A PROJECT REPORT ON

SIGN LANGUAGE DETECTION

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

NIKHIL KULKARNI	71914255F
SHIVALI MATE	71914303K
ATHARVA KULKARNI	71914023E

“ येथे बहुतांचे हित ”

DEPARTMENT OF COMPUTER ENGINEERING

MARATHWADA MITRA MANDAL'S COLLEGE OF ENGINEERING,

KARVENAGAR, PUNE-411052

SAVITRIBAI PHULE PUNE UNIVERSITY
2021-2022

“येथे बहुतांचे हित”

CERTIFICATE

This is to certify that the project report entitles

“SIGN LANGUAGE DETECTION”

Submitted by

NIKHIL KULKARNI	71914255F
SHIVALI MATE	71914303K
ATHARVA KULKARNI	71914023E

is a bonafide student of this institute and the work has been carried out by him under the supervision of **Prof. S. B. Jadhav** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

(Prof. S. B. Jadhav)
Guide,
Department of Computer Engineering
(12, Sentence case)

(Dr. Kalpana Thakre)
Head,
Department of Computer Engineering

(Dr. Vijay N. Gohokar)
I/C Principal, MMCOE

Marathwada Mitra Mandal's College of Engineering, Karvenagar, Pune – 411052

Place: Pune
Date:

ACKNOWLEDGEMENT

We are very thankful to all the teachers who have provided us valuable guidance towards the completion of this on **“Sign Language Recognition’’** as part of the syllabus of bachelor’s course. We express our sincere gratitude towards cooperative department who has provided us with valuable assistance and requirements for the system development.

We are very grateful and want to express our thanks to **Prof. Shailaja Jadhav** for guiding us in right manner, correcting our doubts by giving us their time whenever we required, and providing their knowledge and experience in making this project.

We are also thankful to the HOD of our Computer Engineering department **Prof. Dr. Kalpana Thakre** for his moral support and motivation which has encouraged us in making this project.

The acknowledgement will be incomplete if we do not thank our Principal **Dr. Vijay N. Gohokar** who gave us his constant support and motivation which has been highly instrumental in making our project.

NAME OF THE STUDENTS:

NIKHIL KULKARNI

SHIVALI MATE

ATHARVA KULKARNI

ABSTRACT

Conversing to a person with hearing disability is always a major challenge. Sign language has indelibly become the ultimate panacea and is a very powerful tool for individuals with hearing and speech disability to communicate their feelings and opinions to the world. It makes the integration process between them and others smooth and less complex. However, the invention of sign language alone is not enough. There are many strings attached to this boon. The sign gestures often get mixed and confused for someone who has never learnt it or knows it in a different language. However, this communication gap which has existed for years can now be narrowed with the introduction of various techniques to automate the detection of sign gestures.

So, we would like to present an application to mitigate this issue and reduce the dependencies on third parties like translators. We have studied different approaches taken and discussed them. Recognition of hand signs is a popular topic since the 90s and various findings have been uncovered in the following years. We have written about popular methods for this. For this application to work have to weigh in feature extraction, data acquisition, pre-processing and modification of data, feature extraction, segmentation, object detection, and predictions using a machine learning model.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATION	ILLUSTRATION
ASL	American Sign Language
HMM	Hidden Markov Model
SVM	Support Vector Machine
CNN	Convoluted Neural Network
ANN	Artificial Neural Network

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 OVERVIEW

Hearing loss can range from mild to severe and can be attributed to anything including age, occupational hazards such as military combat and exposure to excessive noise at work, injury to the ear, infections and more. In addition, hearing loss does not just make things quieter, it can distort sounds and cause normal speech to sound jumbled and garbled. There are different sign languages present for different languages like ISL for Indian sign language, ASL for American sign languages, DSL for Devanagari sign languages.

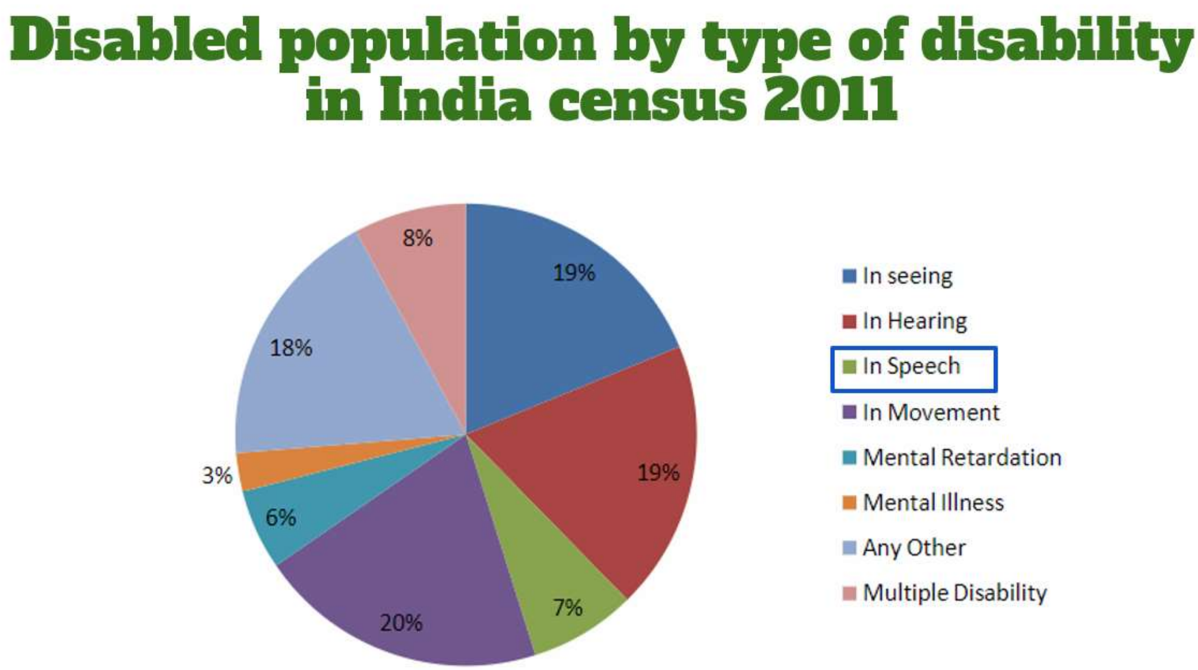


Figure 1.1: Disabled population by type of disability in India census 2011

As majority of people do not understand sign language, deaf people suffer through many problems in communicating with the mass. They are heavily relied on translators to interpret the sign language and act as a bridge between them. This not only halts the progress of physically disabled but also leads to heavy dependency on translators which is not optimal. As translators do charge hefty amount and are not always available at any hour this option is heavily flawed and does need another alternative which can be dependable.

Hence, we propose to develop an app which can detect the hand signs of a person and then convert it into sentences which we can easily read/listen. This will not only dispose any dependency on translator but will also cut cost of translators. We plan to make this app available on every smartphone so that no person feels disconnected with rest of the world. Hand sign recognition includes phases like image capturing, pre-processing, extraction of region, feature extraction and feature matching for recognition which will be further discussed.

1.2 MOTIVATION

Deaf people have difficulty in communicating with people. The only medium through which deaf people can communicate is through hand signs. Unfortunately, many people do not have knowledge about hand signs and thus create a problem for clear communication. Hence, we propose an application to convert hand signs into written language.

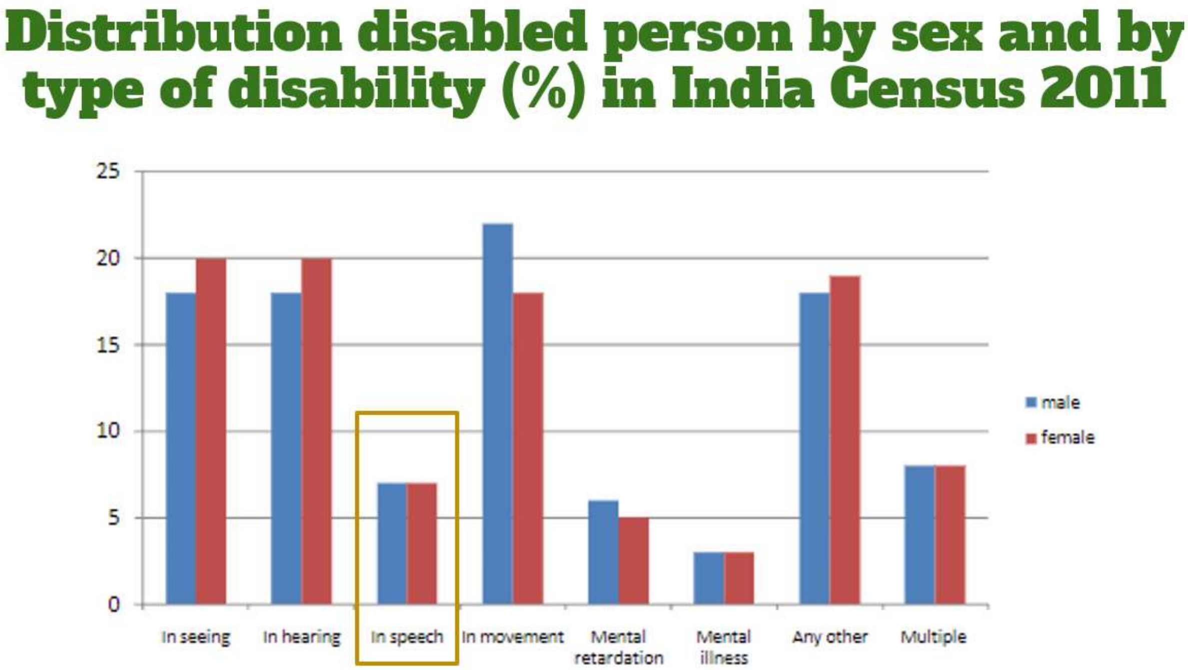


Figure 1.2: Distribution disabled person by sex and by type of disability (%) in India

1.3 PROBLEM DEFINITION

Given a hand gesture, implementing such an application which detects predefined American sign language (ASL) in a real time through hand gestures and providing facility for the user to be able to store the result of the character detected in a txt file, also allowing such users to build their customized gesture so that the problems faced by persons who aren't able to talk vocally can be accommodated with technological assistance and the barrier of expressing can be overshadowed.

1.4 PROJECT SCOPE

1.4.1 Introduction

We will be using desktop application for the same and will use computer/laptop camera for video stream. The basic requirement like functional camera, decent lighting goes without saying. Also, we need to make sure that video is getting streamed on the app.

1.4.2 Project Scope

This project would consist of an app which can be then used for dumb people to help them communicate. If correct improvements are made then it has the capacity to be used at globally. But being able to close gap between communications is a big achievement. The accuracy of this app is very high so it won’t be giving any wrong output which is major advantage.

CHAPTER 2: LITERATURE SURVEY

Sr. No	Paper Name	Year	Author	Publisher	Review
1.	Sign Language Recognition	2011	Helen Cooper, Brian Holt,and Richard Bowden	Springer	• Found a way to reduce the expense of recognition and proposed a multimodal for recognition • Proposed changes to long used HMMs • Talked about drawbacks of converting alphabets to language
2.	Sign language recognition using image-based hand gesture recognition techniques	2016	Ashish S. Nikam, Aarti G. Ambeka	IEEE	• Made several progress in image-based recognition for hand gestures • Also proposed a way where points were assigned to hand and thenused to locate and track them • Made a desktop application using the following methodology
3.	Real-Time American Sign Language Recognition fromVideo Using Hidden Markov Models	1997	THAD STARNER AND ALEX PENTLAN D	Springer	• They found a theoretical approach to recognize basic hand signs whichincluded pronoun, noun, verb, color. • They also figured to detect the hand through video stream and wereable to capture movements of hand. • Devised four step approach to analyze hand signs through Markov Models. [HMM] • Improvements like more words and better approach to capture handsigns through video stream.

