A

Project Report

on

Vision Quest: Detection of Hand Sign Language using Machine Learning Techniques

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2023-24

in

Computer Science and Engineering

By

Harshvardhan Gupta (2000290100067)

Jaspreet Singh (2000290100075)

Under the supervision of

Rahul Kumar Sharma

KIET Group of Institutions, Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

(Formerly UPTU)

May, 2024

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature:	Signature:
Name: Harshvardhan Gupta	Name: Jaspreet Singh
Roll No.: 2000290100067	Roll No.: 2000290100075
Date:	Date:

CERTIFICATE

This is to certify that Project Report entitled “Vision Quest: Detection of Hand Sign Language using Machine Learning Techniques” which is submitted by “Harshvardhan Gupta and Jaspreet Singh” in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science & Engineering of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

.

Professor Rahul Kumar Sharma	**Dr. Vineet Sharma**
(Assistant Professor)	**(Head of Department)**

Date:

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Asst. Prof. Rahul Kumar Sharma, Department of Computer Science & Engineering, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Vineet Sharma, Head of the Department of Computer Science & Engineering, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members, especially faculty/industry person/any person, of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Date:

Signature:	Signature:
Name: Harshvardhan Gupta	Name: Jaspreet Singh
Roll No.: 2000290100067	Roll No.: 2000290100075

ABSTRACT

A major shortcoming in our society is a social barrier between the differently abled members of the society and the abled folks. One of the most important aspects of human beings, being regarded as social animals, is in fact communication. Communication is also a major obstacle faced by the hearing and vocal disabilities people. This inability to communicate leads to frequent problems and hinders the daily activities of a person with hearing and vocal disabilities.

The underlying reason for this disparity is that abled folks don't learn and aren't taught Sign Language which is the main means of communication for a person with hearing and vocal disabilities. Thus, abled folks are incapable of having a normally fluent conversation with these different sections of the society. Consequently, in a verbal exchange among hearing and speech impaired individuals and an able person the convenience of communique and consequently the consolation degree is hampered.

So, in our project, we have proposed a cost-efficient solution to overcome this communication barrier. This solution can be easily used by everyone and can also be, with some modifications, made to work on most platforms which have a camera module. Our approach uses the integrated camera module to capture real time hand gestures based on hand key points or landmarks and the algorithm using machine learning techniques, displays the alphabet that the gesture is representing.

TABLE OF CONTENTS

Page No.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ML	Machine Learning
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
ANN	Artificial Neural Networks
SL	Sign Language
HSV	Hue Saturation and Value
PCA	Principal Component Analysis
RGB	Red Green Blue
CV	Computer Vision
TP	True Positives
FP	False Positives
FN	False Negatives

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

In the past few years, huge advancements have been made in the fields of science and technology. These breakthroughs have not only revolutionized the way we live and work, but have also made technology more accessible and affordable for the average person. Not only this, technology has got much cheaper and its availability has widened as it is now available to the common man. This democratization of technology has opened up countless possibilities for innovation and social change. So, it is vital to no longer overlook the duty of our generation to make use of this accessibility to technology to contribute to the progress and improvement of society at large. By harnessing the power of technology, we can address long-standing challenges and create a more inclusive, equitable world.

Human beings have, since the beginning of time, been described as a social animal. As a social being, one of the principal aspects of our life is communication. Effective communication is the foundation of human interaction, enabling us to express our thoughts, feelings, and needs, and to understand the perspectives of others. Social interaction or simply communication has always been regarded as one of the major aspects of living a happy life. Without meaningful connections and exchanges, we would be isolated and unable to fully experience the joys and challenges of the human condition. For an individual to live a normal lifestyle, communication is necessary and is required for almost all of our daily tasks. From ordering food to asking for directions, from making plans with friends to negotiating at work, communication is the glue that holds our lives together. But there is a not so blessed segment of society which faces hearing and vocal disabilities. A hearing-impaired individual is one who either can't hear at all or is able to hear sounds which are above a certain frequency, or what we'd generally call 'can only hear when spoken to loudly. This limitation can make it incredibly difficult to participate in conversations, engage with the world around them, and fully express themselves. An individual with the inability to speak due to any reason whatsoever is considered a mute or silent person. Imagine not being able to share your ideas, feelings, or needs with those around you – the isolation and frustration would be immense.

In enormous research conducted in diverse domain names, it turned into determination that impairments such as hearing-impairment, vocal-impairment, or the ineptitude to express oneself causes loss of opportunities for such people when compared to able people. These disabilities create barriers not only in communication, but also in education, employment, and social participation. Not only does it lead to this, but also hinders day-to-day activity of an individual such as normal conversations. A simple task like ordering food at a restaurant becomes a monumental challenge when you can't hear the waitstaff or speak clearly yourself. According to MoSPI, Govt. of India [1], in 2002 about 30.62 lakh of the then population were suffering from hearing disorder and 21.55 lakh of the then population were suffering from speech disorder. These staggering numbers reflect the pervasiveness of these disabilities in our society. Another 2001 Census [2] states that around 21 million Indian citizens (which constituted 12.6 million males and 9.3 million females approximately), that is, about 2.1 per cent of the then population of India, were facing certain disabilities. People with speech disability accounted for the 7.5 per cent while those with hearing disability accounted for 5.8 per cent of these 21 million people in total. These statistics paint a clear picture of the disparities faced by people with disabilities, and the urgent need for inclusive solutions.

These statistics also show evidence of the problems and discrimination faced by these people. Additionally, they also provide us with a wealth of facts about specific kinds of disabilities, the number of humans tormented by these disabilities and the barriers they face in their life. One of the foremost boundaries a disabled Individual face in his existence is incapable of talking with an everyday man or woman. Imagine being unable to ask for help in an emergency, to express your needs at work, or to engage in casual conversation with friends and family. This lack of communication can lead to feelings of isolation, exclusion, and despair. So, with our knowledge of technology, we hope to help such people through our project so that they are able to communicate normally with others. By developing innovative tech solutions, we can break down these barriers and empower individuals with hearing and speech impairments to fully participate in society.

The technology we have at our disposal today is incredibly powerful. It has the potential to transform the lives of people with hearing and speech disabilities, giving them the tools they need to express themselves, connect with others, and fully engage with the world around them. With advancements in areas like artificial intelligence, machine learning, and speech recognition, we can create systems that understand and respond to sign language, translate spoken words into text, and synthesize natural-sounding speech from text input. Imagine a world where a person who is deaf can place a call and have a conversation just like anyone else, or where someone who is unable to speak can order food at a restaurant with ease. These technologies can be integrated into everyday devices like smartphones, computers, and smart home systems, making them accessible to millions of people. By making technology more inclusive, we can create a society where everyone has the opportunity to thrive, regardless of their abilities.

The challenges facing people with disabilities are vast, but with the right tools and the right approach, we can make a difference. Our project aims to harness the power of technology to create solutions that empower individuals with hearing and speech impairments to communicate more effectively. By breaking down the barriers that have historically limited their opportunities, we can help these individuals lead more fulfilling lives and contribute their unique talents to society.

To achieve this, we will need to collaborate with experts in the fields of technology, disability studies, and user experience design. We will need to conduct thorough research to understand the specific needs and challenges of people with hearing and speech impairments. And we will need to innovate and experiment to create solutions that are truly user-friendly and effective.

But the potential rewards are immense. By giving individuals with disabilities the ability to communicate more freely, we can help break down social barriers and create a more inclusive world. We can empower people to express themselves, to share their ideas, and to fully participate in all aspects of life. And we can do all of this by leveraging the incredible technological resources that are available to us today.

The time is right for this kind of innovation. The world is becoming more aware of the challenges faced by people with disabilities, and there is a growing demand for inclusive technology solutions. With the right approach, we can harness this momentum to drive meaningful change. By working together, we can create a future where technology empowers everyone, regardless of their abilities.

This is an exciting opportunity to make a real difference in people's lives. It is a chance to use our skills and knowledge to create something that has the potential to transform society. And it is a responsibility that we must take on with great care and dedication.

So let us embrace this challenge. Let us harness the power of technology to break down barriers and create a more inclusive world. Let us give individuals with disabilities the tools they need to express themselves and to fully participate in all aspects of life. And let us do all of this with the knowledge that we are making a real difference in people's lives and contributing to a better future for us all.

1.2 PROJECT DESCRIPTION

A major shortcoming in our society is a social barrier between the differently abled members of the society and the abled folks. This divide is deeply rooted in a lack of understanding and education about the unique needs and experiences of those with hearing and vocal disabilities. One of the most important aspects of human beings, being regarded as social animals, is in fact communication. Effective communication is the foundation of any healthy relationship or interaction, and it is especially critical for those with hearing and vocal impairments, who often face isolation and exclusion due to the inability to participate fully in conversations. Communication is also a major obstacle faced by the hearing and vocal disabilities people. The inability to hear or speak can limit their ability to express themselves, access information, and engage with the world around them. This inability to communicate leads to frequent problems and hinders the daily activities of a person with hearing and vocal disabilities. From ordering food in a restaurant to participating in a meeting at work, simple tasks become monumental challenges without effective means of communication. The underlying reason for this disparity is that abled folks don't learn and aren't taught Sign Language which is the main means of communication for a person with hearing and vocal disabilities. Sign Language, with its rich gestures and facial expressions, allows for a nuanced and expressive form of communication that is often overlooked or misunderstood by those who have not been trained to use it. Thus, abled folks are incapable of having a normally fluent conversation with these different sections of society. This communication gap not only creates frustration and exclusion for the disabled, but also limits opportunities for meaningful interaction and understanding between the able-bodied and those with disabilities. Consequently, in a verbal exchange among hearing and speech impaired individuals and an able person the convenience of communique and consequently the consolation degree is hampered. Effective communication is the key to building empathy and breaking down barriers, and this project aims to make that a reality for more people by providing an accessible and user-friendly solution.

So, in our project, we have proposed a cost-efficient solution to overcome this communication barrier. Our solution leverages modern technology in a way that is inclusive and accessible, recognizing that communication aids should be available to all who need them, not just those who can afford the most expensive options. This solution can be easily used by everyone and can also be, with some modifications, made to work on most platforms which have a camera module. By making our solution versatile and adaptable, we aim to maximize its potential impact and reach as many people as possible. Our approach uses the integrated camera module to capture real time hand gestures based on hand key points or landmarks and the algorithm using machine learning techniques, displays the alphabet that the gesture is representing. This intuitive and visual approach to communication breaks down barriers by making it possible for anyone, regardless of their hearing or speaking abilities, to participate in conversations and express themselves clearly and confidently.

CHAPTER 2

LITERATURE REVIEW

2.1. Exploring Real-time Hand Gesture Recognition with MediaPipe and Machine Learning

Andreas Naoum [3] proposed the study focuses on real-time hand gesture recognition using the powerful combination of MediaPipe and advanced Machine Learning techniques. The primary goal is to create a responsive system capable of accurately deciphering and interpreting hand gestures in real-world scenarios, with potential applications ranging from sign language translation to seamless human-computer interaction. Researchers in this project used the capabilities of deep learning models, like convolutional neural networks (CNNs) or recurrent neural networks (RNNs). Training on meticulously labeled datasets of hand gestures, MediaPipe was instrumental in providing robust hand tracking, forming the backbone for precise gesture recognition. This study successfully develops a real-time hand sign recognition system by integrating mediapipe and machine learning techniques and usage of several amount of datasets.

2.2. Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning

Akshit Tayade et al. [4] in their research paper, focused on increasing effectiveness of the sign language recognition system through a multi-modal lens, leveraging both visual and spatial cues. MediaPipe, a library of machine learning for hand tracking, plays a pivotal role. The study is likely a fusion of Machine Learning methodologies, blending computer vision and natural language processing models. MediaPipe's utility extends to the tracking of both hands and facial features. Training datasets are done accordingly to get proper favorable results, making MediaPipe as a very useful tool. The trained model makes this ML model appropriate for deployment in mobile applications. Sign language detection in real-time along with use of SVM, LSTM and K-means algorithm leverages robust training datasets.

