MediaPipe is a framework that provides customizable Machine Learning (ML) solutions (such as face and hand detection, hair segmentation, motion tracking, etc.) for live and streaming media. Their solution for hand detection and tracking is called MediaPipe Hands, and it employs ML to provide palm detection and a hand landmark model which consists of 21 3D landmarks, as shown in Figure.

These 3D landmarks are each composed of x, y, and z coordinates. x and y correspond to the landmark position, normalized from 0 to 1 by the image's width and height, respectively. The z component represents how close the landmark is to the camera. We will only use the x and y coordinates in this tutorial. Additionally, the solution provides a label related to the predicted handedness of the detected hand, indicating left or right.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

2.OpenCV

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly for real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage, then Itseez (which was later acquired by Intel). The library is cross-platform and licensed as free and open-source

software under Apache License 2. Starting in 2011, OpenCV features GPU acceleration for real-time operations.

OpenCV' s application areas include:

- 2D and 3D feature toolkits
- Egomotion estimation
- Facial recognition system
- Gesture recognition
- Human–computer interaction (HCI)
- Mobile robotics
- Motion understanding
- Object detection
- Segmentation and recognition
- Stereopsis stereo vision: depth perception from 2 cameras
- Structure from motion (SFM)
- Motion

GUIDING PRINCIPLES

1. **Non-intrusive and Contactless**: The finger counter detection system based on digital image processing does not require any physical contact with the fingers being counted. This non-intrusive nature makes it more hygienic and suitable for various applications where physical contact may not be desirable, such as healthcare, public spaces, or interactive displays.

2. **Scalability and Adaptability**: The use of Python for implementing the finger counter detection system allows for easy scalability and adaptability. Python offers a wide range of libraries and frameworks for image processing, computer vision, and machine learning, which can be leveraged to enhance the system's capabilities or adapt it to different environments and requirements.

3. **Accuracy and Reliability**: Digital image processing techniques, when properly implemented and calibrated, can provide accurate and reliable finger counting results. By leveraging advanced image processing algorithms and techniques, the system can handle variations in lighting conditions, hand orientations, and different finger shapes, improving overall accuracy.

4. **Accessibility**: The use of Python as the programming language for the project makes it accessible to a wide range of developers and researchers. Python has a simple and intuitive syntax, a vast community of developers, and extensive documentation, making it easier for individuals to contribute, modify, and expand the finger counter detection system.

5. **Versatility and Applications**: The developed finger counter detection system can have numerous applications across various domains. It can be used in interactive displays, sign language recognition, hand gesture-based interfaces, fitness tracking, virtual reality, augmented reality, and more. The versatility of the system opens up possibilities for innovative and interactive user experiences.

OUTPUT

1.COUNTING ZERO FINGER.

2.COUNTING ONE FINGER

3.COUNTING TWO FINGERS

4.COUNTING THREE FINGER

5.COUNTING FIVE FINGERS

FUTURE SCOPE

While the current implementation of the finger counter detection system offers valuable advantages and capabilities, there are several avenues for future development and expansion such as:

1. **Enhanced Accuracy and Robustness**: Further research can focus on improving the accuracy and robustness of the finger counter detection system. This can involve exploring advanced image processing techniques, machine learning algorithms, or deep learning approaches.

2. **Real-time Gesture Recognition**: Expanding the project to include real-time gesture recognition capabilities can open up opportunities for various interactive applications. By combining the finger counter detection system with machine learning algorithms, it becomes possible to recognize specific gestures or hand poses, enabling more advanced interactions in virtual reality, augmented reality, or gaming scenarios.

3. **Mobile and Embedded Systems Integration**: Adapting the finger counter detection system for mobile devices or embedded systems can enhance its portability and usability. This can involve optimizing the algorithms for resource-constrained environments, leveraging dedicated hardware accelerators.

4. **Integration with Existing Systems**: Exploring the integration of the finger counter detection system with other existing systems or applications can unlock new possibilities. For example, integrating it with healthcare monitoring systems, sign language recognition platforms, or human-computer interaction frameworks can enable seamless and innovative applications.

5. **Usability and Accessibility Improvements**: Consideration should be given to making the system more accessible to a wider range of users.

CONCLUSION

In conclusion, this project aimed to develop a finger counter detection system using digital image processing techniques with Python. The project successfully demonstrated the feasibility and potential of using these technologies to accurately count the number of fingers in real-time, providing a non-intrusive and cost-effective solution.

Through an extensive review of the literature, various digital image processing techniques and algorithms were explored and evaluated. The implementation of the system leveraged Python's libraries and frameworks, making it scalable, adaptable, and accessible to a wide range of developers.

The advantages of the project were significant. The non-intrusive nature of the system made it suitable for applications where physical contact is undesired, such as healthcare or public spaces. Additionally, its cost-effectiveness, scalability, and adaptability made it a viable solution for various domains and environments.

Looking towards the future, there are several avenues for further improvement and expansion. Enhancements in accuracy and robustness, multi-hand and multi-gesture recognition, real-time gesture recognition, integration with mobile and embedded systems, and user interface refinements present exciting opportunities for future development. Additionally, expanding the dataset and addressing usability and accessibility considerations can enhance the system's performance and inclusivity.

Through this project, we have not only gained valuable insights into the complexities of finger counting and digital image processing but also contributed to the advancement of non-intrusive, cost-effective, and accurate finger counting solutions. The potential impact of this technology on various domains is substantial, and we look forward to witnessing its integration and further refinement in future applications.

REFERENCES

1. S. Sridhar, Digital Image Processing, Oxford University Press, 2ndEdition, 2016.

2. Tom M. Mitchell, Machine Learning, McGraw-Hill Education, 2013

3. Aurelien Geron, Hands-on Machine Learning with Scikit-Learn &Tensor Flow, O'Reilly, Shroff

4. Publishers and Distributors Pvt. Ltd 2019