4.	Sign Language Recognition Using Convolutional Neural Networks	2015	Lionel Pigou	Springer	• Made a model for recognition of hand signs by using CNN • Achieved validation score of 0.789675 and accuracy of f 91.70%(8.30% error rate) • Study on ReLUs and their effect on model
5.	Real time finger tracking and contour detectionfor gesture recognition usingOpenCV	2016	Ruchi Manish Gurav	IEEE	• Proposed methodology to capture images from OpenCV • Suggested filtering of segmentation and classification to classify • Got an accuracy of 92% with Convex Hull and 70% accuracy with AdaBoost.

CHAPTER 3: SOFTWARE REQUIREMENTS SPECIFICATIONS

3.1 ASSUMPTIONS AND DEPENDENCIES

3.1.1 Assumptions

1. The user should be familiar with ASL so that he can correctly show the sign and get to know whether the prediction done by the system is correct.
2. It is assumed the video stream will get proper inputs with respect to the model.
3. The camera on laptop/computer is assumed to be working so that it can take video stream.

3.1.2 Dependencies

1. The user should sit in a well-lit room for better predictions.
2. It is dependent on if the person is using this app.
3. It is also dependent if person is using ASL and no other hand sign language.
4. It is dependent on current hand signs and it will take time to update it.

3.2 FUNCTIONAL REQUIREMENTS

3.2.1 Functional Requirement 1

The foremost functional requirement includes the detection of hand sign(s). In this section, the model will detect the hand signs obtained through video stream.

3.2.2 Functional Requirement 2

The next functional requirement includes storing of new hand sign(s)/gesture(s) which are not present in the database.

3.2.3 Functional Requirement 3

We propose to make a real time application using flask.

3.3 EXTERNAL INTERFACE REQUIREMENTS

3.3.1 User Interface

The requirements section of hardware includes minimum of 100 GB hard disk and 4GB RAM with 1 GHz or higher speed. The primary requirements include a memory of 1 GB for the application of python. User interface of this program is the common windows interface, nothing additional is required. The System user interface should be intuitive, such that 99.9% of all new system users are able to use Proposed System without any assistance. User registered in to system and log in then input to system as gesture and wait for response for system.

3.3.2 Software Interface

The computer this software going to be install need to have python IDE equal or above, Windows 7. On that Windows platform python, python version 3.* will be installed and that will be the platform the particular software will be run. There will be an python IDE data transmission .

3.3.3 Communication Interface

Communication architecture defines the frequency and fidelity of information flow between individuals in your organization. It helps structure how and when you communicate, both within a team and cross-functionally. The specific tactics are unique to each organization, but it requires proactive thought and investment.

3.4 NON-FUNCTIONAL REQUIREMENTS

3.4.1 Performance Requirement

- System can produce results faster on 4GB of RAM.
- It may take more time for peak loads at main node.
- The system will be available 100% of the time. Once there is a fatal error, the system will provide understandable feed back to the user.

3.4.2 Software Quality Attributes

The system considers following non-functional requirements to provide better functionalities and usage of system.

1. **Availability:**
   The system is available during 24 hours of a day.

2. **Usability:**
   The system is designed keeping in mind the usability issues considering the end-users who are developers/programmers. It provides detailed help which would lead to better and faster learning. Navigation of system is easy.

3. **Consistency:**
   Uniformity in layout, screens, Menus, colors, scheme, format.

4. **Performance:**
   The performance of the system is fast and as per user requirement. From this system gives expected outcome in less time and less space since efficiency is higher. Speed is totally depending on the response of the database and connection type.

5. **Extendibility:**

   Prevention in the system is done by the system only, in which we make changes in the system later on.

6. **Reusability:**

   The hand signs stored in the system can be used again for detection of hand signs without the need for creating the same hand signs for detection purpose.

3.5 SYSTEM REQUIREMENTS

3.5.1 Database Requirements

A proper database needs to be created containing all the alphabets along with their corresponding ASL sign. We have created a database for each alphabet by taking 10* frames for each of the alphabet. Hence, even if there is a slight change in the position of fingers of the user, the system will be able to predict the hand sign correctly thus giving a better accuracy.

3.5.2 Software Requirements

1. Operating System: Windows 10
2. Language: Python
3. Packages: TensorFlow, Keras
4. Libraries: OpenCV, pyttsx3

3.5.3 Hardware Requirements

1. Processor: Intel® Core ™ i5-8265U CPU @ 1.60GHz 1.80 GHz
2. Installed RAM: 8 GB
3. System type: 64-bit operating system, x64-based processor

3.6 ANALYSIS MODELS

3.6.1 SDLC Model:

We are using waterfall model:

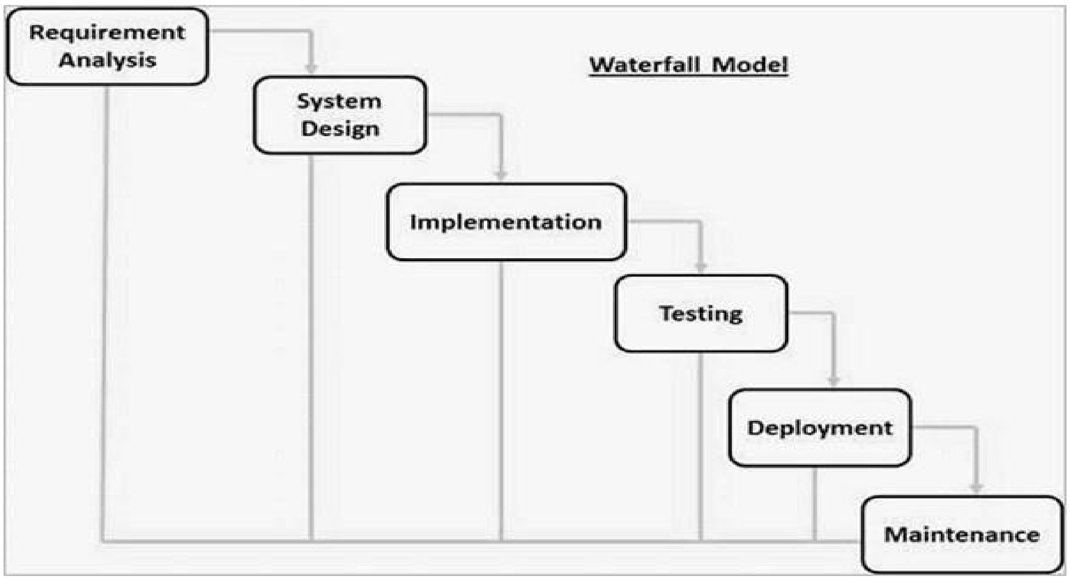


Figure 3.6: SDLC Model

1. **Requirement gathering and analysis:**

   In this step of waterfall we identify what are various requirements are need for the project such are software and hardware required, database, and interfaces.

2. **System Design:**

   In this system design phase we design the system which is easily understood for end user i.e. user friendly. We design some UML diagrams and data flow.

3. **Implementation:**

   In implementation phase of our project we have implemented various module required of successfully getting expected outcome at the different module levels. With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

4. **Testing:**

   The different test cases are performed to test whether the project module are giving expected outcome in assumed time. All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

5. **Deployments of System:**

   Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.

6. **Maintenance:**

   There are some issues which come up in the client environment. To fix those issues patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment. All these phases are cascaded to each other in which progress is seen as flowing steadily downwards like a waterfall through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

CHAPTER 4: SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

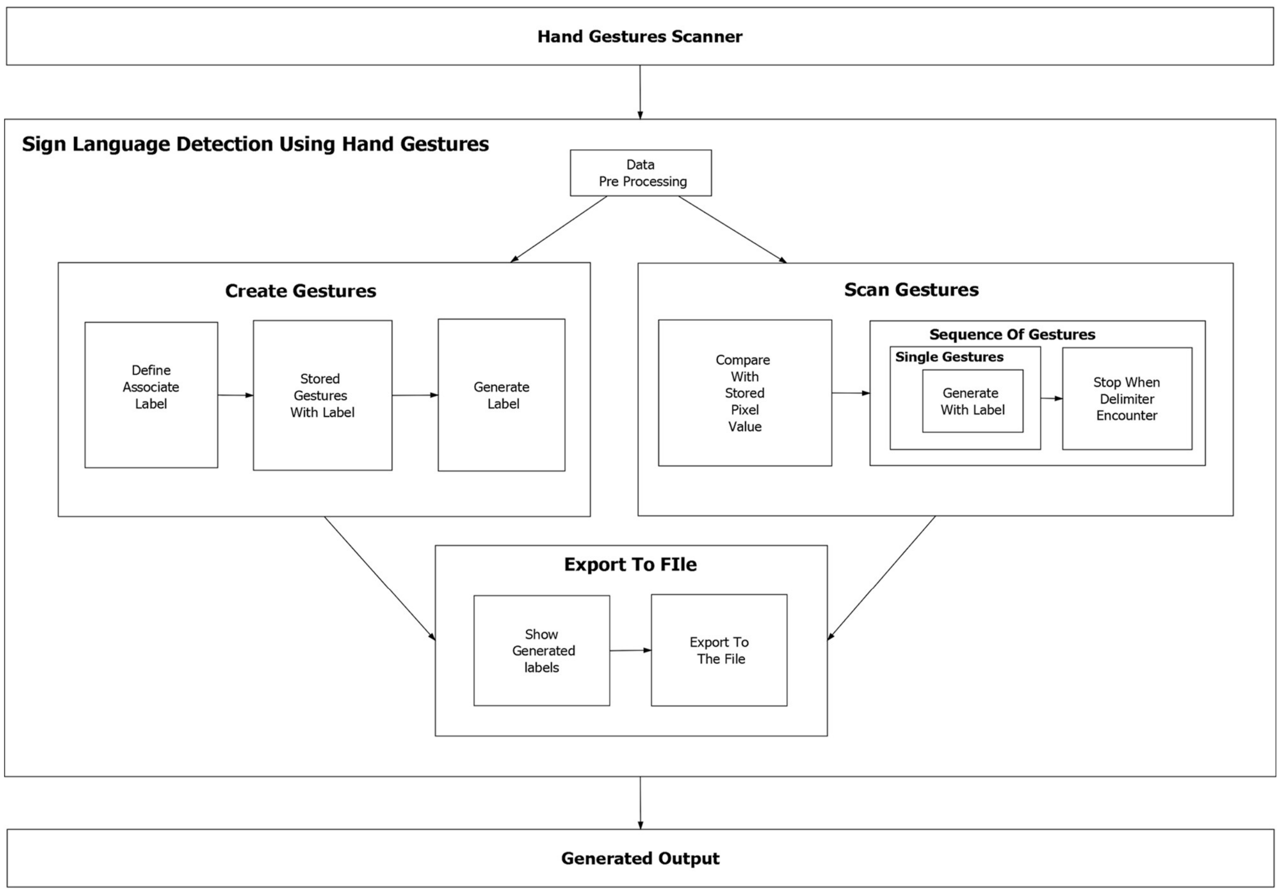


Figure 4.1: System Architecture

There are basically modules in the project namely:

1. **Create Gesture**:
   To create a new gesture for a new hand sign which is not present in the database.

2. **Scan Gesture**:
   To scan and predict the hand gesture done by the user by comparing it to the database.

3. **Export:**
   After the gesture has been identified, the corresponding alphabet/ word is stored in a ".txt" file.

Sign Linguistic Features can be mostly divided into three categories namely Manual Features, Non-Manual Features, and Finger Spelling.

- Manual Features:
  These features involve gestures done with hand employing hand shape and motion to interpret its meaning. These hand-shaped orientation poses or hand trajectories can be Tracking and Non-tracking based. Tracking of hands can be a crucial task as they may move faster and are often subject to motion blurring. Challenges are also faced when there is constant change in background and lightning which may fail in recognition and be time-consuming. But usage of proper methods may offer accurate results. Hand signs are recognized rather than classified and traced as it can be a significant issue. Combinations of images and binary skin-intensity images and derivatives are used. For this Sobel, filters may be applied. Following this, some geometric features are combined with appearance-based features in the form of movements. This creates a system with both tracing and non-tracking-based approaches.