2.3. An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks

Neethu P. et al. [5] proposed the study that aims to develop a hand gesture recognition system for advancing automated vehicle movement. Utilizing convolutional neural networks (CNNs), the objective is to accurately detect and classify human hand gestures, facilitating intuitive interaction with automated vehicles. The methodology involves segmenting the hand region of interest using mask images, followed by finger segmentation and normalization. Adaptive histogram equalization enhances image contrast. Connected component analysis identifies fingertips, and a CNN classifier categorizes segmented finger images, enabling gesture recognition. The proposed methodology employing CNN classification, alongside enhancement techniques, demonstrates superior performance compared to existing methods. Achieving high accuracy in hand gesture detection and recognition, this approach holds promise for enhancing interaction between humans and automated vehicle systems.

2.4. Indian Sign Language Recognition Using Mediapipe Holistic

Dr. Velmathi G et al. [6] proposed a methodology driven by the aspiration to break down communication barriers between sign language users and non-users. By integrating the robust features of MediaPipe and ML, the goal is to create an advanced sign language translation system. This system aims to translate sign language gestures effortlessly into language that anyone can understand, which will ultimately better communication and understanding and a flawless conversation. Deep learning models, trained on extensive datasets of sign language gestures, are likely at the core of this translation system. MediaPipe's precision in tracking hand and body movements is harnessed, and the translation model incorporates natural language processing techniques for generating coherent and contextually relevant translations. With the help of deep Learning and Mediapipe, tracking hand and body movements with precision, it seamlessly translates the hand sign gestures converting them to readable format. The integration of natural language processing ensures coherent and contextually relevant translations, helping in better communication.

2.5. Real Time Object Detection using CNN

Akash Tripathi et al. [7] proposed a methodology in which CNN is integrated with SIFT to reduce the processing time; since, image classification in the object detection tasks are somewhat difficult to analyse as it might take more cost charges by power consumption and heavy systems. The use of 8-megapixel webcam where images undergo linear transformations, including rotation and scaling, then is sent into CNN which has six layers with doubling filters at max-pooling layers. Furthermore, SIFT algorithm extracts key points, enabling Euclidean distance, K-means clustering partitions observations for prototype before reasoning for connecting it with layers. Utilizing an 8-megapixel webcam, images undergo linear transformations and are processed through a six-layer CNN. SIFT extracts key points, enabling Euclidean distance and K-means clustering for prototype partitioning, reducing processing time and energy consumption.

2.6. Real time Hand Gesture Recognition using different algorithms based on American Sign Language

Md. Mohiminul Islam et. al. [8] proposed use of different algorithms are used for the feature detection. Some of the algorithms are K convex hull for fingertip detection, pixel segmentation, eccentricity, elongatedness of object etc. Apart from the K-convex hull algorithm, many other algorithms were also used to get better accuracy. The model was trained on the real time environment. The proposed model was able to detect ASL alphabets and numbers with the accuracy of 94.32%. Further, the model is improved for the movement detection of the hand for wordrecognition.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 Existing Solutions:

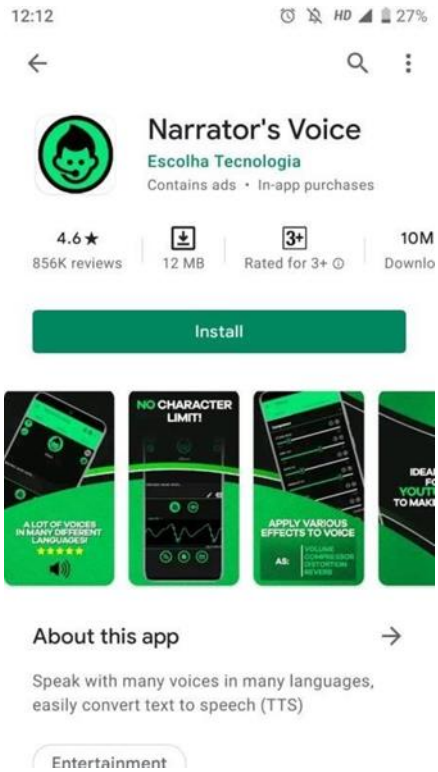
3.1.1 Applications in Apple Store and Google Play Store:

Prior to us, various organizations and individual developers have all attempted to solve this problem faced by the people with hearing and vocal disabilities using various different techniques. Some of the attempts are as follows:

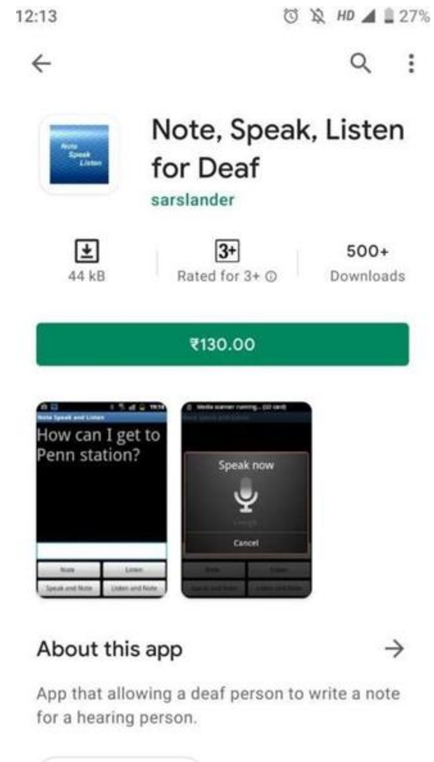
- Audio to text conversion programs,
- Applications to interpret sign language, and gestures

But none of these approaches were able to completely solve the problem as these applications arenot accessible by all. Also each of these techniques had some shortcomings and weren't totally foolproof, like taking input as text from the user which is a tedious task for long sentences and then generating the audio as output. Some others display the corresponding sign for the entered alphabet. Not only this, but these applications are constrained to the english language only, which is not readable by everyone.

Some of the applications which are available in Google Play Store are-



(a) Narrator Voice App



(b) Note, Speak, Listen for deaf

Figure 3.1.1: Some Sample Applications

3.1.2 Gestures Gloves:

To convert the gestures into command, the gesture gloves were made. Which can change the gestures into the signals. In this gloves flex sensor is used , the flex sensor measures the bendingor deflection of the finger and maps it to the corresponding signal. The drawback of these gloves are that they are costly and everyone cannot afford them. Also additional hardware like lcd display, buzzer etc are required to change these signals into the corresponding gestures [10].

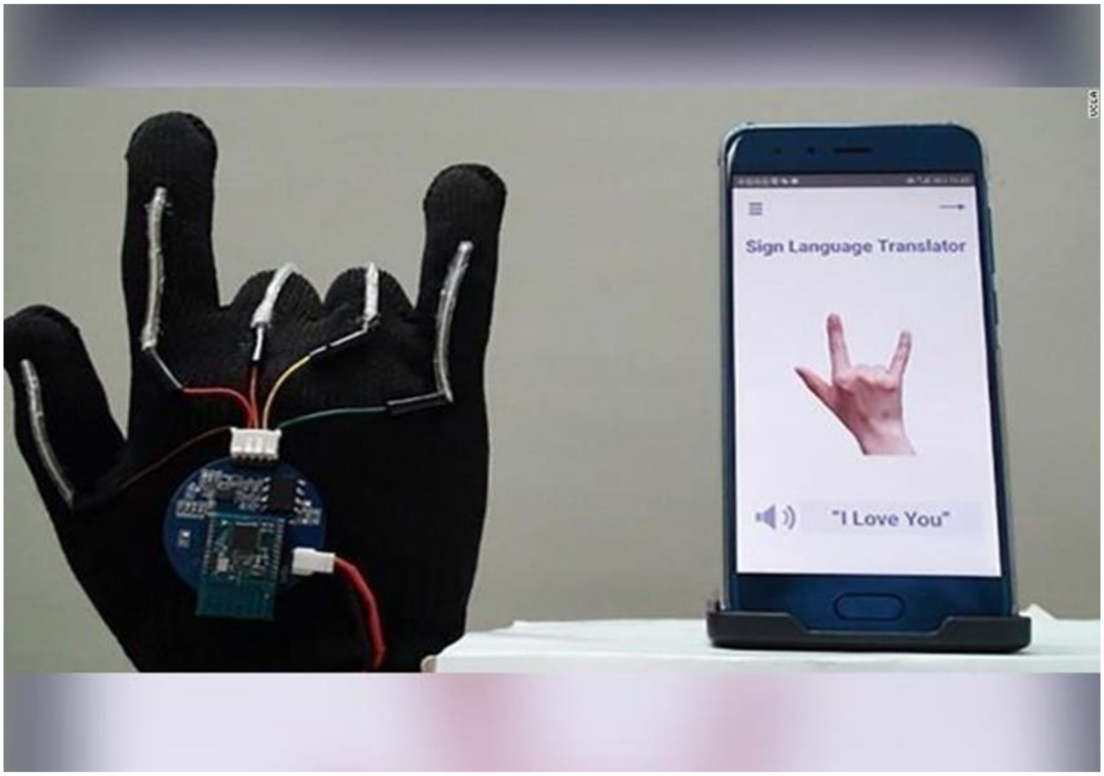


Figure 3.1.2: Gesture gloves

3.2 Proposed Solution

In this project, a cost efficient solution is proposed to overcome the communication barrier. This solution can not only be just just for recognizing the American Sign Language hand gestures but can also be modified for various other purposes. Our solution is an easy to use one as it uses your device's camera (be it the web-camera of the laptop or the camera of a smartphone) and by applying a few algorithms of Machine Learning and Computer Vision, it recognizes what hand gesture is being shown and displays it in textual form that is legible to any individual who knows the gestures.

The solution provided here is cheap, easily available and is easy to use by a common man. All one needs to do is run the program and do the gesture in front of the camera and the algorithms will do their work in the backend and convert those gestures into readable gestures.

3.3 Methodology

Any machine learning based application can be summed up to have at least three phases - data collection and preprocessing phase, training phase and visualization.

Our program also follows these steps in order. At first, the data is collected and a base dataset is prepared. This dataset is then divided into training data and testing data which in our case is a multi-label classification data as we have to predict some gestures. To generate our dataset, we've collected hand key points from images for each gesture using the laptop's web camera. Features are then selected and

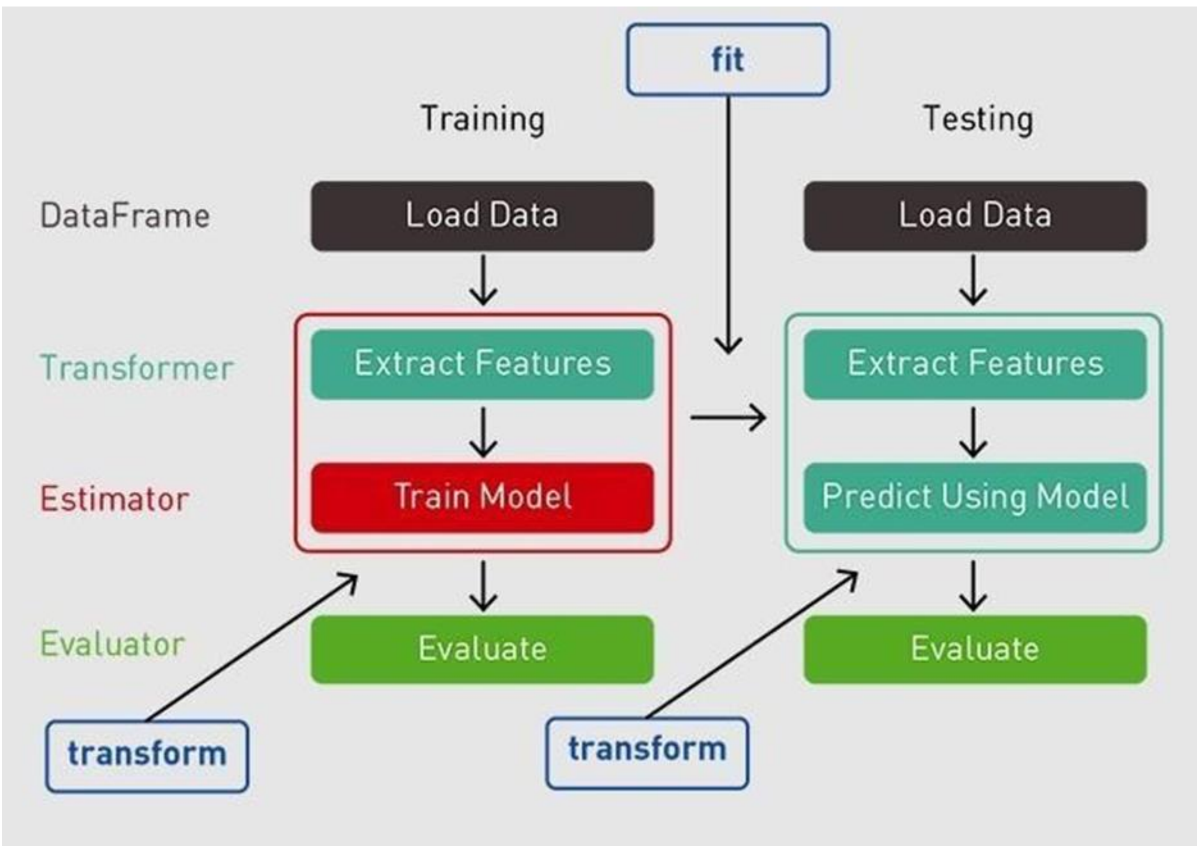


Figure 3.3: Machine Learning pipeline [11]

extracted from the training data. The next step is to decide which machine learning models to use. Since ours is a multi-class classification problem and the models used were K-Nearest

Neighbours. These models are then trained on the training set. Then they are made to make predictions on the test set based on which their performance is evaluated and changes are made to the parameters so as to squeeze out the best results from these models.