- Fingerspelling

  Fingerspelling is an extension of manual features. Demonstration of a robust finger tracking system that uses stereo cameras for depth, edges, and color was done. Using many different approaches, the system detects and tracks fingers and merges them into a model. These approaches or channels are combined using Bayesian networks.

4. Non-Manual Features

   There is significant information in the non-manual channels as well. These facial expressions include head poses and lip shapes used for lip reading. Computation of optical flow on local face features was done, to determine which regions of the face move to create each expression. This relies on facial movements but can be perplexed by insufficient or combined expressions.

4.2 ASL DATABASE

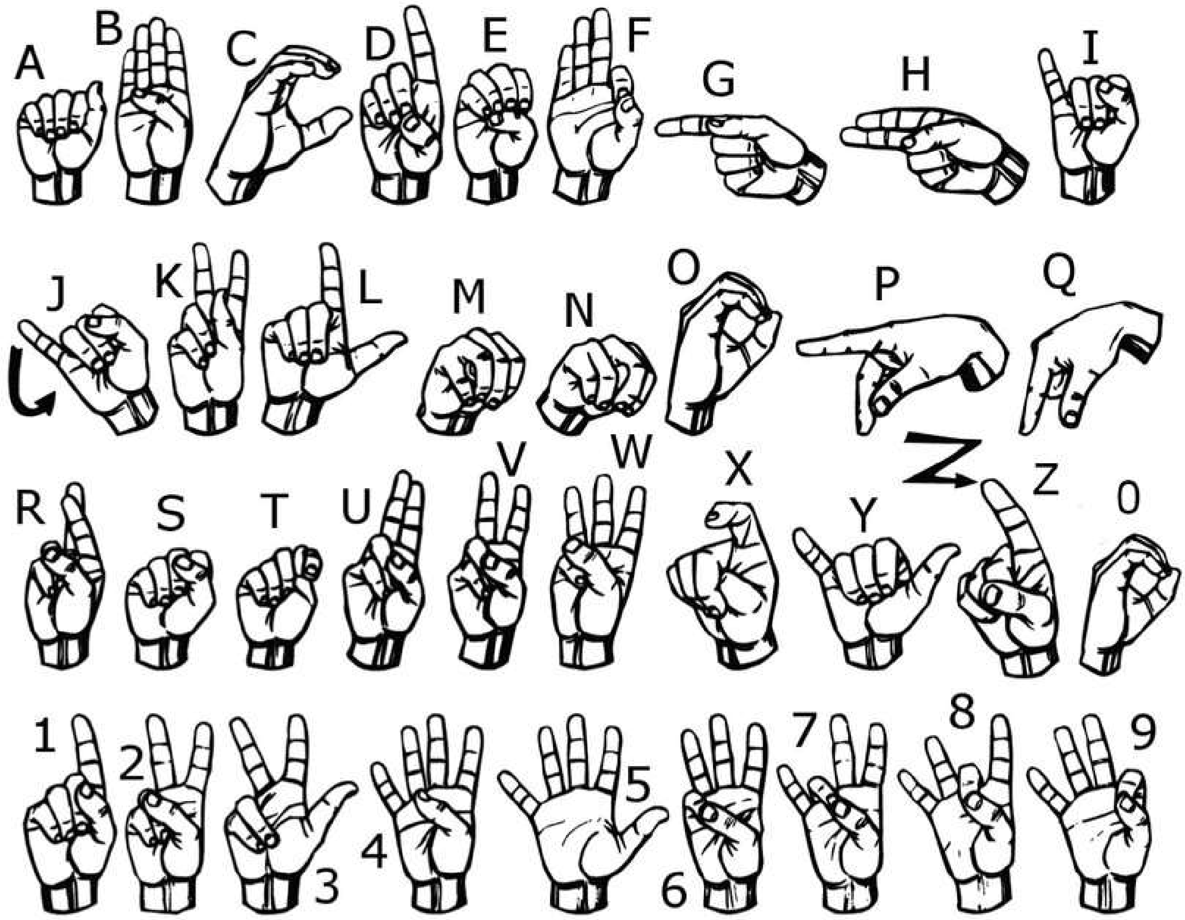


Figure 4.2: ASL Database

4.3 USE CASE DIAGRAM

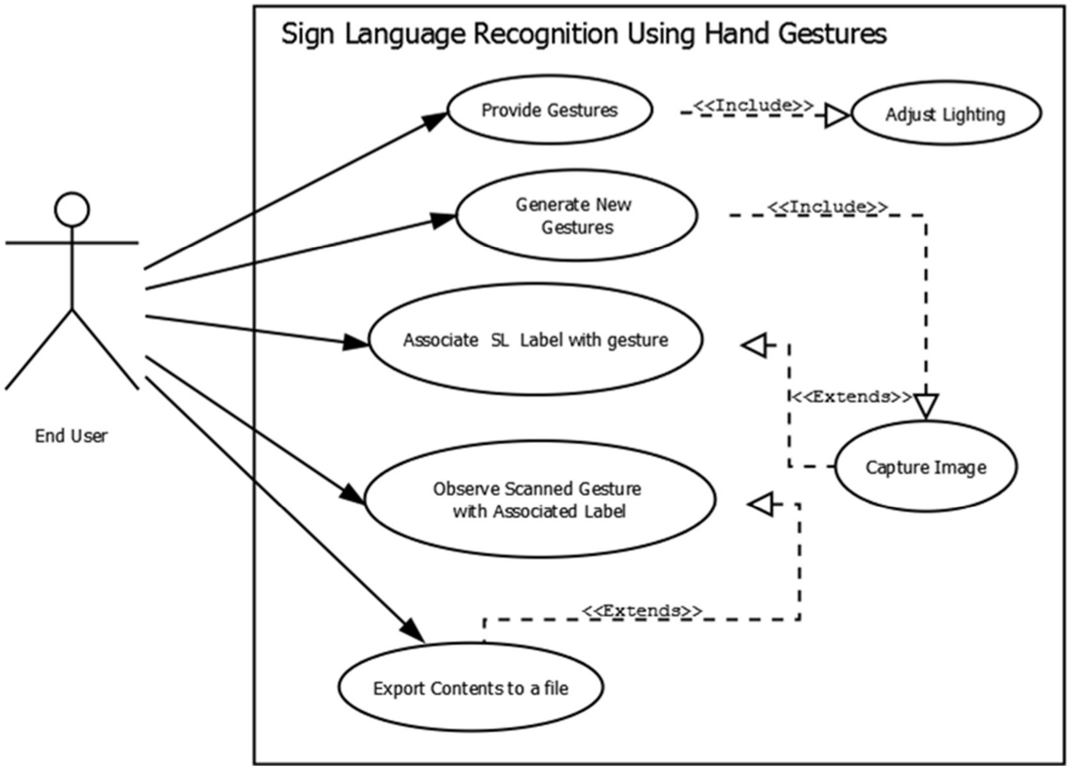


Figure 4.3: Use Case Diagram

4.4 ACTIVITY DIAGRAM

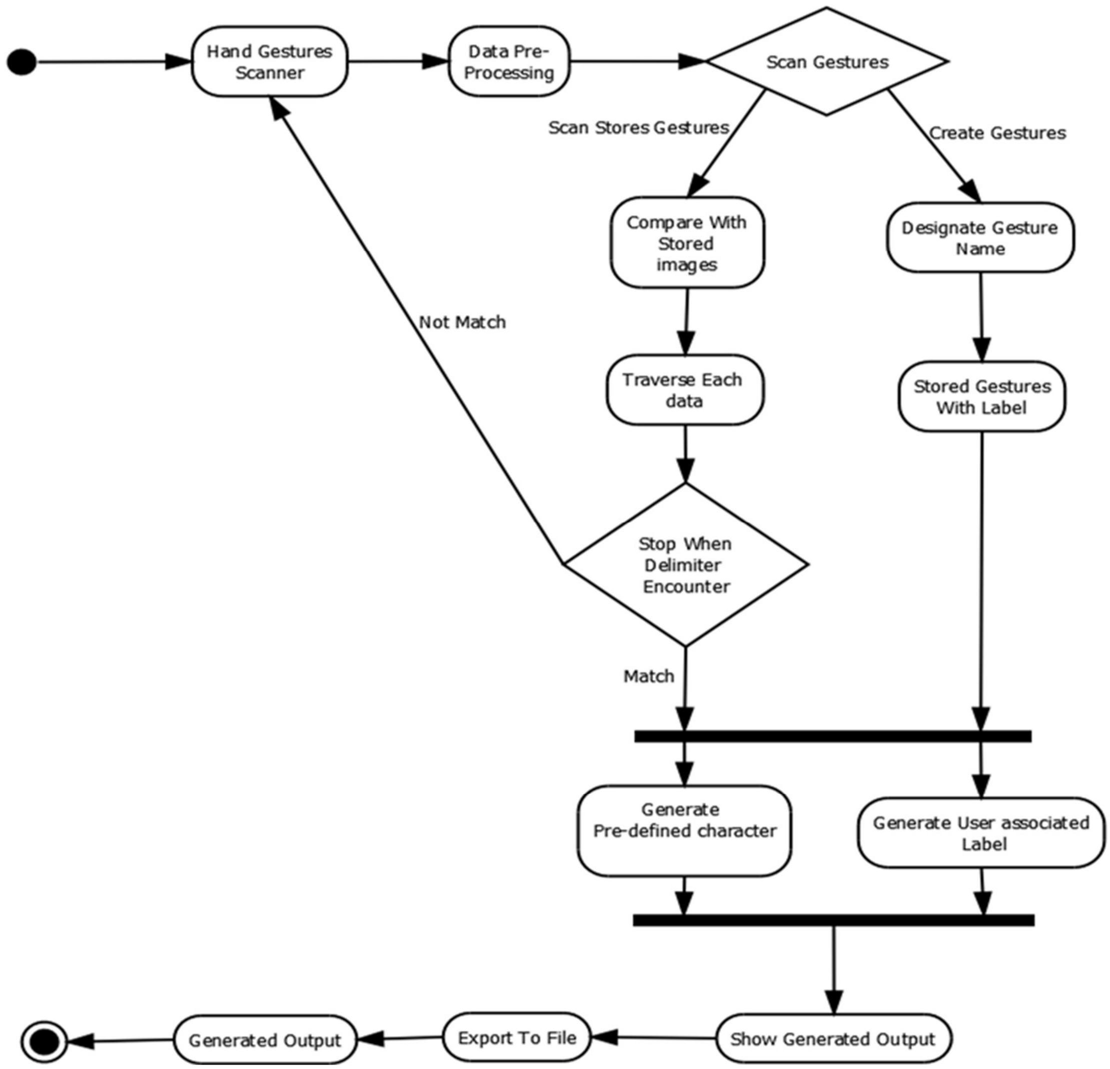


Figure 4.4: Activity Diagram

CHAPTER 5: PROJECT PLAN

5.1 PROJECT ESTIMATE

5.1.1 Reconciled Estimates

The project schedule is the tool that communicates what work needs to be performed, which resources of the organization will perform the work and the timeframes in which that work needs to be performed. The project schedule should reflect all of the work associated with delivering the project on time. Without a full and complete schedule, the project manager will be unable to communicate the complete effort, in terms of cost and resources, necessary to deliver the project. Online project management software allows project managers to track project schedules, resources, budgets and project related assets in real time. The project schedule can be viewed and updated by team members associated with the project, keeping everyone well informed on the overall project status.

5.1.2 Project Resources

Manually Created Dataset.

5.2 Risk Management

Managers can plan their strategy based on four steps of risk management which prevails in an organization. Following are the steps to manage risks effectively in an organization:

- Risk Identification
- Risk Quantification
- Risk Response
- Risk Monitoring and Control

5.2.1 Risk Identification

Managers face many difficulties when it comes to identifying and naming the risks that occur when undertaking projects. These risks could be resolved through structured or unstructured brainstorming or strategies. It's important to understand that risks pertaining to the project can only be handled by the project manager and other stakeholders of the project. Risks, such as operational or business risks will be handled by the relevant teams. The risks that often impact a project are supplier risk, resource risk and budget risk. Supplier risk would refer to risks that can occur in case the supplier is not meeting the timeline to supply the resources required. Resource risk occurs when the human resource used in the project is not enough or not skilled enough. Budget risk would refer to risks that can occur if the costs are more than what was budgeted.