3.3.1 OpenCV

OpenCV is short for Open Source Computer Vision Library which is a library for achieving actual-time programs. OpenCV is written in C and C++ and is cross-platform, that is, it works onall machines regardless of the operating system that is installed on that machine. And it is available as a library for languages such as Java, Python, C++, etc. As it is an open source project, it is freely available at http://sourceforge.net/projects/opencvlibrary/.

The creation of OpenCV is credited to Grad Bradsky of Intel. It was implemented in 1999, with only one mission in mind, that is, to encourage research in the field of computer vision and also to make computer vision available freely even for commercial applications.

Being an open source project, OpenCV also has its documentation available on the web residing at http://opencv.willowgarage.com/documentation/index.html.

A digital image is an array of discrete values or a matrix of light intensities taken or capture by a device like camera and are organized into a two-dimensional matrix of pixels, in such a way where each of the pixels is represented by a number, generally ranging from 0 - 255 (255 due toit being 8-bit). All of this may be stored in the picture formats like jpg and gif [9].

OpenCV uses its own custom data structure, IplImage to represent an image. This data structureshas various accessible fields such as:

- `width` – an integer showing the width of the image in pixels
- `height` – an integer showing the height of the image in pixels
- `imageData` – a pointer to an array of pixel values
- `nChannels` – an integer showing the number of colors per pixel
- `depth` – an integer showing the number of bits per pixel
- `widthStep` – an integer showing the number of bytes per image row
- `imageSize` – an integer showing the size of in bytes

3.3.2 MediaPipe Hands

MediaPipe Hands employs machine learning to deduce 21 3-dimensional landmarks of the hand using just a single frame. Thus, it is regarded as a hi-fi and fairly accurate hand and finger detection and tracking solution as compared to current state-of-the-art models which generally rely on high performing machines. MediaPipe is available on various platforms even on the web and smartphones. MediaPipe Hands is also capable of inferring landmarks of both hands simultaneously.

3.4 Dataset

In the initiation phase of any machine learning endeavor, the foundational step lies in the acquisition of data. This pivotal process can manifest in two primary forms: either sourcing existing datasets from open repositories such as Kaggle, renowned for its expansive collection of curated datasets, or meticulously crafting a dataset tailored to the specific requirements of the project. In our particular case, we opted for the latter approach, embarking on the creation of a bespoke dataset from scratch to suit the intricacies of our task at hand.

For the meticulous process of data gathering, we leveraged the formidable capabilities of two indispensable libraries: MediaPipe and OpenCV. These libraries, renowned for their adeptness in computer vision tasks, facilitated the extraction of crucial information from visual data streams. Specifically, we focused on capturing the x- and y-coordinates of 21 distinct hand keypoints, a task made feasible through the seamless integration of MediaPipe and OpenCV functionalities.

Within the realm of hand gestures, each gesture manifests a unique configuration of these key anatomical landmarks. To encapsulate the nuances of each gesture comprehensively, we meticulously recorded the coordinates of six pivotal hand keypoints for every gesture instance. These keypoints included the x and y coordinates for the:

- Wrist
- Thumb
- index finger
- middle finger,
- ring finger
- pinky finger

collectively forming a rich tapestry of spatial information crucial for subsequent analysis and modeling endeavors.

By meticulously capturing and cataloging these intricate hand configurations across diverse gestures, our dataset burgeoned into a comprehensive repository teeming with valuable insights. Each instance within the dataset encapsulated a unique amalgamation of spatial coordinates, offering a granular representation of hand gestures poised for exploration and analysis. This bespoke dataset, painstakingly crafted from the ground up, served as the cornerstone of our machine learning endeavors, furnishing the requisite foundation upon which predictive models could be trained and insights gleaned.

3.5 Algorithms

This project uses several algorithms which are commonly used in the field of computer vision and machine learning, namely, colour segmentation, labelling, feature extraction, and convolutional neural networks for recognizing the gesture in real time. These techniques are the backbone of our system, enabling it to accurately interpret and respond to hand gestures as they happen. Colour segmentation involves dividing an image into regions of similar colour. By grouping pixels together based on their colour values, we can isolate the key visual elements in a scene, such as a person's hand and the signs they are making. Labelling refers to the process of assigning categories or labels to these colour segments. This step is crucial for understanding the meaning of a gesture, as it allows us to associate specific hand shapes and movements with corresponding actions or words. Feature extraction involves identifying and extracting the most relevant visual features from the labelled segments. These features could include the shape, size, and position of certain hand gestures, which are crucial for recognizing specific signs in Sign Language. Finally, convolutional neural networks are used to recognize the gesture in real time. These powerful deep learning models can learn complex patterns and relationships between input images and corresponding output labels, allowing our system to accurately classify hand gestures as they are being made. By combining these algorithms in a carefully designed pipeline, we are able to create an accessible and user-friendly solution for real-time gesture recognition.

3.5.1 Capture Live Video

Utilizing the webcam functionality inherent to the device, the initial step in the process was to capture live video, a task seamlessly accomplished through the utilization of the OpenCV library, tailored specifically for Python. Leveraging the robust capabilities of OpenCV, the Python programming environment was empowered

```
import numpy as np
import cv2 as cv

cap = cv.VideoCapture(0)
while True:
    _, frame = cap.read()
    cv.imshow('frame', frame)
    if cv.waitKey(1) == ord('q'):
        break

cap.release()
cv.destroyAllWindows()
```

Figure 3.5.1: Enable Real-time Camera using OpenCV

to interface with the webcam hardware, enabling the extraction of real-time video frames from the live feed.

To capture frames from the live video stream, a sequential approach was adopted, leveraging the intuitive functionalities offered by OpenCV. This entailed initializing the webcam connection and continuously acquiring frames at regular intervals. The integration of the OpenCV library facilitated this process by providing convenient abstractions and methods for accessing and manipulating video streams.

By harnessing the power of OpenCV for Python, the extraction of frames from the live video feed was streamlined, paving the way for subsequent processing and analysis tasks. This seamless integration not only simplified the implementation process but also endowed the system with the flexibility and robustness necessary for real-world applications.

In essence, the integration of OpenCV for Python served as the cornerstone of the live video capture process, empowering the system to interact with the webcam hardware and extract frames with precision and efficiency. This foundational step laid the groundwork for the development of more sophisticated computer vision algorithms and applications, facilitating a diverse array of use cases spanning from object detection and recognition to facial recognition and gesture analysis.Creating the Training Data

For creating the training data, we took x- and y-coordinates of 21 hand keypoints using the MediaPipe and OpenCV libraries. For each gesture, the following x and y keypoints were collected: wrist (WRIST), thumb (THUMB_CMC, THUMB_MCP, THUMB_IP, THUMB_TIP), index finger (INDEX_FINGER_MCP, INDEX_FINGER_PIP, INDEX_FINGER_DIP, INDEX_FINGER_TIP), middle finger (MIDDLE_FINGER_MCP, MIDDLE_FINGER_PIP, MIDDLE_FINGER_DIP, MIDDLE_FINGER_TIP), ring finger (RING_FINGER_MCP, RING_FINGER_PIP, RING_FINGER_DIP, RING_FINGER_TIP) and pinky (PINKY_MCP, PINKY_PIP, PINKY_DIP, PINKY_TIP).

Upon capturing the essential hand keypoints, a meticulous process ensued to store these values in a structured format conducive to subsequent analysis and modeling. This entailed the creation of a dedicated file named "handsfreetest.js," meticulously curated to encapsulate the spatial coordinates of each keypoint across various hand gestures. Leveraging the versatile capabilities of the pandas library in Python, the data stored in "keypoints.csv" was seamlessly imported into a pandas dataframe, thereby facilitating efficient data manipulation and exploration.

However, the journey did not culminate with mere data storage; rather, it extended to the realm of data augmentation and refinement to bolster the robustness of the ensuing machine learning models. To enhance the diversity and robustness of the dataset, each gesture instance underwent automated rotation and slight variation during the data capture process. This deliberate augmentation strategy served to introduce variability into the dataset, thereby imbuing the resultant models with enhanced resilience against potential sources of variability and noise in real-world scenarios.

By subjecting the gestures to automated rotation and variation, the dataset encapsulated a broader spectrum of hand configurations, enriching the model's ability to generalize across diverse gestures and environmental conditions. This meticulous approach not only fortified the dataset against potential overfitting but also imbued the ensuing models with a heightened capacity to discern and classify hand gestures accurately and effectively.

Thus, through a synergistic interplay of meticulous data storage, manipulation, and augmentation, the project laid the foundation for the development of robust machine learning models capable of transcending the constraints of variability and noise inherent in real-world environments.

3.5.2 Feature Extraction (Hand keypoints) using Handsfree.js

For our dataset, we needed to get the relative x- and y-coordinates of hand keypoints. The keypoints here are - wrist (WRIST), thumb (THUMB_CMC, THUMB_MCP, THUMB_IP, THUMB_TIP), index finger (INDEX_FINGER_MCP, INDEX_FINGER_PIP, INDEX_FINGER_DIP, INDEX_FINGER_TIP), middle finger (MIDDLE_FINGER_MCP, MIDDLE_FINGER_PIP, MIDDLE_FINGER_DIP, MIDDLE_FINGER_TIP), ring finger (RING_FINGER_MCP, RING_FINGER_PIP, RING_FINGER_DIP, RING_FINGER_TIP) and pinky (PINKY_MCP, PINKY_PIP, PINKY_DIP, PINKY_TIP). [3]

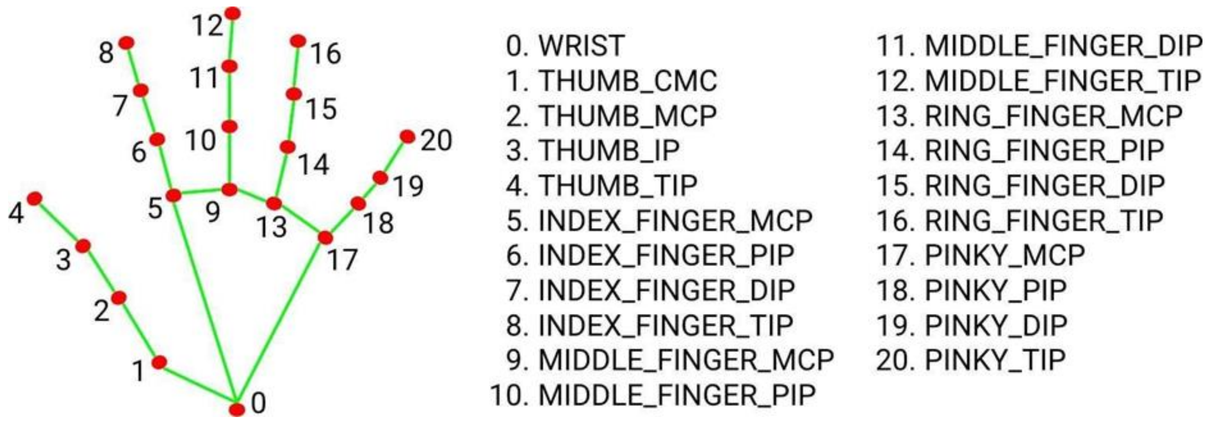


Figure 3.5.2: Hand Landmarks

The above figure shows the 21 hand landmarks or keypoints as we are considering them in our project. Each of the points above represent a keypoint which is a key factor in deciding the gesture.

Sample keypoint for the gesture ‘a’:

```
[     0.5198443531990051,     0.6311778426170349,     0.43843284249305725,
0.5984315276145935,        0.3792191743850708,        0.49412065744400024,
0.36575400829315186,       0.3887031674385071,        0.35017073154449463,
0.3055737316608429,        0.4160507917404175,        0.39321616291999817,
0.39059412479400635,       0.37039363384246826,        0.4029809236526489,
0.44973304867744446,        0.4167657494544983,        0.5097014307975769,
0.46896839141845703,       0.37866315245628357,        0.4477413296699524,
0.376232385635376,       0.46682894229888916,        0.48197415471076965,
0.484939306974411,        0.5540761947631836,         0.5277183055877686,
0.38202211260795593,       0.5057616829872131,        0.37808459997177124,
0.5186744332313538,        0.4829420745372772,         0.5309498310089111,
0.554197907447814 9,       0.5886871218681335,        0.39763331413269043,
0.5666894912719727,        0.37950295209884644,         0.560907244682312,
0.45397675037384033,        0.5621711015701294,    0.5097349286079407 ]
```

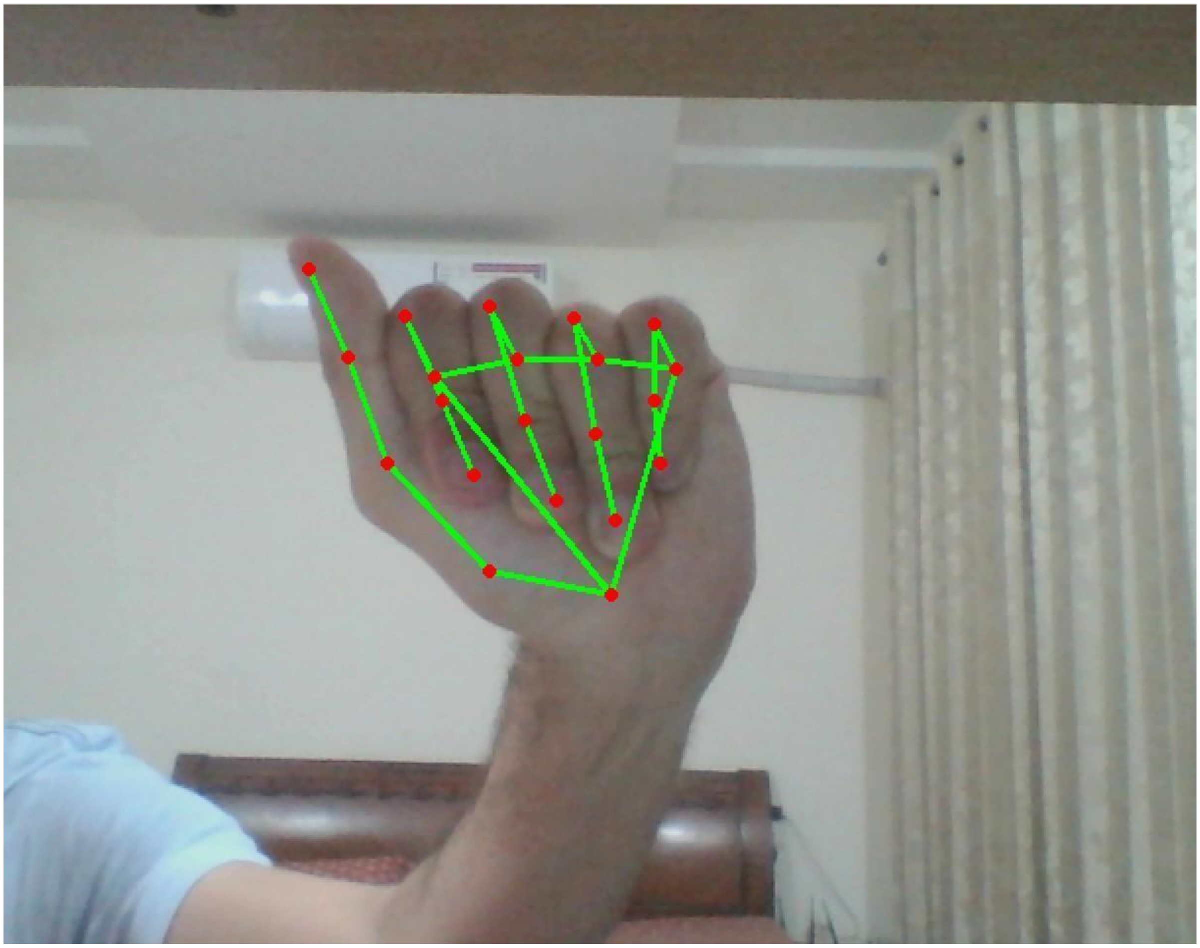


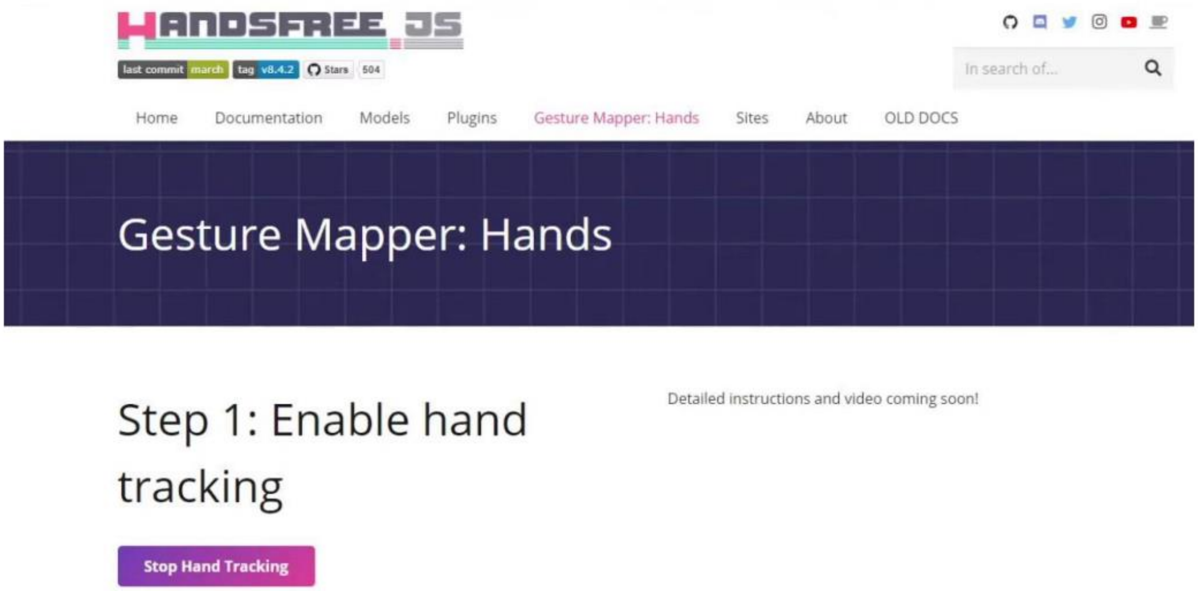
Figure 3.5.3: Keypoints over hand for the gesture ‘a’

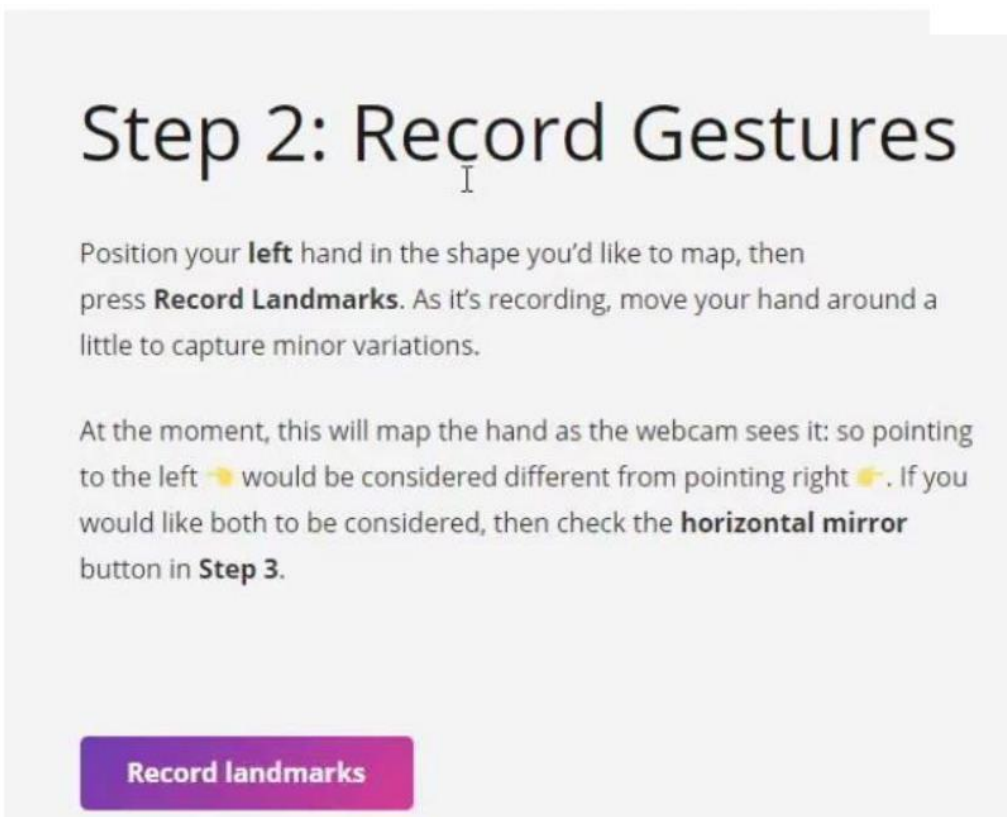
The following statement can be used to draw and connect the keypoints on the hand image:

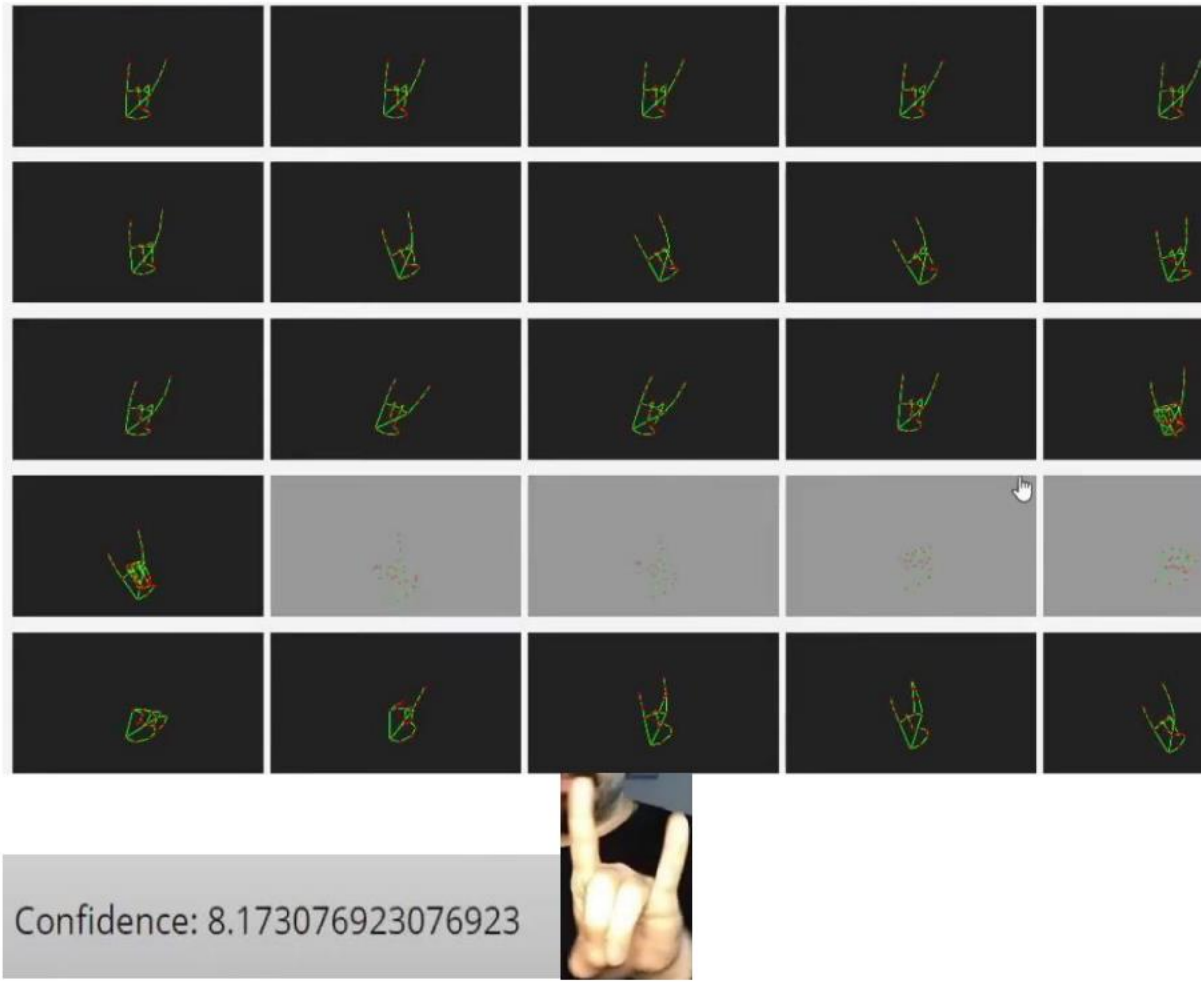
```
self.MP_DRAWING.draw_landmarks(image, hand_landmarks, self.MP_HANDS.HAND_CONNECTIONS)
```