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality.

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Description 1	Low	Low	High	High
2	Description 2	Low	Low	High	High

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 5.2: Risk Probability Definitions

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact Definitions

Risk ID	1
Risk Description	Enter User Current Symptoms
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Strategy
Risk Status	Occurred

Risk ID	2
Risk Description	Suggest Tablets, Diet Plan, Exercise and Doctor List
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Better testing will resolve this issue.
Risk Status	Identified

Risk ID	3
Risk Description	Training and Testing Module
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Very High
Response	Accept
Strategy	Example Running Service Registry behind proxy balancer
Risk Status	Identified

Table 5.4: Risk Analysis

5.3 PROJECT SCHEDULE:

Schedule		**Date**	**Project Activity**
August	2nd Week	15/08/2021	Project Topic Selection
	3rd Week	30/08/2020	Synopsis Submission
September	1st Week	02/09/2021	Presentation On Project Ideas
	2nd Week	03/09/2021	Presentation on Review 1 of Project
October	1st Week	04/10/2021	Presentation on Review 2 of Project
November	2nd Week	20/11/2021	Presentation on Review 3 of Project
December	3rd Week	21/12/2021	Semester I final presentation
	4th Week	26/12/2021	Discussion and implementation of 1st and 2nd module
January	1st Week	7/01/2022	Discussion on flow of project and designing new module
February	1st Week	06/02/2022	Modification of modules.
	2nd Week	13/02/2022	Designed test cases for our module.
	3rd Week	21/02/2022	Worked on user interface.
March	1st Week	06/03/2022	Integration of all modules.
	2nd Week	05/04/2021	Review 1 presentation of second semester.
April	2nd Week	18/04/2022	Review 2 presentation of second semester.
May	1st Week	07/05/2022	Review 3 presentation of second semester.

Table 5.5: Project Schedule

5.3.1 Project Task Set

T:1 Searching for project topic.

T:2 Submitting synopsis.

T:3 Enquiring about different aspects for topic selection.

T:4 Literature survey : Paper survey.

T:5 Literature survey.

T:6 Study of system architecture.

T:7 Study of UML - Use case diagram.

T:8 Study of UML - Activity Diagram.

T:9 Study of UML - Database diagram.

T:10 Study of Data Flow Diagram.

T:11 Studying for creating a first module.

T:12 Creating a first module with test cases.

T:13 Finalizing a first module with test cases.

T:14 Recovering error of database.

T:15 Presentation on phase I of project.

T:16 Studying for creating a second module.

T:17 Finalizing a second module with test cases.

T:18 Studying and preparing a GUI for second module.

T:19 Checking a solution for slow prediction issue

T:20 Removing of the problem.

T:21 Implement various options and buttons.

T:22 Adding create module.

5.3.2 Task Network

Task	Hours	Dependencies	Developer Assign
T1	36	-	D1,D2,D3,D4
T2	02	T1	D1,D2,D3,D4
T3	22	T1	D1,D2,D3,D4
T4	16	T3	D3,D4
T5	38	T4	D1,D2
T6	06	T3.T5	D2
T7	06	T4	D1,D2
T8	07	-	D1,D2
T9	21	-	D1,D2
T10	17	T4	D3,D4
T11	34	T4,T5	D1,D2
T12	04	T11	D2,D4
T13	06	T12	D1,D3

Table 5.6: Task Network

5.3.3 Timeline Chart

Activity	**I week**	**II week**	**III week**	**IV week**	**V week**	**VI week**	**VII week**	**VIII week**	**IX week**
Initiate the project									
Communication									
Literature survey									
Define scope									
Develop SRS									
Plan the project									
Design mathematical model									
Feasibility Analysis									
Develop work breakdown structure									
Planning project schedule									
Design UML and other diagrams									
Design test plan									
Design risk management plan									

Table 5.7: Timeline Chart

5.4 TEAM ORGANIZATION

Task	Hours	Dependencies	Developer Assigned
T1	36	-	D1,D2,D3,D4
T2	02	T1	D1,D2,D3,D4
T3	22	T1	D1,D2,D3,D4
T4	16	T3	D3,D4
T5	38	T4	D1,D2
T6	06	T3.T5	D2
T7	06	T4	D1,D2
T8	07	-	D1,D2
T9	21	-	D1,D2
T10	17	T4	D3,D4
T11	34	T4,T5	D1,D2
T12	04	T11	D2,D4
T13	06	T12	D1,D3

Table 5.8: Team Organization

5.4.2 Management Reporting and Communication

Project Communications Management includes the processes that are required to ensure timely and appropriate planning, collection, creation, distribution, storage, retrieval, management, control, monitoring, and the ultimate disposition of project information. Project managers spend most of their time communicating with team members and other project stakeholders, whether they are internal (at all organizational levels) or external to the organization. Effective communication creates a bridge between diverse stakeholders who may have different cultural and organizational backgrounds, different levels of expertise, and different perspectives and interests, which impact or have an influence upon the project execution or outcome.

The Project Communications Management processes are as follows:

Plan Communications Management:

The process of developing an appropriate approach and plan for project communications based on stakeholder's information needs and requirements, and available organizational assets.

Manage Communications:

The process of creating, collecting, distributing, storing, retrieving and the ultimate disposition of project information in accordance with the communications management plan.

Control Communications:

The process of monitoring and controlling communications throughout the entire project life cycle to ensure the information needs of the project stakeholders are met.

CHAPTER 6: PROJECT IMPLEMENTATION

6.1 OVERVIEW OF PROJECT MODULES

- Data Pre-Processing:

  In this module, based on the object detected in front of the camera its binary images is being populated. Meaning the object will be filled with solid white and background will be filled with solid black. Based on the pixel's regions, their numerical value in range of either 0 or 1 is being given to next process for module.

- Scan Single Gesture:

  A gesture scanner will be available in front of the end user where the user will have to do a hand gesture. Based on Pre-Processed module output, a user shall be able to see associated label assigned for each hand gestures, based on the predefined American Sign Language (ASL) standard inside the output window screen.

- Create gesture:

  A user will give a desired hand gesture as an input to the system with the text box available at the bottom of the screen where the user needs to type whatever he/she desires to associate that gesture with. This customize gesture will then be stored for future purposes and will be detected in the upcoming time.

- Formation of a sentence:

  A user will be able to select a delimiter and until that delimiter is encountered every scanned gesture character will be appended with the previous results forming a stream of meaning-full words and sentences.

- Exporting:

  A user would be able to export the results of the scanned character into an ASCII standard textual file format.

6.2 TOOLS AND TECHNOLOGIES USED

6.2.1 Hand Gesture Recognition:

Recognition of the hand gesture in kinds of literature based on extracted features is divided into three groups, as follows:

1. High-Level Features-Based Approaches:
   Aim to figure out the position of the palm and joint angles such as the fingertips, joint location, or anchor points of the palm. Whereas, effect collisions or occlusions on the image are difficult to detect after features are extracted , and sensitivity segmentation performance on 2D hand image are the problem that occurred frequently. The gestures are defined from the results with a set of rules and conditions from the vectors and joints of the hands.

2. Low-Level Feature-Based Approaches:
   Utilized these features for could be extracted quickly for robust to noise. Zhou discovered recognition of the hand shape as a cluster-based signature using a novel distance metric called Finger Earth's Distance. Stanner determines the bounding region of the hand elliptically for implement hand recognition based on principal axes. Yang did research using the optical flow of the hand region as a low-level feature. Low-Level Feature-Based is not efficient when cluttered background.

3. 3D Reconstruction-Based Approaches: Use the 3D model of features for achieving the construe of hand completely. Research showed that successfully segmenting the hand in skin color needs similarity and high contrast of the background related to the hand through structured light to bring in 3D of depth data. Another one uses a stereo camera to track numerous interest points of the superficies of the hand which results in difficulty for handle robust 3D reconstruction, despite data contains 3D has valuable information that can help dispose of vagueness. See for more 3D reconstruction-based approach.

From kinds of literature, there are three Hand gesture recognition methods, as follows:

1. Machine Learning Approaches:
   The resulting output came from the stochastic process and approach based on statistical modelling for dynamic gestures such as PCA, HMM, advanced particle filtering, and condensation algorithm .

2. Algorithm Approaches:
   Collection of encoded conditions and restraints manually for defining as gestures in dynamic gestures. Galveia applied a 3rd-degree polynomial equation to determine the dynamic component of the hand gestures (create a 3rd-degree polynomial equation, recognition, reduced complexity of equations, and comparison handling in gestures library).

3. Rule-based Approaches:
   Suitable for dynamic gestures either static gestures which are contained a set of pre-encoded rules and features inputs. The features of input gestures are extracted and compared to the encoding rules that are the flow of the recognized gestures. Matching between gestures with rule and input which is outputted approved as known gestures.

6.2.2 MediaPipe Framework:

Today, there are many frameworks or libraries of machine learning for hand gesture recognition. One of them is MediaPipe. The MediaPipe is a framework designed to implement production-ready machine learning that must build pipelines to perform inference over arbitrary sensory data, has published code accompanying research work, and build technology prototypes. In MediaPipe, graph modular components come from a perception pipeline along with the function of inference model function, media processing model, and data transformations. Graph of operations are used in others machine learning such as Tensor flow, MXNet, PyTorch, CNTK, OpenCV 4.0.

Using MediaPipe for hand gesture recognition has been researched by Zhang before, using a single RGB camera for AR/VR application in a real-time system that predicts a hand skeleton of the human. We can develop a combined MediaPipe using other devices. The MediaPipe implements pipeline in Figure consists of two models for hand gesture recognition as follows:

1. A palm detector model processes the captured image and turns the image with an oriented bounding box of the hand,
2. A hand landmark model processes on cropped bounding box image and returns 3D hand key points on hand.
3. A gesture recognizer that classifies 3D hand key points then configuration them into a discrete set of gestures.

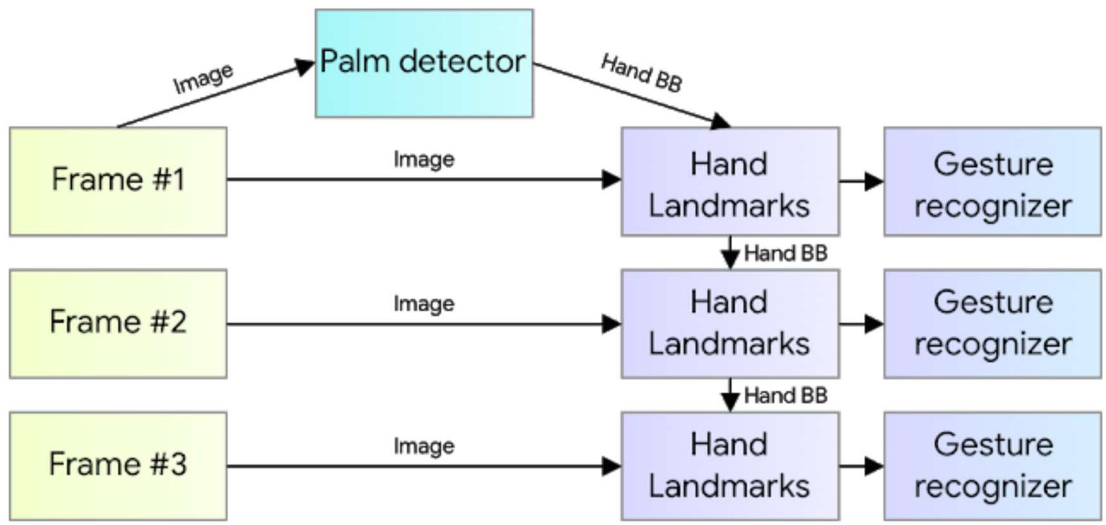


Figure 6.1: Hand Perception Pipeline Overview

MediaPipe framework has built detect initial palm detector called BlazePalm. Detecting the hand is a complex task. Step one is to train the palm instead of the hand detector, then using the non-maximum suppression algorithm on the palm, where it is modeled using square bounding boxes to avoid other aspect ratios and reducing the number of anchors by a factor of 3-5. Next, encoder-decoder of feature extraction that is used for bigger scene context-awareness even small objects, lastly, minimize the focal loss during training with support a large number of anchors resulting from the high scale variance.