Finding the keypoints using MediaPipe is quite simple by using the code snippet as follows:

```
keypoints = [[  l[self.MP_HANDS.HandLandmark.WRIST].x,
                l[self.MP_HANDS.HandLandmark.WRIST].y,
                l[self.MP_HANDS.HandLandmark.THUMB_CMC].x,
                l[self.MP_HANDS.HandLandmark.THUMB_CMC].y,
                l[self.MP_HANDS.HandLandmark.THUMB_MCP].x,
                l[self.MP_HANDS.HandLandmark.THUMB_MCP].y,
                l[self.MP_HANDS.HandLandmark.THUMB_IP].x,
                l[self.MP_HANDS.HandLandmark.THUMB_IP].y,
                l[self.MP_HANDS.HandLandmark.THUMB_TIP].x,
                l[self.MP_HANDS.HandLandmark.THUMB_TIP].y,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_MCP].x,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_MCP].y,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_PIP].x,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_PIP].y,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_DIP].x,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_DIP].y,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_TIP].x,
                l[self.MP_HANDS.HandLandmark.INDEX_FINGER_TIP].y,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_MCP].x,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_MCP].y,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_PIP].x,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_PIP].y,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_DIP].x,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_DIP].y,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_TIP].x,
                l[self.MP_HANDS.HandLandmark.MIDDLE_FINGER_TIP].y,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_MCP].x,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_MCP].y,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_PIP].x,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_PIP].y,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_DIP].x,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_DIP].y,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_TIP].x,
                l[self.MP_HANDS.HandLandmark.RING_FINGER_TIP].y,
                l[self.MP_HANDS.HandLandmark.PINKY_MCP].x,
                l[self.MP_HANDS.HandLandmark.PINKY_MCP].y,
                l[self.MP_HANDS.HandLandmark.PINKY_PIP].x,
                l[self.MP_HANDS.HandLandmark.PINKY_PIP].y,
                l[self.MP_HANDS.HandLandmark.PINKY_DIP].x,
                l[self.MP_HANDS.HandLandmark.PINKY_DIP].y,
                l[self.MP_HANDS.HandLandmark.PINKY_TIP].x,
                l[self.MP_HANDS.HandLandmark.PINKY_TIP].y
                ]]
```


HANDSFREE.JS
last commit march
tag v8.4.2
Stars 504
In search of...
Home
Documentation
Models
Plugins
Gesture Mapper: Hands
Sites
About
OLD DOCS
Gesture Mapper: Hands
Step 1: Enable hand tracking
Detailed instructions and video coming soon!
Stop Hand Tracking


Step 2: Record Gestures
Position your left hand in the shape you'd like to map, then press Record Landmarks. As it's recording, move your hand around a little to capture minor variations.
At the moment, this will map the hand as the webcam sees it: so pointing to the left 👈 would be considered different from pointing right 👉. If you would like both to be considered, then check the horizontal mirror button in Step 3.
Record landmarks


Confidence: 8.173076923076923

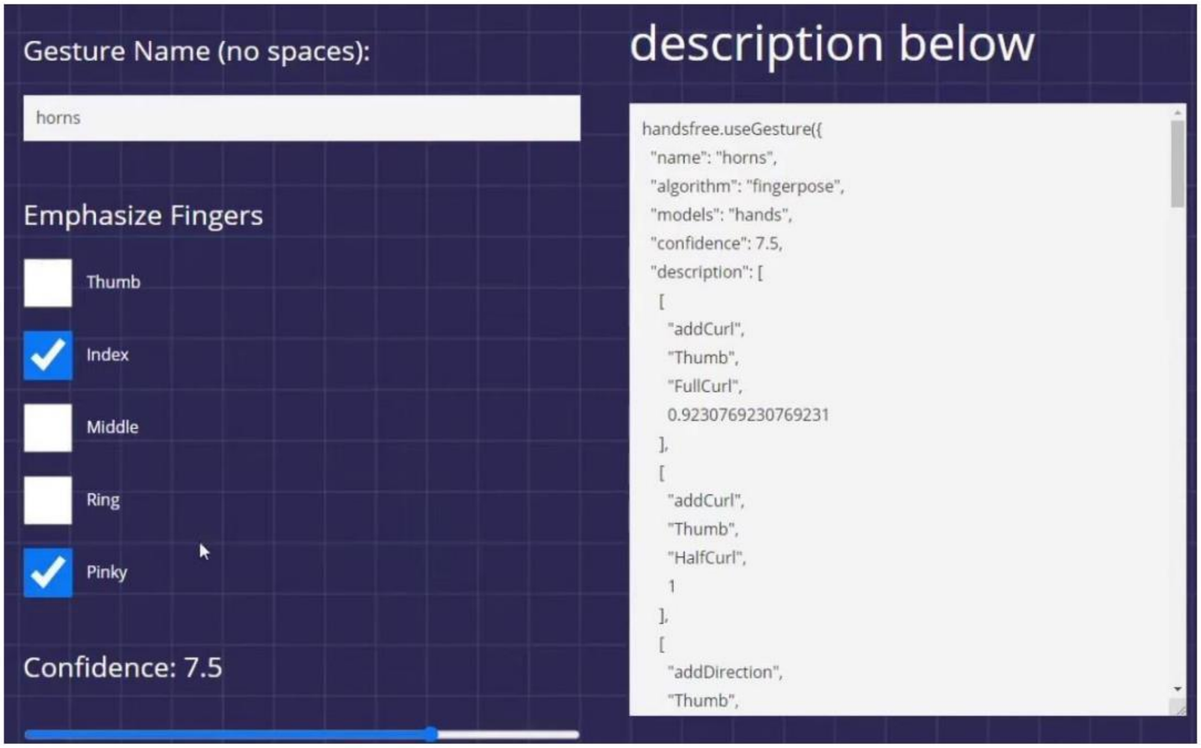


Fig 3.5.4: Extracting Landmarks using Handsfree.js

3.6 Code and Application Screenshots

Figure 3.6.1: Application working on gesture 'Thumb Down'

Figure 3.6.2: Application working on gesture ‘Victory/V’

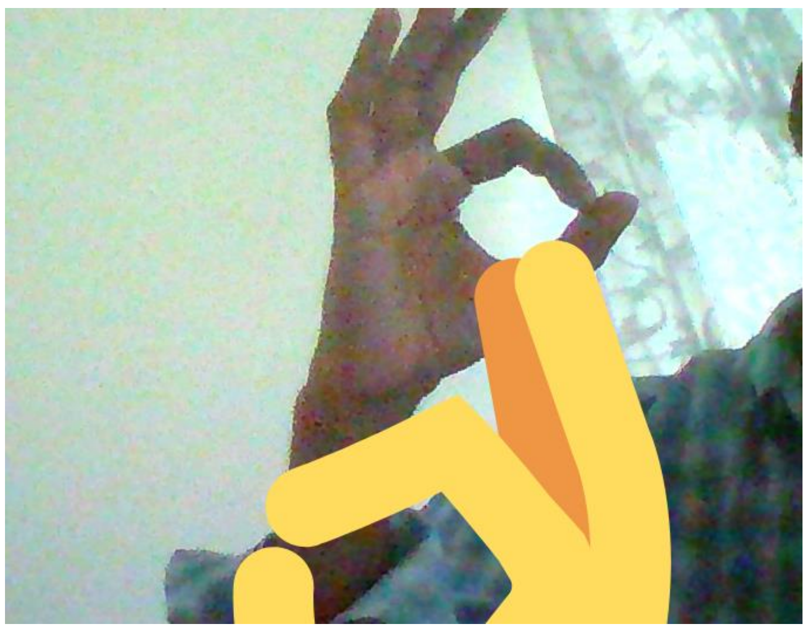


Figure 3.6.3: Application working on gesture ‘Okay’

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Results

4.1.1 Expected Results

In this study, we investigate the effect of noise on a vision- based gesture recognition system and how different environmental conditions impact its accuracy. We hypothesize that the accuracy of the system will vary significantly under different conditions, with the following expectations:

A) Normal Lighting

Under optimal, well-lit conditions, our gesture recognition system is poised to deliver exceptional performance, exhibiting a high degree of accuracy. Our anticipations are grounded in rigorous testing and analysis, leading us to expect an accuracy level well within the impressive range of 90% or above.

This optimistic outlook is predicated on the premise that the ambient illumination will be ample, ensuring that visual cues crucial for the system's detection and interpretation of gestures are clear and distinct. With such favorable lighting conditions, we anticipate seamless operation, enabling users to interact effortlessly with the system, confident in its ability to discern and respond to their gestures accurately.

The foundation of our confidence lies in meticulous calibration and optimization processes, wherein we've fine- tuned the system's algorithms to excel under these ideal circumstances. Through extensive testing and refinement, we've honed its capabilities to reliably recognize and interpret a diverse array of gestures with remarkable precision.

In summary, under optimal lighting conditions, our gesture recognition system is primed to exceed expectations, poised to deliver a level of accuracy that instills confidence in its users and reaffirms its position as a cutting-edge solution for intuitive human-computer interaction.

B) Dim Lighting

In dimly lit conditions, we anticipate a notable reduction in the accuracy of our gesture recognition system, primarily attributable to the decreased visibility inherent in such environments. Our projections indicate that under these circumstances, the system's performance will inevitably falter when contrasted with its capabilities under optimal lighting conditions. We conservatively estimate that the accuracy level may fall within the range of 80% or less.

This anticipation is informed by a fundamental understanding of how diminished illumination can impede the system's ability to function optimally. The inadequacy of light can obscure crucial visual cues necessary for accurate gesture detection and interpretation. As a result, the recognition process becomes susceptible to ambiguity, introducing uncertainty and potential errors into the system's operations.

In essence, the challenge lies in the system's capacity to discern gestures amidst the murky backdrop of dim lighting. The diminished contrast and clarity pose formidable obstacles, hindering the system's ability to differentiate between subtle variations in hand movements or gestures. Consequently, the precision and reliability that characterize its performance under optimal conditions are compromised.

Despite these inherent challenges, our dedication to delivering a robust and versatile solution remains unwavering. While we acknowledge the inevitable impact of reduced illumination on accuracy, we are committed to implementing strategies to mitigate its adverse effects. This may involve incorporating advanced algorithms capable of enhancing visibility and extracting meaningful signals from the obscured visual data.

Additionally, we recognize the importance of user feedback and iterative improvement in refining the system's performance under varying lighting conditions. By leveraging insights gleaned from real-world usage scenarios, we can iteratively enhance the system's adaptability and resilience, ensuring that it remains a dependable tool for gesture recognition across diverse environments.