6.2.3 Pre-Processing of Images to get Multi-hand Landmarks using MediaPipe:

MediaPipe is a framework that enables developers for building multi-modal (video, audio, any times series data) cross-platform applied ML pipelines. MediaPipe has a large collection of human body detection and tracking models which are trained on a massive and most diverse dataset of Google. As the skeleton of nodes and edges or landmarks, they track key points on different parts of the body. All co-ordinate points are three-dimension normalized. Models build by Google developers using Tensorflow lite facilitates the flow of information easily adaptable and modifiable via graphs.

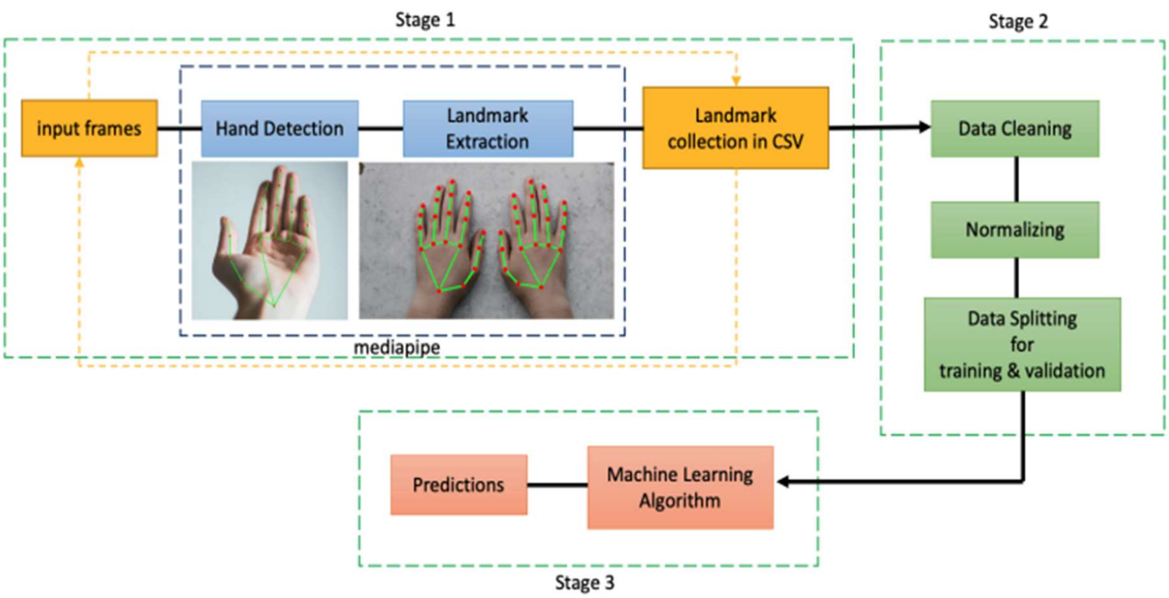


Figure 6.2: Proposed method to detect hand gestures and predict sign language

MediaPipe pipelines are composed of nodes on a graph which are generally specified in pbtxt file. These nodes are connected to C++ files. Expansion upon these files is the base calculator class in Mediapipe. Just like a video stream this class gets contracts of media streams from other nodes in the graph and ensures that it is connected. Once, rest of the pipelines nodes are connected, the class generates its own output processed data. Packet objects encapsulating many different types of information are used to send each stream of information to each calculator. Into a graph, side packets can also be imposed, where a calculator node can be introduced with auxiliary data like constants or static properties.

This simplified structure in the pipeline of dataflow enables additions or modifications with ease and the flow of data becomes more precisely controllable. The Hand tracking solution has an ML pipeline at its backend consisting of two models working dependently with each other: a) Palm Detection Model b) Land Landmark Model. The Palm Detection Model provides an accurately cropped palm image and further is passed on to the landmark model. This process diminishes the use of data augmentation (i.e. Rotations, Flipping, Scaling) that is done in Deep Learning models and dedicates most of its power for landmark localization.

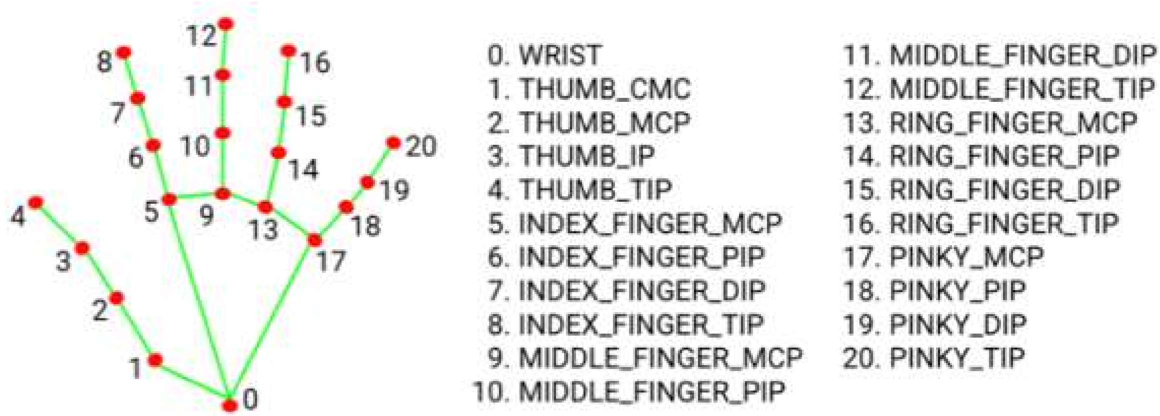


Figure 6.3: Hand Landmarks

The traditional way is to detect the hand from the frame and then do landmark localization over the current frame. But in this Palm Detector using ML pipeline challenges with a different strategy. Detecting hands is a complex procedure as you have to perform image processing and thresholding and work with a variety of hand sizes which leads to consumption of time. Instead of directly detecting hand from the current frame, first, the

Palm detector is trained which estimates bounding boxes around the rigid objects like palm and fists which is simpler than detecting hands with coupled fingers. Secondly, an encoder-decoder is used as an extractor for bigger scene context.

After the palm detection is skimmed over the whole image frame, subsequent Hand Landmark models comes into the picture. This model precisely localize 21 3D hand-knuckle coordinates (i.e., x, y, z-axis) inside the detected hand regions. The model is so well trained and robust in hand detection that it even maps coordinates to partially visible hand. Figure shows the 21 landmark points detection by the Hand Landmark model. Now that we have a functional Palm and Hand detection model running, this model is passed over our dataset of various language. Considering the American Sign Language dataset, we have a to z alphabets. So, we pass our detection model over every alphabet folder containing images and perform Hand detection which yields us the 21 landmark points as shown in Figure. The obtained landmark points are then stored in a file of CSV format. A simultaneous, elimination task is performed while extracting the landmark points. Here, only the x and y coordinates detected by the Hand Landmark model is considered for training the ML model. Depending upon the size of the dataset around 10-15 minutes is required for Landmark extraction.

6.2.4 Data Cleaning and Normalization:

As mentioned, we are only considering x and y coordinates from the detector, each image in the dataset is passed through to collect all the data points under one file. This file is then scraped through the pandas' library function to check for any nulls entries. Sometimes due to blurry image, the detector cannot detect the hand which leads to null entry into the dataset. Hence, it is necessary to clean these points or will lead to biasness while making the predictive model. Rows containing these null entries are searched and using their indexes removed from the table. After the removal of unwanted points, we normalized x and y coordinates to fit into our system. The data file is then prepared for splitting into training and validation set. 80% of the data is retained for training our model with various optimization and loss function, whereas 20% of data is reserved for validating the model.

Predictive analysis of different sign languages are performed using machine learning algorithms and Support Vector Machine (SVM) outperformed other algorithms. The details of the analysis are discussed in table 2 in the result section. SVM is effective in high dimensional spaces. In the case where the number of samples are greater than the number of dimensions, SVM performs effectively. SVM is a cluster of supervised learning methods capable of classification, regression and outliers detection.

6.3 ALGORITHM DETAILS

6.3.1 CNN:

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics. A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

6.3.1 HMM:

The HMM has rightfully enjoyed popularity since the '90s for hand sign recognition. HMMs are particularly suited for the problem of Sign Language Recognition. In the later years Parallel Hidden Markov Models (PaHMMs) were proven to be superior to HMM.ge due to the intrinsic parallel nature of the phonemes. The major drawback though is that regular HMMs are simply not scalable in terms of handling the parallel nature of phonemes present in sign. PaHMMs can be used as a better alternative for this problem by utilizing modeling parallel processes independently and combining output probabilities afterward. Its main advantage is it is easily extended to deal with strong TC tasks. Embedded re-estimation is possible, easy to understand and the disadvantage being it makes large assumptions about the data. A huge number of parameters needs to be set. Training data required is large.

6.3.2 ANN:

From the table, it is clear that we can use deep ANN to get better accuracy than regular ANN as we get a 5% better accuracy rate and 15% rate in testing speed.

6.3.3 K means:

The strategy used is to determine k points called centers as the sum of the distances of all data points to their respective cluster centers. Its main advantage is that it is computationally faster and produces tighter clusters. But it lacks in some areas as a prediction of K is difficult for a fixed number of clusters. Different initial partitions resulting in different final clusters seem to be an issue too.

6.3.4 K-nearest neighbours:

It uses the closest training of the feature space using instance-based learning and takes in the class of the nearest neighbour. It is easy to implement, lowest complexity and carefully chosen features give good results but are sensitive to arbitrary attributes. Hence it has the lowest accuracy rate of 78%.

6.3.5 Support Vector Machine:

It non-linearly maps the input data dimensional space, having linearly separated data for classification. It has a higher prediction accuracy of about 96% and is also robust for errors in training examples. It also has a fast evaluation of the learned target function and has a recognition time of 0.017

CHAPTER 7: SOFTWARE TESTING

7.1 Type of Testing

Manual Testing

7.2 TEST CASES AND TEST RESULTS

7.2.1 Test Case: Predict Hand Sign Gesture Y

Test Methodology:

The camera captures the images from the video stream. Hand Detection and Landmark Extraction is performed with the help of mediapipe. Prediction is done based on the trained model and the sign is predicted.

Test Results:

The Sign made by the user is Y

7.2.2 Test Case: Predict Hand Sign Gesture H

Test Methodology:

The camera captures the images from the video stream. Hand Detection and Landmark Extraction is performed with the help of mediapipe. Prediction is done based on the trained model and the sign is predicted.

Test Results:

The Sign made by the user is H

7.2.3 Test Case: Predict Hand Sign Gesture and form Words and Sentences

Test Methodology:

The camera captures the images from the video stream. Hand Detection and Landmark Extraction is performed with the help of mediapipe. Prediction is done based on the trained model and the sign is predicted. As the Add button is pressed the letter predicted is concatenated to form word and in turn a sentence.

Test Results:

A sentence is formed

7.2.4 Test Case: Create a new Hand Gesture

Test Methodology:

The camera captures the images from the video stream. Hand Detection and Landmark Extraction is performed with the help of mediapipe. These landmarks are recorded in the database and prediction model can be created.

Test Results:

A new Hand Gesture is created.

7.2.5 Test Case: Export to txt File

Test Methodology:

The predicted text is generated and stored. This can be written in a file from python and this file is then exported as a text file.