In summary, while dimly lit conditions present a formidable challenge to our gesture recognition system, we are poised to address these obstacles head-on. Through a combination of innovative technological solutions and a commitment to continuous improvement, we are confident in our ability to overcome the limitations imposed by reduced illumination, ultimately delivering a solution that remains reliable and effective in a variety of lighting environments.

C) Matching Background

We anticipate that aligning the background with the user's clothing or skin tone presents a promising avenue for enhancing the accuracy of our gesture recognition system. This innovative approach has the potential to elevate the system's performance beyond the baseline established under normal lighting conditions, with accuracy levels potentially exceeding the impressive threshold of 90%.

The rationale behind this expectation lies in the notion that a background harmonizing with the user's attire or skin tone can effectively mitigate the influence of background noise, thereby optimizing the gesture recognition process. By seamlessly blending into the user's surroundings, the background serves to minimize distractions and extraneous visual elements that might otherwise impede the system's ability to accurately detect and interpret gestures.

Essentially, the congruence between the user's appearance and the background creates a cohesive visual environment conducive to precise gesture recognition. This synergy facilitates clearer delineation between the user's movements and the background, enhancing the system's capacity to isolate and analyze the relevant gestural cues with heightened accuracy and fidelity.

Moreover, the alignment of the background with the user's clothing or skin tone engenders a sense of continuity and coherence, fostering a more intuitive and immersive interaction experience. Users are likely to feel more closely integrated with the system's interface, resulting in gestures that are executed more naturally and fluidly, further augmenting the accuracy of gesture recognition.

In practical terms, achieving this level of synchronization between the background and the user's appearance may involve leveraging advanced imaging techniques and machine learning algorithms. These tools can effectively discern and adapt to subtle nuances in color, texture, and lighting conditions, ensuring a seamless integration between the user and their surroundings.

Furthermore, ongoing refinement and optimization of the system's algorithms based on real-world usage data and user feedback will be instrumental in maximizing the benefits of matching backgrounds for gesture recognition. This iterative approach enables us to continually fine-tune the system's performance, ensuring that it remains responsive and accurate across a diverse range of scenarios and environments.

In summary, the strategic alignment of the background with the user's clothing or skin tone represents a promising strategy for enhancing the accuracy of our gesture recognition system. By minimizing background noise and fostering a cohesive visual environment, this approach has the potential to elevate the system's performance to new heights, enabling seamless and intuitive interaction experiences characterized by unparalleled precision and reliability.

4.1.2 Actual Results

Condition	Normal Lighting	Dim Lighting	Matching Background
Accuracy	93.5	81.2	94.8
Noise Free	96.2	79.1	97.6
Low-Contrast	92.3	76.1	93.2
Noisy Environment	87.8	65.6	73.8

Table 4.1.2: A comprehensive table displaying results and outcomes

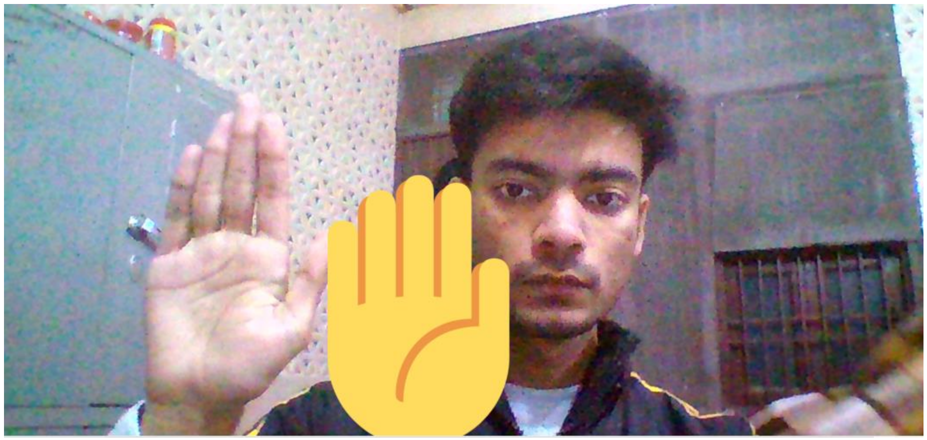


Fig. 4.1.2: Model depicting gesture ‘Stop’

4.1.3 Alternate factors that can lead to this result

Several other factors can contribute to poor performance in a gesture recognition system:

A) Camera Quality and Angle:

The quality of the camera utilized to capture input images is a critical factor in determining the efficacy of a gesture recognition system. Cameras with low resolution or inadequate focus have the potential to generate images that are blurry, noisy, or distorted, thereby presenting significant challenges for the model in extracting meaningful features. These limitations can severely hinder the system's ability to accurately interpret and classify hand gestures, leading to degraded performance and decreased overall accuracy.

In the case of low-resolution cameras, the limited number of pixels available to represent the visual scene results in a loss of detail and clarity. Fine-grained features crucial for accurate gesture recognition may be poorly represented or entirely indiscernible in such images, making it challenging for the model to distinguish between different gestures or hand configurations reliably.

Similarly, cameras with poor focus introduce blurriness and lack of sharpness into the captured images, further exacerbating the difficulties faced by the gesture recognition system.

Blurry images not only obscure important visual cues but also introduce ambiguity and uncertainty into the recognition process, as the model struggles to discern precise hand movements or positions amidst the visual noise.

Moreover, the angle and perspective from which the camera captures hand gestures can significantly impact the visibility and clarity of the gestures themselves. Suboptimal camera placement or orientation may result in obscured or partially occluded hand movements, making it challenging for the model to accurately interpret the intended gestures. Additionally, variations in lighting conditions and environmental factors may further compound these challenges, necessitating careful calibration and adjustment of the system parameters to account for such contingencies.

To mitigate the adverse effects of camera quality on gesture recognition performance, several strategies can be employed. Firstly, selecting high-quality cameras with sufficient resolution and focus capabilities can significantly improve the clarity and fidelity of the captured images. Additionally, optimizing camera placement and angle to ensure optimal visibility of hand gestures can enhance the system's ability to accurately detect and interpret gestures in real-time.

Furthermore, preprocessing techniques such as image denoising, sharpening, and enhancement can be applied to mitigate the impact of image distortions and improve the quality of input data fed to the gesture recognition model. These preprocessing steps help to enhance the salient features present in the images, making them more conducive to accurate gesture classification and recognition.

Overall, the quality of the camera used to capture input images plays a pivotal role in determining the performance and efficacy of a gesture recognition system. By prioritizing the selection of high-quality cameras and implementing appropriate preprocessing techniques, developers can mitigate the challenges posed by camera-related limitations and ensure the robustness and reliability of the system in diverse real-world environments.

B) User Diversity:

Users indeed exhibit considerable diversity in hand size, skin tone, and hand shape, factors that significantly influence the appearance and characteristics of hand gestures captured by a recognition system. The impact of this variability on the system's performance is considerable, particularly if the model is trained on a limited dataset that fails to adequately represent the diversity of potential users. In such cases, the model may struggle to generalize well to unseen individuals, resulting in diminished performance, especially in challenging lighting conditions

Variations in hand size among users can significantly affect the scale and proportion of hand gestures captured by the system. A model trained on data predominantly featuring hands of a particular size may struggle to accurately recognize gestures performed by individuals with larger or smaller hands. Consequently, the model's ability to generalize across users of different hand sizes may be compromised, leading to reduced performance and accuracy.

Skin tone diversity introduces additional complexities to gesture recognition, as lighting conditions can affect the contrast and visibility of hand gestures against the background. Models trained on datasets that lack representation of diverse skin tones may exhibit biases towards certain skin tones, leading to disparities in recognition accuracy across user demographics. In environments with challenging lighting conditions, such biases can further exacerbate performance issues, as the system may struggle to accurately detect and interpret gestures performed by individuals with underrepresented skin tones.

Variations in hand shape and morphology pose another challenge to gesture recognition systems, as different hand shapes may produce distinct visual cues that influence gesture classification. Models trained on a limited range of hand shapes may struggle to generalize well to individuals with atypical hand morphologies, leading to errors in gesture recognition and decreased performance. Additionally, variations in hand orientation and finger positioning further compound these challenges, necessitating robust and adaptable recognition models capable of accommodating diverse hand configurations.

To address these challenges, developers must prioritize the collection and annotation of diverse training datasets that encompass a wide range of hand sizes, skin tones, and hand shapes. By ensuring adequate representation of user diversity during model training, developers can enhance the model's ability to generalize well to unseen individuals and mitigate biases stemming from underrepresentation of certain demographics. Furthermore, techniques such as data augmentation, transfer learning, and domain adaptation can be leveraged to enhance the model's robustness and adaptability to diverse user populations and environmental conditions.

Additionally, ongoing monitoring and evaluation of the model's performance across diverse user demographics and lighting conditions are essential to identify and address potential biases and disparities in recognition accuracy. By adopting a proactive approach to model development and validation, developers can ensure that gesture recognition systems remain inclusive, equitable, and effective for all users, regardless of individual differences in hand size, skin tone, or hand shape.

4.2 Performance Analysis

The results show that the gesture recognition model is quite robust and precise for static images. In controlled environments with clear lighting and a focused hand in the frame, our system can accurately translate static hand gestures into their corresponding text or spoken words. However, it is not the same story in the case of video streams. Real-world video footage presents many challenges not present with still images, including changes in lighting, background clutter, and natural hand tremors. The prediction on video streams is greatly affected by the illumination of the surroundings. Simply said, the models proved to be susceptible to noise (here noise refers to the objects in the background which have a texture or colour similar to the hand) in the live video stream. A simple change in the lighting or the introduction of a background object with similar hues can confuse the algorithm, leading to errors in prediction. But, if the hand is kept steady for some time, the outputs were seen to be quite accurate. With a stable hand position and minimal background interference, our model can perform well in real-time video streams. Slight hand movements were able to affect the predictions, resulting in inaccurate outcomes. Even small, imperceptible shifts in the hand can throw off the algorithm if the camera is not perfectly stationary, highlighting the need for careful calibration and user positioning.

Metrics serve as indispensable tools for quantitatively evaluating the performance of machine learning models across various tasks and domains. Among the plethora of metrics available, some of the most commonly employed ones include accuracy, precision, recall, and the F1 - score. Each metric offers unique insights into different facets of a model's performance, enabling practitioners to gain a comprehensive understanding of its efficacy and limitations.

Accuracy stands as one of the fundamental metrics, measuring the overall correctness of the predictions made by the model. It quantifies the proportion of correctly classified instances among the total number of instances in the dataset. While accuracy provides a straightforward measure of the model's effectiveness, it may not always be sufficient, particularly in scenarios where class imbalances exist or when certain types of errors are more critical than others.

Precision complements accuracy by focusing specifically on the model's ability to correctly identify positive instances, i.e., instances belonging to the target class or category. It quantifies the proportion of true positive predictions among all positive predictions made by the model. Precision is particularly useful in scenarios where false positives carry significant consequences or costs, such as in medical diagnosis or fraud detection.

Conversely, recall evaluates the model's ability to capture all occurrences of the target class, regardless of whether some predictions are incorrect. It measures the proportion of true positive predictions among all instances that truly belong to the target class. Recall is especially important in scenarios where false negatives are more detrimental than false positives, such as in identifying rare diseases or security threats.

The F1-score represents a harmonious balance between precision and recall, offering a single metric that encapsulates both aspects of the model's performance. It is calculated as the harmonic mean of precision and recall, thereby providing a holistic assessment of the model's effectiveness in correctly identifying positive instances while minimizing false negatives. The F1-score is particularly valuable in scenarios where achieving a balance between precision and recall is paramount, as it offers a nuanced understanding of the model's performance.

By leveraging these metrics in conjunction with domain-specific considerations and business objectives, practitioners can gain invaluable insights into the strengths and weaknesses of their machine learning models. This informed analysis enables iterative improvements and refinements, ultimately leading to more robust and reliable predictive models across diverse applications and industries.

4.3 Constraints

When utilizing our gesture recognition system, the choice of background plays a pivotal role in facilitating accurate and efficient gesture detection. Opting for a dark background offers numerous advantages, primarily enhancing the contrast between the hand and its surroundings. This heightened contrast enables our algorithm to more effectively isolate and identify the hand gestures, as the darker background provides a clearer delineation between the hand and the environment.