Test Results:

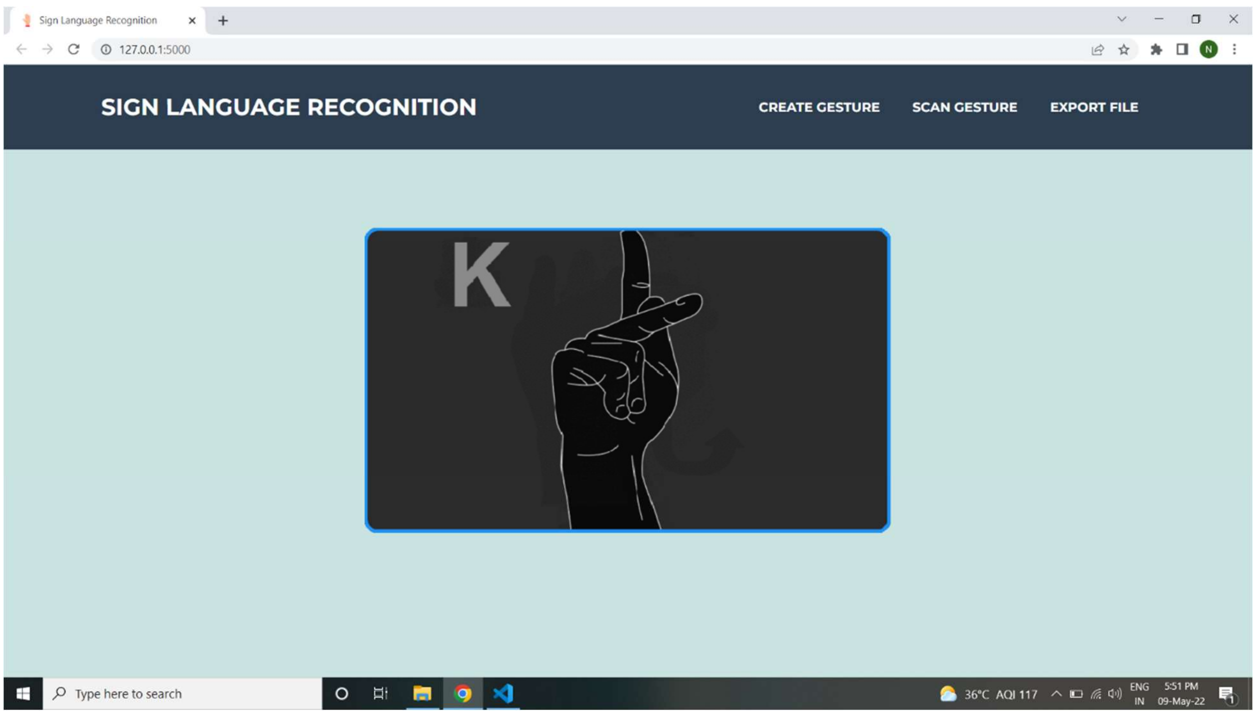
The text file with predicted text is generated.

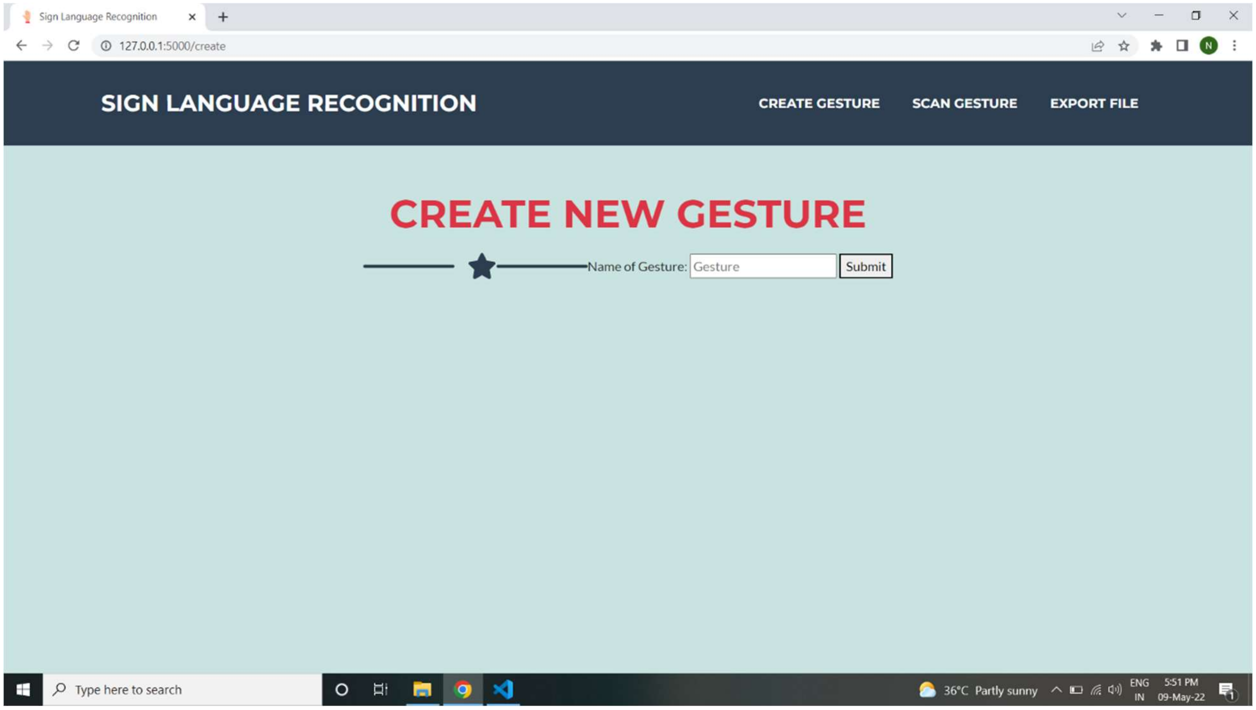
CHAPTER 8: RESULTS

8.1 OUTCOMES

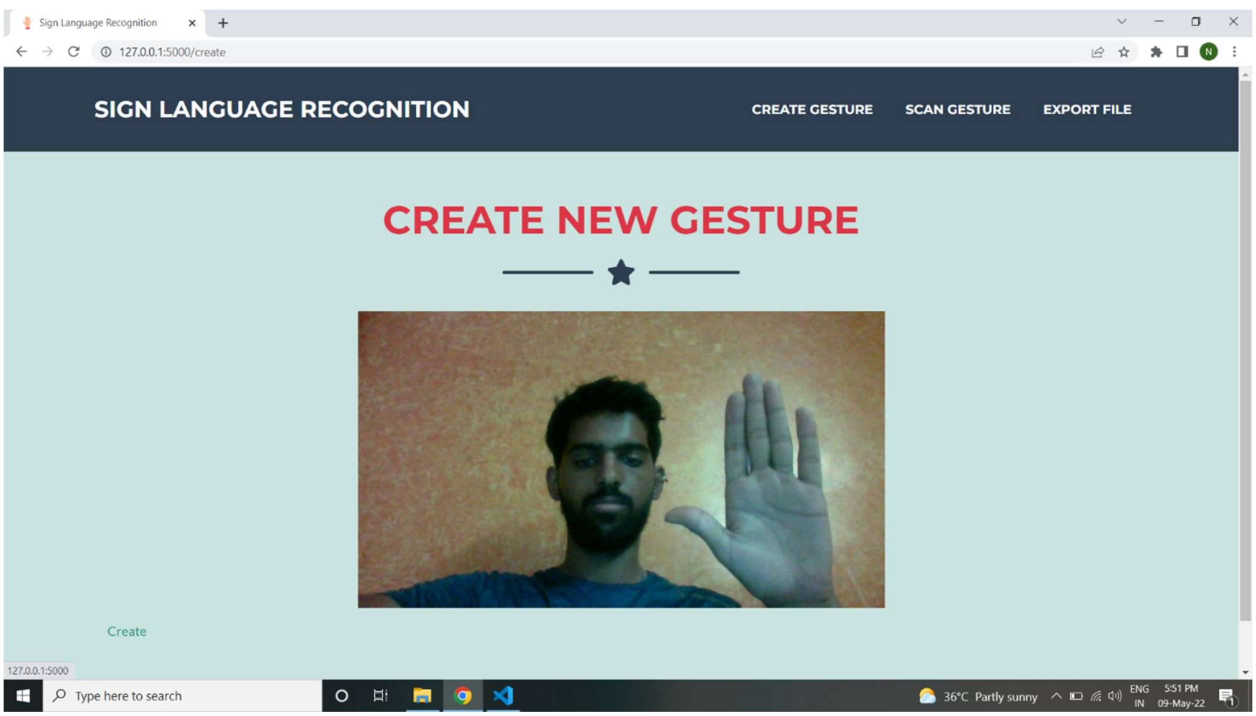
The system was able to predict the hand gestures correctly. The model which we had trained gave us accurate results and thus the project was implemented successfully.