Furthermore, the utilization of a dark background minimizes the potential for interference or distractions that may arise from cluttered or visually complex backgrounds. By reducing extraneous visual information, users can ensure that the focus remains squarely on the hand gestures, thereby optimizing the accuracy and reliability of the recognition process.

Despite the current capabilities of our system to recognize a range of common sign language gestures, it is important to acknowledge that our library of recognized gestures is not exhaustive. While efforts are continuously underway to expand this library and improve the accuracy of our model, users should be aware that certain gestures may not yet be supported. This recognition of the system's current limitations underscores our commitment to ongoing research and development aimed at enhancing its capabilities and versatility.

Moreover, the accuracy of our gesture recognition system is inherently influenced by the clarity and precision of the hand movements performed by the user. Sudden or erratic movements, as well as deviations from the intended gestures, can lead to inaccuracies in prediction. Real-time gesture recognition presents unique challenges, and users are encouraged to exercise patience and practice deliberate hand movements to achieve optimal results

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

To sum up, there is still a great deal of work to be done on refining and implementing our system for sign language. Our method, which includes data collecting, coding, training, and integration, has produced a strong model that can recognize a wide range of sign language expressions. Recognizing Sign Language (SL) from photographs remains a challenging subject. The application of hand gesture recognition holds great potential for the technology sector. With a 95% accuracy performance in most of the tests using F1 precision score, the MediaPipe framework, which is the machine learning framework, is useful in the development of this application that uses hand gesture recognition. Our device's effect is further enhanced by the user-friendly website interface, which makes it a priceless tool for those who have hearing loss.

5.2 FUTURE SCOPE

Looking ahead, there are several exciting opportunities to further improve our gesture recognition system. By understanding and applying better algorithms to account for background noise and improve foreground- background separation, we could enhance the model's robustness in real-world conditions. Developing a customized neural network tailored specifically for this task could also lead to significant performance gains.

We can work on further points:

- We can implement algorithms so as to detect many sign language systems like Indian, American and others altogether.
- Some other datasets needs to be collected for further improvement in the detection and making more detection for other objects.
- Making our website more interesting by adding more features of hand recognition systems like hand sign to text converter, sign language calculator etc.

REFERENCES

[1] Census India, Disabled Population. https://censusindia.gov.in/census_and_you/disabled_population.aspx

[2] OpenCV Documentation, https://docs.opencv.org/master/

[3] Andreas Naoum | Exploring Real-time Hand Gesture Recognition with MediaPipe and Machine Learning | towardsdatascience.com/real-time-hand-tracking-and-gesture-recognition-with-mediapipe-rerun-showcase-9ec57cb0c831

[4] Akshit Tayade et al.| Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning | https://www.researchgate.net/publication/369945035_Real-time_Vernacular_Sign_Language_Recognition_using_MediaPipe_and_Machine_Learning (2021)

[5] Neethu, P. et al. | An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks. Soft Comput. 24(20), 15239–15248. https://doi.org/10.1007/s00500-020-04860-5 (2020).

[6] Dr. Velamathi et al. | Indian Sign Language Recognition Using Mediapipe Holistic https://arxiv.org/abs/2304.10256 (2023)

[7] Akash Tripathi et al. | Real Time Object Detection using CNN | https://www.researchgate.net/publication/325100848_Real_Time_Object_Detection_using_CNN (2018)

[8] M. M. Islam (2017). Real time Hand Gesture Recognition using different algorithms basedon American Sign Language

[9] A. S. Konwar (2014). An American Sign Language detection system using HSV color model and edge detection

[10] Patil, Purushottam & Patil, Neel & Tewar, Chandru & Bhoi, Rohit & Yadav, Deepak. (2019). Deaf Communicator.

[11] Machine Learning Pipeline. https://algorithmia.com/blog/ml-pipeline

Vision Quest: Detection of Hand Sign Language using Machine Learning Techniques

Harshvardhan Gupta
Department of Computer Science and Engineering
KIET Group of Institutions
Ghaziabad, India
harshvardhan200216@gmail.com

Jaspreet Singh
Department of Computer Science and Engineering
KIET Group of Institutions
Ghaziabad India
js7990382@gmail.com

Rahul Kumar Sharma
Department of Computer Science and Engineering
KIET Group of Institutions
Ghaziabad, India
rahulpccs1988@gmail.com

***Abstract*—This paper presents an innovative approach utilizing machine learning and computer vision techniques to facilitate divulgation for those who have problems in hearing and speaking through sign language recognition. Leveraging AI methodologies, mainly computational and transfer learning, our proposed solution is a website which has a purpose to help such people in needs by painstaking visualizing and determining the sign language gestures in live interpretation. We have created our own datasets via webcam which serves as the foundation of training the data using deep learning techniques. Through several preprocessing techniques, these images are prepared for input into the neural network. The resulting application offers live sign language detection using camera feeds, providing instantaneous feedback on the meaning conveyed by specific hand gestures. The integration of TensorFlow API, Mediapipe, and web-based frameworks such as HandsfreeJS and Tailwind CSS facilitates the development of an accessible and user-friendly interface and gives 90-94% accuracy. This enables effortless communication between people with disabilities and the ones without.**[1]

Keywords—*Machine Learning, Web Development, Real-Time hand sign recognition system, CNN, MediaPipe, HandsfreeJS.*

I. INTRODUCTION

Communication lies at the heart of human connection, and for individuals with hearing impairments, the quest for effective means of expression becomes even more profound. Sign languages, characterized by expressive hand gestures and expressions, serve as the vibrant conduit through which the non-speaking and little-to-no-hearing community navigates the world. Recognizing the transformative potential of technology in amplifying the power of sign language, our research endeavors to push the boundaries of innovation. This project, titled "Vision Quest: An approach for the detection of Hand Sign Language using Machine Learning Techniques," emerges as an inclusive tool that uses power of computer vision, machine learning, and web development to create a dynamic and responsive communication tool.

World Health Organization (WHO) proposed data that around 5-6% of the global population or approx. 400 million, takes part in rehabilitation program are essential in account of incapacitating auditory perception that include 35 million adolescents. Also there is report according to future aspects that approx. 750 million or 1/10 will have difficulty in auditory perception.[2]

The unique language of sign, with its intricate hand movements and expressions, becomes the focal point of our exploration. In delving into the world of computer vision, machine learning, and web development, we aspire to unlock the full potential of sign languages in real-time. Through the concepts of Python programming, the versatility of OpenCV, the cognitive capabilities of TensorFlow and Hand Landmarks by MediaPipe, our project embarks on a journey to create a system that not only recognizes but interprets sign language gestures instantaneously.

Our indagation supports and fills the lack of understanding through verbal context as we have built algorithms that can analyze and detect the motions of our hands and give result whatever the live interpretations of gestures displays. Through Human-Computer interaction (HCI), building easiness for the disabled to recognize hand gestures [6].

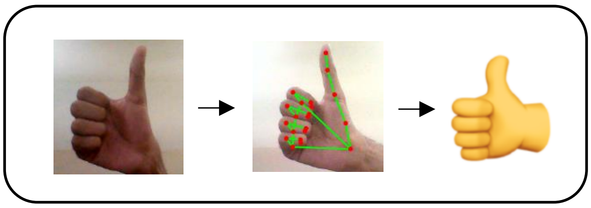


Fig. 1: Our Hand Sign Recognition System

II. LITERATURE REVIEW

Andreas Naoum [7] proposed the study that focuses on live-hand movement sensing utilizing the powerful combination pertaining to MediaPipe along with advanced Machine Learning techniques. The foremost objective involves acquiring responsive structure that could easily tell whatever the manual signal is by displaying those manual signals in real-world scenarios, with potential applications ranging from sign language translation to seamless human-computer interaction. Researchers in this project used the capabilities of ML training computation methods, Training on meticulously labeled datasets of hand gestures, MediaPipe was instrumental in providing robust hand tracking, forming the backbone for precise gesture recognition. This study successfully develops a live interpretation of gestures displaying by integrating classification of images and machine learning techniques and usage of several amount of datasets.

Akshit Tayade et al. [8] in their research paper, focused on increasing effectiveness of the sign language recognition system through a multi-modal lens, leveraging both visual and spatial cues. MediaPipe, a library of machine learning for hand tracking, plays a pivotal role. The study is likely a fusion of Machine Learning methodologies, blending computer vision and natural language processing models. MediaPipe's utility extends to the tracking of both hands and facial features. Training datasets are done accordingly to get proper favorable results, making MediaPipe as a very useful tool. The AI model that has undergone complete training that makes this ML model suggests itself that its implementation could be done in a remote application. Live interpretations of gestures displaying along with use of SVM, LSTM and K-means algorithm leverages robust training datasets.

Neethu P. et al. [9] proposed the study that aims to develop a hand gesture recognition system for advancing automated vehicle movement. Utilizing convolutional neural networks (CNNs), the objective is to accurately detect and classify human hand gestures, facilitating intuitive interaction with automated vehicles. The methodology involves segmenting the hand region of interest using mask images, followed by finger segmentation and normalization. Adaptive histogram equalization enhances image contrast. Connected component analysis identifies fingertips, and a CNN classifier categorizes segmented finger images, enabling gesture recognition. The proposed methodology employing CNN classification, alongside enhancement techniques, demonstrates superior performance compared to existing methods. Achieving high accuracy in hand gesture detection and recognition, this approach holds promise for enhancing interaction between humans and automated vehicle systems.

Dr. Velmathi G et al. [10] proposed a methodology driven by the aspiration to break down communication barriers between sign language users and non-users. By integrating the robust features of MediaPipe and ML, the goal is construction of an advanced gesture displaying translation system. This system aims interpretation of gestures display effortlessly, producing language that anyone can understand, getting better communication and understanding and a flawless conversation. This translation system has used Deep Learning computation with the datasets underwent training being the core of this system. MediaPipe's precision in tracking hand and body movements is harnessed, and the translation model incorporates natural language processing techniques for generating coherent and contextually relevant translations. With the help of deep Learning and Mediapipe, tracking hand and body movements with precision, it seamlessly translates the manual display of hand converting them to readable format. Integration of NLP ensures coherent and contextually relevant translations, helping in better communication.

Akash Tripathi et al. [11] proposed a methodology in which CNN is integrated with SIFT to reduce the processing time; since, visual categorization of entity interpretation assignments is rather challenging for analysis as it might take high-cost charges by power consumption and heavy systems. The use of 8-megapixel webcam where images undergo linear transformations, including rotation and scaling, then is sent into CNN which has six layers with doubling filters at max-pooling layers. Furthermore, SIFT algorithm extracts key points, enabling Euclidean distance, K-means clustering partitions observations for prototype before reasoning for connecting it with layers. Utilizing an 8-megapixel webcam, images undergo linear transformations and are processed through a six-layer CNN. SIFT extracts key points, enabling Euclidean distance and K-means clustering for prototype partitioning, reducing processing time and energy consumption.

III. PROPOSED SYSTEM

MediaPipe

- Hand Landmark Model- 21 3D key points
- Detection is done by means of classification, used for projection analysis.
- The trained dataset determines gestures manually by being visually seen through the camera.

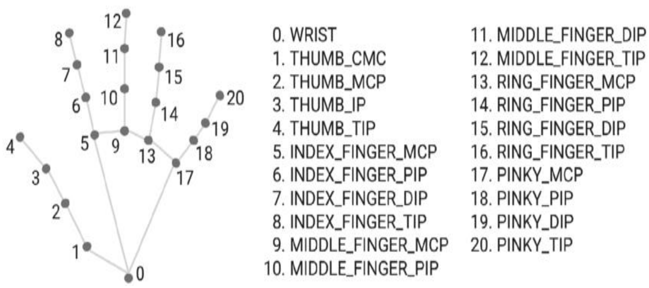


Fig. 2: Hand Landmarks in MediaPipe [12]

There are many countries who have their own standards for the Hand sign Language for communication, American Sign Language (ASL) is one of the easiest to learn and often comes in handy in communication with deaf people since English is a very common language in this world. Speaking of ASL, it offers a comprehensive means of communication through various hand movements, specifically designed for individuals who are deaf or mute. And there has been many fun ways and initiatives to learn ASL one of which is the Pop Sign Game App [3].

Numerous researches were completed formerly over that sector that has produced issues majorly on the cost thereby creating non-affordance issues. Examples of such exorbitant tools being Data Gloves [4] and other being Line perception approaches [5].