8.2 SCREENSHOTS:

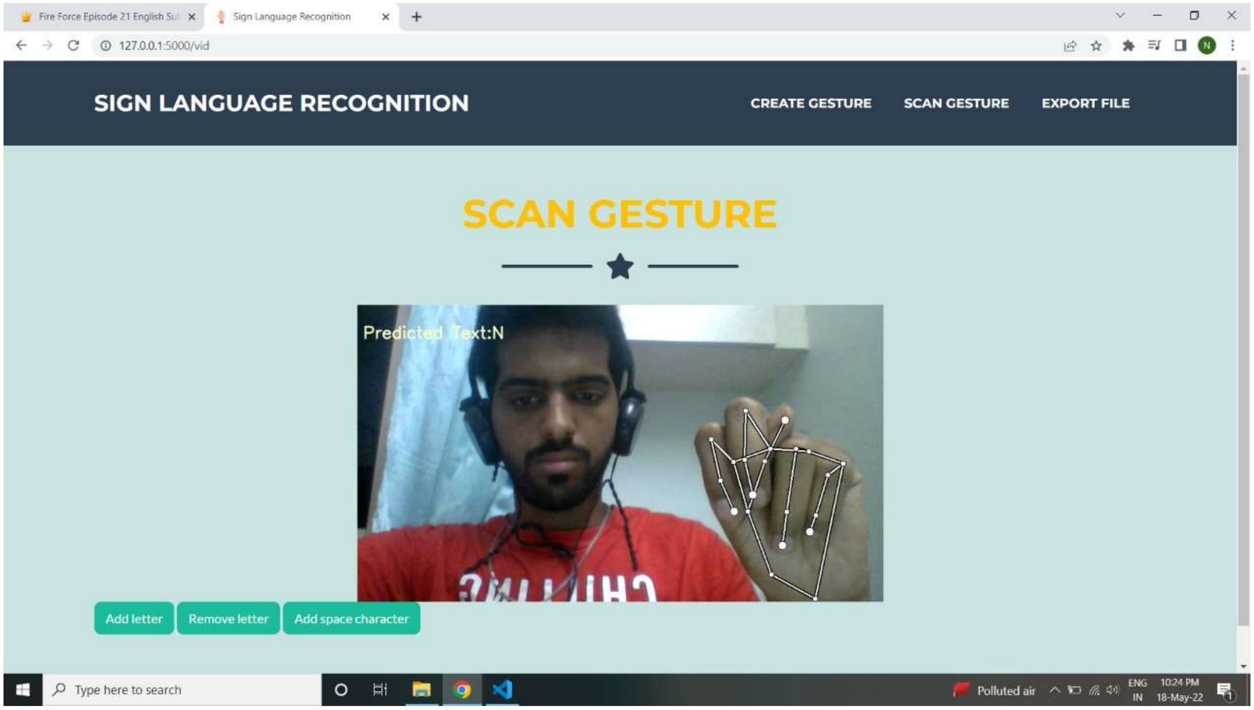


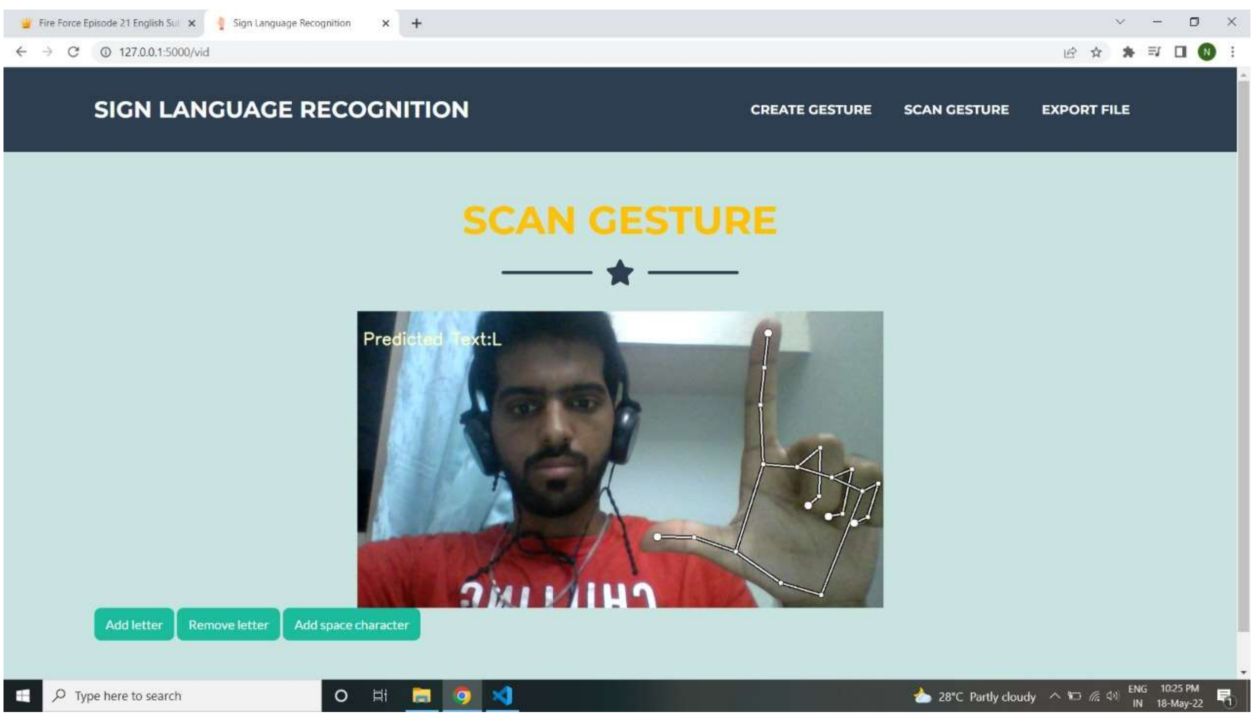


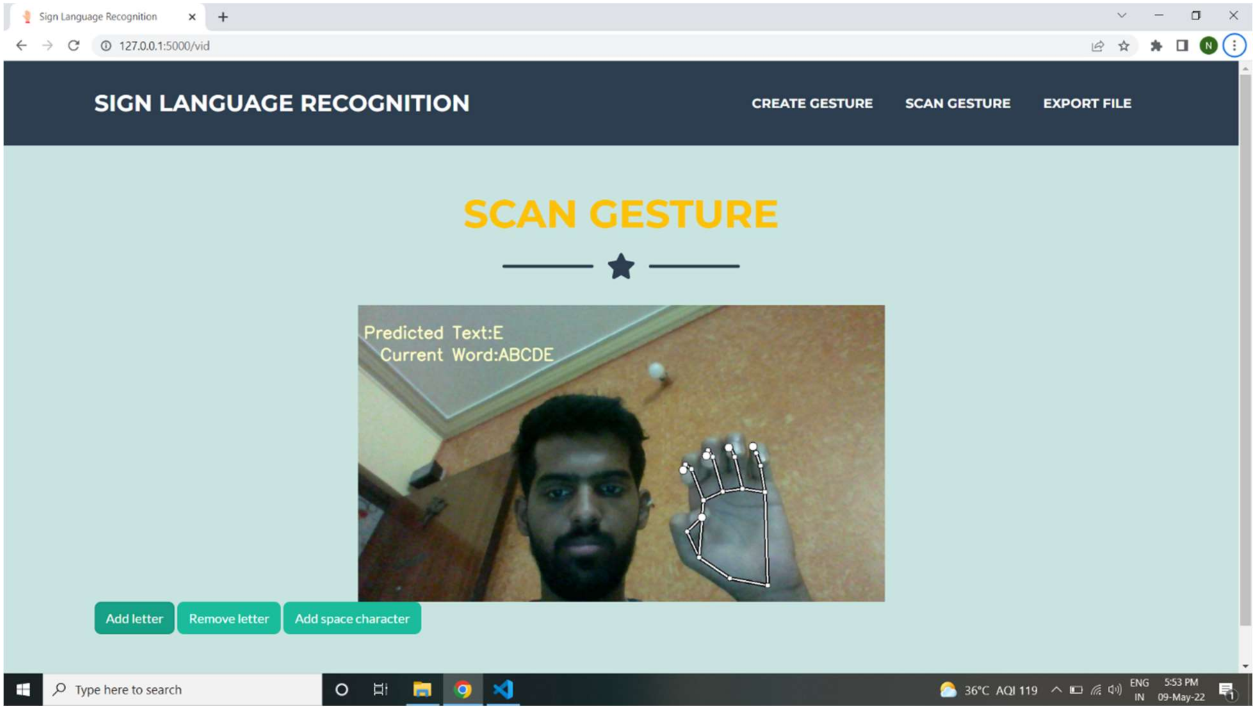
Sign Language Recognition
127.0.0.1:5000/create
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
CREATE NEW GESTURE
Name of Gesture:
Gesture
Submit
Type here to search
36°C Partly sunny

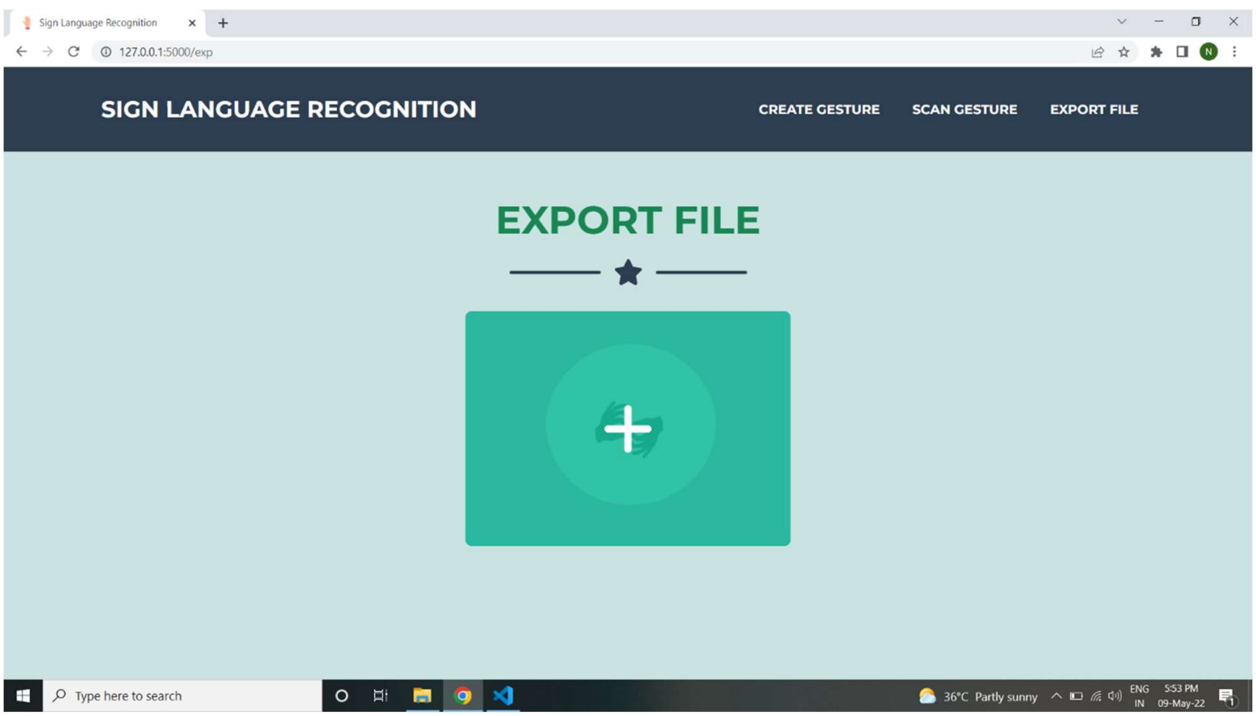


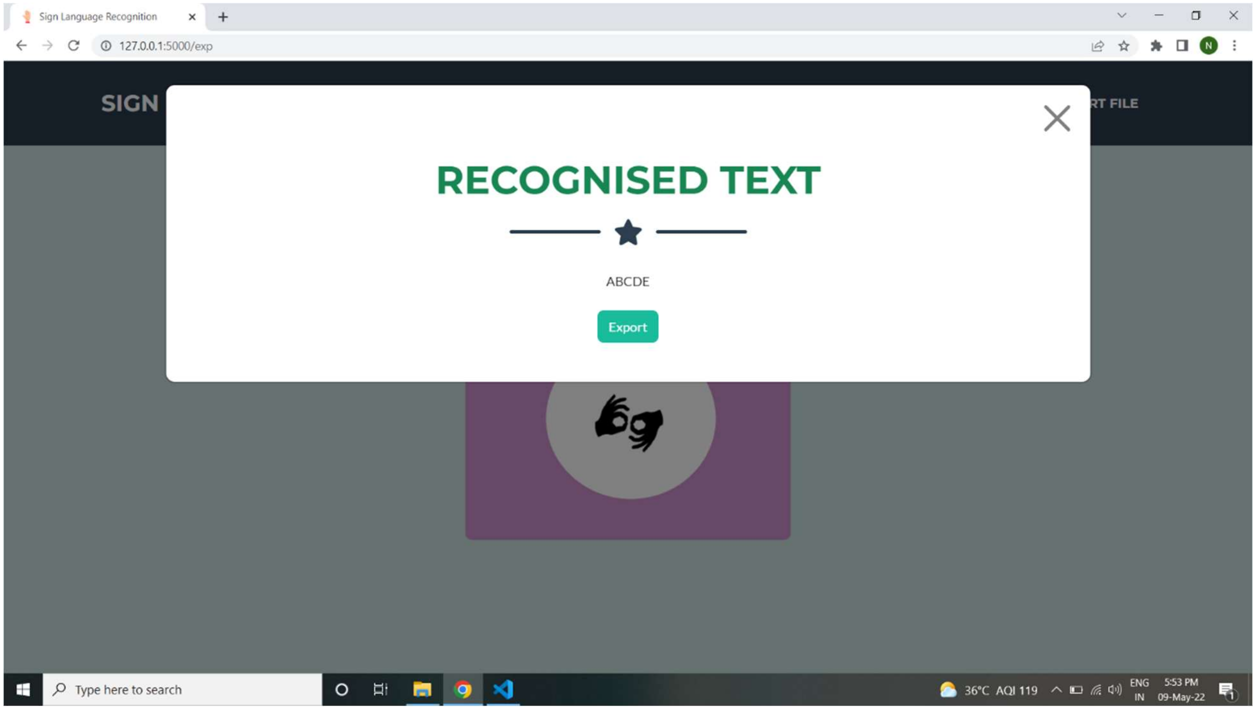
Sign Language Recognition
127.0.0.1:5000/create
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
CREATE NEW GESTURE
Create
Type here to search
36°C Partly sunny


127.0.0.1:5000/vid
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
SCAN GESTURE
Predicted Text:N
Add letter
Remove letter
Add space character


127.0.0.1:5000/vid
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
SCAN GESTURE
Predicted Text:L
Add letter
Remove letter
Add space character


Sign Language Recognition
127.0.0.1:5000/vid
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
SCAN GESTURE
Predicted Text:E
Current Word:ABCDE
Add letter
Remove letter
Add space character
Type here to search
36°C AQI 119
ENG IN
5:53 PM
09-May-22


Sign Language Recognition
127.0.0.1:5000/exp
SIGN LANGUAGE RECOGNITION
CREATE GESTURE
SCAN GESTURE
EXPORT FILE
EXPORT FILE
Type here to search
36°C Partly sunny
ENG IN
5:53 PM
09-May-22


Sign Language Recognition
127.0.0.1:5000/exp
SIGN
RT FILE
RECOGNISED TEXT
ABCDE
Export
Type here to search
36°C AQI 119
ENG
IN
5:53 PM
09-May-22

CHAPTER 9: CONCLUSIONS

9.1 Conclusion

We have studied multiple techniques to recognize hand gestures and sign languages. It is presented on the basis of fragmentation, feature extraction, feature matching used in HGRS.A comprehensive study of performances different techniques have been made in this paper. These observations are noted in order to be used in further studies to try and improve the efficiency and accuracy of hand sign recognition which can further help the hearing-impaired people.

9.2 Future Work

The future work for this project would be to

- Addition of gestures to form words or sentences
- Hosting the project
- Creating an mobile application of the same

9.3 Applications

- To improve the communication gap between hearing impaired and the others
- Could remove the dependency of translator for understanding hand signs
- Can also be used to remove dependency on different forms of hand sign language

APPENDIX

DETAILS OF PAPER PUBLICATION:

Name of the Journal:

International Journal of Scientific Research in Computer Science, Engineering and Information Technology

Topic:

Sign Language Recognition

Link:

https://ijsrcseit.com/CSEIT228234

Published in :

Volume 8 | Issue 2 | pp. 200-204 | March-April 2022

DOI:

https://doi.org/10.32628/CSEIT228234

ISSN :

2456-3307

Certificate:

IJSRCSEIT

Scientific Journal Impact Factor = 6.135
Online ISSN : 2456-3307
UGC Journal No : 64718

International Journal of Scientific Research in Computer Science, Engineering and Information Technology

CERTIFICATE OF PUBLICATION

Ref : IJSRCSEIT/Certificate/Volume 8/Issue 2/7848 09-Apr-2022

This is to certify that **Nikhil Kulkarni, Shivali Mate, Atharva Kulkarni, Shailaja Jadhav** have published a research paper entitled *'Sign Language Recognition'* in the International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), Volume 8, Issue 2, March-April 2022.

This Paper can be downloaded from the following IJSRCSEIT website link
https://ijsrcseit.com/CSEIT228234
DOI : https://doi.org/10.32628/CSEIT228234
IJSRCSEIT Team wishes all the best for bright future

Editor in Chief
IJSRCSEIT
website : http://ijsrcseit.com

Peer Reviewed and Refereed International Journal

Paper:

International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT228234

Sign Language Recognition

Nikhil Kulkarni*1, Shivali Mate*1, Atharva Kulkarni*1, Shailaja Jadhav2
*1 Computer Engineer, Marathwada Mitra Mandal's College of Engineering, Pune, Maharashtra, India
2 Assistant Professor, Marathwada Mitra Mandal's College of Engineering, Pune, Maharashtra, India

Article Info
Volume 8, Issue 2
Page Number : 200-204

Publication Issue :
March-April-2022

Article History
Accepted: 01 April 2022
Published: 09 April 2022

ABSTRACT

There seems to be an issue between the communication of differently abled and others. That is why we have decided to make an application to mitigate this issue and reduce the dependencies on third parties like translators. We have studied different approaches taken and discussed them. Recognition of hand signs is a popular topic since the 90s and various findings have been uncovered in the following years. We have written about popular methods for this. For this application to work we have to weigh in feature extraction, object detection, and predictions using a machine learning model.

Keywords : Feature detection, Image Processing, CNN, HMM, K means, SVM.

I. INTRODUCTION

Hearing loss can be of any level and have mild to severe effects. This hearing loss can be attributed to anything including age, occupation, and exposure to extreme noise. In addition, hearing loss does not just make things quieter, it can distort sounds and cause normal speech to sound jumbled and garbled.

As we know that there are various languages used for speaking, there also exist different languages for hand signs like American Sign Language (ASL), Indian Sign Language (ISL), and Devanagri Sign Language (DSL). As the majority of people do not understand sign language, deaf people suffer many problems communicating with the masses. They heavily rely on translators to interpret the sign language and act as a bridge between them. This not only halts the progress of the physically disabled but also leads to heavy dependency on translators which is not optimal. As translators charge a hefty amount and are not always available at any hour, this option is heavily flawed and needs another dependable alternative.

Hence, we propose to develop an app that can detect the hand signs of a person and then convert them into sentences that we can easily read/listen to. This will not only dispose of any dependency on translators but will also cut the cost of translators. We plan to make this

application available on every smartphone so that no person feels disconnected from the rest of the world. Hand sign recognition includes phases like image capturing, pre-processing, extraction of the region,

:

Plagiarism Report:

Dupli Checker

PLAGIARISM SCAN REPORT

2% Plagiarised	98% Unique		Date	2021-12-22
			Words	939
			Characters	5821

Content Checked For Plagiarism

Here we are going to discuss various machine learning models. Firstly we are going to talk about CNN.

1. HMM:

The HMM has rightfully enjoyed popularity since the '90s for hand sign recognition. HMMs are particularly suited for the problem of Sign Language Recognition. In the later years Parallel Hidden Markov Models (PaHMMs) were proven to be superior to HMM.ge due to the intrinsic parallel nature of the phonemes. The major drawback though is that regular HMMs are simply not scalable in terms of handling the parallel nature of phonemes present in sign. PaHMMs can be used as a better alternative for this problem by utilizing modeling parallel processes independently and combining output probabilities afterward. Its main advantage is it is easily extended to deal with strong TC tasks. Embedded re-estimation is possible, easy to understand and the disadvantage being it makes large assumptions about the data.

A huge number of parameters needs to be set. Training data required is large

2. CNN

CNN takes an architectural approach as given below:

According to the proposed CNN model, we can get an accuracy of around 90% which is great as compared to other techniques. Although it takes greater time to train the CNN model it predicts competitively less time than others.

3. ANN

From the table, it is clear that we can use deep ANN to get better accuracy than regular ANN as we get a 5% better accuracy rate and 15% rate in testing speed.

4. K means

The strategy used is to determine k points called centers as the sum of the distances of all data points to their respective cluster centers. Its main advantage is that it is computationally faster and produces tighter clusters. But it lacks in some areas as a prediction of K is difficult for a fixed number of clusters. Different initial partitions resulting in different final clusters seem to be an issue too.

5. K-nearest neighbors

It uses the closest training of the feature space using instance-based learning and takes in the class of the nearest neighbor. It is easy to implement, lowest complexity and carefully chosen features give good results but are sensitive to arbitrary attributes. Hence it has the lowest accuracy rate of 78%.

6. Support vector machine

It non-linearly maps the input data dimensional space, having linearly separated data for classification. It has a higher prediction accuracy of about 96% and is also robust for errors in training examples. It also has a fast evaluation of the learned target function and has a recognition time of 0.017

B. Image Processing

Sign Linguistic Features can be mostly divided into three categories namely Manual Features, Non-Manual Features, and Finger Spelling [3].

1. Manual Features:

These features involve gestures done with hand employing hand shape and motion to interpret its meaning. These hand-shaped orientation poses or hand trajectories can be Tracking and Non-tracking based [3]. Tracking of hands can be a crucial task as they may move faster and are often subject to motion blurring. Challenges are also faced when there is constant change in background and lightning which may fail in recognition and be time-consuming. But usage of proper methods may

REFERENCES:

[1] Arpita Haldera , Akshit Tayade, "Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning", International Journal of Research Publication and Reviews, s Vol (2) Issue (5) (2021) Page 9-17, ISSN 2582-7421

[2] Ketan Gomase, Akshata Dhanawade, Prasad Gurav, Sandesh Lokare, "Sign Language Recognition using Mediapipe", International Research Journal of Engineering and Technology (IRJET), Volume: 09 Issue: 01 | Jan 2022, p-ISSN: 2395-0072, e-ISSN: 2395-0056

[3] Mayuri Murkute, Jayshree R. Pansare "Comprehensive Survey of Performance of Techniques for Hand Gesture Recognition System for Sign Languages", International Journal of Scientific Research in Computer Science, Engineering and Information Technology 2017, IJSRCSEIT, Volume 2, Issue 6, ISSN: 2456-3307

[4] Elif Baykal1 ·Hulya Dogan1 · Mustafa Emre Ercin2 · Safak Ersoz2 · Murat Ekinci1 "Transfer learning with pre-trained deep convolutional neural networks for serous cell classification", Springer Science+Business Media, LLC, part of Springer Nature 2019

[5] Helen Cooper, Brian Holt, and Richard Bowden, "Sign Language Recognition"

[6] Real-time Anish Gurav, Premanand K. Kadbe, "Real-time Finger Tracking and Contour Detection for Gesture Recognition using OpenCV", 2015 International Conference on Industrial Instrumentation and Control (ICIC) College of Engineering Pune, India. May 28-30, 2015

[7] Ashish S. Nikam, Aarti G. Ambekar, "Sign Language Recognition Using Image-Based Hand Gesture Recognition", Techniques2016 Online International Conference on Green Engineering and Technologies (IC-GET)

[8] Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, and Benjamin Schrauwen "Sign Language Recognition Using Convolutional Neural Networks", ELIS, Ghent University, Ghent, Belgium

[9] Thad Starner, Alex Pentland, "Real-time American sign language recognition from video using hidden Markov models", M. ShIlh and R. Jain (etis.), Motion-Based Recognition, 227-243, 1997 Kluwer Academic Publishers

[10] Shivashankara Ss, Dr.Srinath S, "American Sign Language Recognition System: An Optimal Approach", I.J. Image, Graphics, and Signal Processing, 2018, 8, 18-30

[11] Mehreen Hurroo, Mohammad Elham Walizad, "Sign Language Recognition System using Convolutional Neural Network and Computer Vision", International Journal of

Engineering Research & Technology (IJERT) ISSN: 2278-0181 IJERTV9IS120029 Vol. 9 Issue 12, December-2020

[12] ER. Kanika Goyal, Amitoj Singh, "InDifferently-abled Recognition System for Differently-abled People", Journal on Today's Ideas Tomorrow's Technologies (2): 145-151, DOI: 10.15415/jotitt.2014.22011.

[13] Sangeetha R, Elakkiya S, "An Innovative Artificial Replacement to Facilitate Communication between Visually and Hearing- Impaired People", International Journal of Scientific Research in Computer Science, Engineering and Information Technology 2017, IJSRCSEIT, Volume 2, Issue 5, ISSN: 2456-3307

[14] Zhao, Ming & Bu, Jiajun & Chen, C. (2002). "Robust background subtraction in HSV color space", Proceedings of SPIE MSAV, vol. 1. 4861. 10.1117/12.456333.

[15] K. Bantupalli and Y. Xie, "American Sign Language Recognition using Deep Learning and Computer Vision," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 4896-4899, DOI: 10.1109/BigData.2018.8622141

[16] Huang, J., Zhou, W., & Li, H. (2015). Sign Language Recognition using 3D convolutional neural networks. IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). Turin: IEEE.

[17] Aditi Kalash, N.S.Garewal, "Sign Language Recognition –Research in Computer Science and Software Engineering", Vol 3, Issue 9, pp.103-106, 2013

[18] Sruthi Upendran, Thamizharasi. An "American Sign Language Interpreter System for Deaf and Dumb Instrumentation, Communication and Computational Technologies (ICCICCT) 2014.

[19] Indriani, Moh.Harris, Ali Suryaperdana Agoes, "Applying Hand Gesture Recognition for User Guide Application Using MediaPipe", International Seminar of Science and Applied Technology (ISSAT 2021), Advances in Engineering Research, volume 207

[20] Sanket Wankhede, Ms.S .B.Jadhav, et al, "A Survey on "Securing Aggregate Queries for DNA Databases", International Journal of Innovative Research in Science, Engineering and Technology 2018, Volume 7 Issue 2 Pages 1208-1213

[21] Parimal Ghodke, Shailaja Jadhav, et al, " Validation Of Warranty Defect Codes To Ensure Vehicle Quality Within Warranty", International Journal of Computer Sciences and Engineering 2019, Volume 7 Issue -5 Pages 873-876