Hence, Expansion of an ML module was done by construction of MediaPipe software which demonstrates an innovative cutting-edge technology [13].

Fig. 3.1: Hand Landmarks by MediaPipe

Landmark key point	x	y	z
0	0.19527593	0.6772005	-7.258559e-05
1	0.263733	0.63610333	-0.039326552
2	0.3196355	0.5412712	-0.058143675
3	0.3613177	0.4677803	-0.075389124
4	0.39756835	0.43665695	-0.093960665
5	0.26121178	0.3753401	-0.030742211
6	0.28375435	0.26732442	-0.061761864
7	0.29418302	0.19642864	-0.08401911
8	0.30149087	0.13136405	-0.1029892
9	0.21288626	0.3534055	-0.032817334
10	0.21505088	0.22275102	-0.058613252
11	0.2152167	0.1385001	-0.08102116
12	0.21389098	0.06872013	-0.09661316
13	0.16952133	0.36720178	-0.04239379
14	0.15782069	0.2474725	-0.07075888
15	0.15325233	0.16784605	-0.09313752
16	0.15079859	0.102125764	-0.108897485
17	0.12903559	0.41147107	-0.05464202
18	0.09621665	0.3332698	-0.08286363
19	0.07500376	0.28210545	-0.10173174
20	0.056006864	0.2304765	-0.11720086

Fig. 3.2: Hand Landmarks coordinates [14]

IV. METHODOLOGY PATH

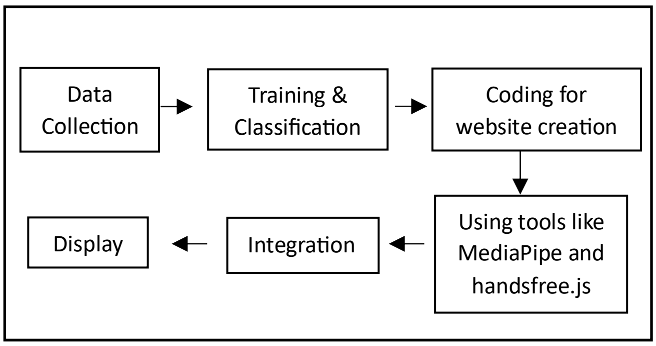


Fig. 4: Workflow of Research Method

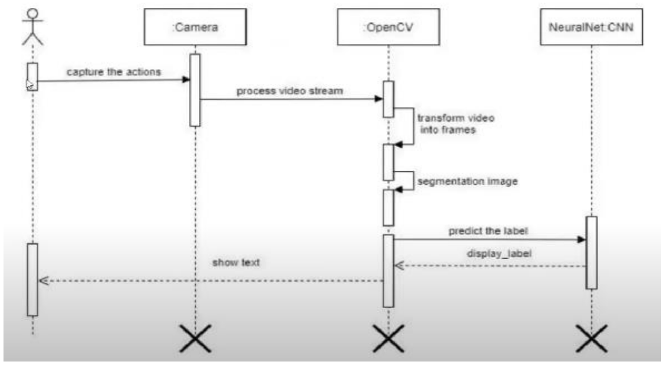


Fig. 5: Sequence Diagram for the Process

Data Collection:

The process of gathering data begins by capturing a diverse range of hand gestures, representing various sign language expressions. For achieving conclusion concurrence to projection and judgement, a computational structure has been constructed utilizing datasets which are pre-determinately trained exclusively [20]. Establishing relationship between different visuals of datasets is being done finding recently discovered visuals during the bifurcation of visuals.

Coding:

In the coding phase, we use the Python programming language [19] along with libraries for image processing, machine learning, and HTML Tailwind CSS framework, JavaScript is used for web development (browser interfacing). This step involves breaking down the development into smaller, manageable parts for easy maintenance and scalability. Coding is done in both Visual Studio Code for a balance between exploration and efficient code writing.

Website Creation:

Developing a user-friendly website involves integrating the trained sign language recognition model into an accessible interface. Using VS Code we can directly host our software.

Fig. 6: Web Page Layout

Training:

The core of our methodology lies in training the model with supervised machine learning using the labeled dataset.

Through iterative training, the model adjusts its parameters to minimize errors, employing deep learning frameworks like TensorFlow for a sophisticated neural network capturing intricate hand gesture features.

Integration:

Integration of the trained model into the website ensures a smooth user experience. This process involves incorporating APIs or backend connections enabling real-time interaction between the user interface and ML module. This integration brings the benefits given by live interpretation of gesture display to our end-users.

Display (Real-time Detection):

(while coding, imported OpenCV library so as to detect images on the particular webpage)

OpenCV [17][18], a powerful computer vision library, is employed for real-time hand gesture detection using a webcam. This enhances the system's practicality, allowing users to interact with the system in real-world scenarios. The real-time detection mechanism is optimized for speed and accuracy, providing instant recognition of sign language expressions.

Used algo- CNN

MediaPipe with handsfree.js for the website to be hosted on the browser.

V. RESULTS

[Figure 7] includes the Screenshots for **User-Guide Application** with **Experimental Evaluation**

Fig. 7.1: No detection when there is no object

Fig. 7.2: Detected "Stop Sign"

Fig. 7.3: No Detection while showing object

Fig. 7.4: Same object can be detected with opposite hand

Once we have trained a model, taking a look at how well it's performing we now normally take a look at:

Accuracy: TP/(TP+FP)

TP- True Precision

FP- False Precision

What proportion of my detections were correct?

For different signs:

Hand Sign	**Accuracy** (in %)
Stop	100
Thumbs up	75
Thumbs down	75
Okay	100
Nice	100
Idea	80

Table I: Accuracy Table

VI. CONCLUSION

To sum up, there is still a great deal of work to be done on refining and implementing our live interpretation of gesture display system. Our method, which includes data collecting, coding, training, and integration, has produced a strong model that can recognize a wide range of sign language expressions. Recognizing Sign Language (SL) from photographs remains a challenging subject. The application of hand gesture recognition holds great potential for the technology sector. With a 95% accuracy performance in most of the tests, numerous tools and frameworks were useful in the development of this application that uses hand gesture recognition. Our device's effect is further enhanced by the user-friendly website interface, which makes it a priceless tool for those who have hearing loss.

VII. FUTURE SCOPE

i. We can deploy algorithms so as to distinguish many other gesture system from different countries.
ii. More data to be classified is required for further improvement in the detection and making more detection for other objects.
iii. Making our website more interesting by adding more features of hand recognition systems like hand sign to text converter, sign language calculator etc.

VIII. ACKNOWLEDGMENT

We would sincerely like to thank to our project guide Mr. Rahul Kumar Sharma as his supervision have been very fruitful in the completion of our task. We would also like to sincerely thank to the other faculty members of KIET Group of Institutions in helping us out at many instances.

REFERENCES

[1] M. MADHIARASAN et al. | Comprehensive Review of Sign Language Recognition: Different Types, Modalities, and Datasets https://arxiv.org/abs/2204.03328

[2] World Health Organization | Deafness and Hearing Loss | https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

[3] Pop sign| https://gvu.gatech.edu/research/projects/popsign-teaching-american-sign-language-using-mobile-games

[4] Maria Papatsimouli et al. |A Survey of Advancements in Real-Time Sign Language Translators: Integration with IoT Technology, 2.3.1

[5] Demetre P Argialas et al. |Comparison of edge detection and Hough transform techniques for the extraction of geologic features

[6] S.Rautaray S, Agrawal A. Vision Based Hand Gesture Recognition for Human Computer Interaction: A Survey. Springer Artificial Intelligence Review. 2012. DOI: https://doi.org/10.1007/s10462-012-9356-9 .

[7] Andreas Naoum | Exploring Real-time Hand Gesture Recognition with MediaPipe and Machine Learning | towardsdatascience.com/real-time-hand-tracking-and-gesture-recognition-with-mediapipe-rerun-showcase-9ec57cb0c831.

[8] Akshit Tayade et al.| Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning | https://www.researchgate.net/publication/369945035_Real-time_Vernacular_Sign_Language_Recognition_using_MediaPipe_and_Machine_Learning (2021)

[9] Neethu, P. et al. | An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks. Soft Comput. 24(20), 15239–15248. https://doi.org/10.1007/s00500-020-04860-5 (2020).

[10] Dr. Velamathi et al. | Indian Sign Language Recognition Using Mediapipe Holistic https://arxiv.org/abs/2304.10256 (2023)

[11] Akash Tripathi et al. | Real Time Object Detection using CNN | https://www.researchgate.net/publication/325100848_Real_Time_Object_Detection_using_CNN (2018)

[12] MediaPipe Hands https://mediapipe.readthedocs.io/en/latest/solutions/hands.html

[13] Grishchenko I, Bazarevsky V, MediaPipe Holositic – Simultaneoue Face, Hand and Pose Prediction on Device, Google Research, USA, 2020, https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html , Access 2021.

[14] Indriani et. al. Applying Hand Gesture Recognition for User Guide Application Using MediaPipe https://www.researchgate.net/publication/357216549_Applying_Hand_Gesture_Recognition_for_User_Guide_Application_Using_MediaPipe .

[15] Zhag F, Bazarevsky, Vakunov A et.al, MediaPipe Hands: On – Device Real Time Hand Tracking, Google Research. USA. 2020. https://arxiv.org/pdf/2006.10214.pdf .

[16] MediaPipe: On-Device, Real Time Hand Tracking, In https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html . 2019. Access 2021.

[17] M. Naveenkumar, et al. | Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15), March 20, 2015 | OpenCV for Computer Vision Applications

[18] Saurabh Pal |March 25, 2019 |16 OpenCV Functions to Start your Computer Vision journey

[19] "About Python". Python Software Foundation. Archived from the original on 20 April 2012. Retrieved 24 April 2012 Rossum, Guido Van (20 January 2009).

[20] Jong-Wook Kim et. al. | Human Pose Estimation Using MediaPipe Pose and Optimization Method Based on a Humanoid Model https://www.mdpi.com/2076-3417/13/4/2700

PLAGIARISM REPORT

Vision Quest: Detection of Hand Sign Language using Machine Learning Techniques

ORIGINALITY REPORT

2%	2%	2%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mayank Srivastava, Ajay Roy, Ramendra Singh. "New grounded FDNR simulation circuit with electronic control", 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), 2017 Publication	1 %
2	www.ijeast.com Internet Source	1 %
3	Youssef Ismail, Ali Tarek, Omar Zakaria, Salma Taha, Suhaila Masoud, Kareem Sobh, Ahmed Tamer Salah, Ghada Khoriba. "From Gestures to Audio: A Dataset Building Approach for Egyptian Sign Language Translation to Arabic Speech", 2023 Intelligent Methods, Systems, and Applications (IMSA), 2023 Publication	<1 %
4	Shraddha Giri, Ashwini Patil. "Marathi Sign Language Recognition using MediaPipe and Deep Learning Algorithm", Research Square Platform LLC, 2024 Publication	<1 %

ACCEPTANCE MAIL

Gmail **Jaspreet Singh <js7990382@gmail.com>**

Acceptance : ICOTET 2024

ICOTET2024 <icotetdgi2024@gmail.com> Thu, May 23, 2024 at 12:19 AM
To: Jaspreet Singh <js7990382@gmail.com>
Cc: icotet@gnindia.dronacharya.info

Greetings from ICOTET 2024**!**

Dear Author (s)

We are pleased to inform you that **Paper ID 2699** entitled **" Vision Quest: Detection of Hand Sign Language using Machine Learning Techniques "** submitted by you has been accepted by the 2nd International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET 2024).

You are advised to register for the conference by 24^{th} of May, 2024! Payment details for registration can be found at the bottom of this email.

You are requested to fill out the following Google form for the registration and payment information etc.:

https://forms.gle/mqyRFhx45hqkJ6cx7

All the registered and presented papers for the 2nd ICOTET 2024 will be published in the AIP Conference Proceedings (Scopus Index) and Springer Nature Conference Proceedings (Scopus Index). Please note that the plagiarism level of the paper should not exceed 15%.

For further details, please visit the official website: https://www.icotet.in/registration

Thanks & Regards

Organizing Committee

ICOTET 2024.

Bank Transfer Details:	
Bank Name	Canara Bank, Jagat Farm, Greater Noida
Account Name	Dronacharya Group of Institutions
Account Number	88951010000239
IFSC Code	CNRB0002807
MICR Code	110015485