# NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM, APPROVED BY AICTE & GOVT.OF KARNATAKA)

## PROJECT REPORT – PHASE 3

on

# "INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE"

*Submitted in partial fulfillment of the requirement for the award of Degree of*

*Bachelor of Engineering*

*in*

*Computer Science and Engineering*

*Submitted by:*


| | |
|---|---|
| **PARINITHA P** | **1NT20CS124** |
| **SHREYA** | **1NT20CS167** |
| **SWATHI K T** | **1NT20CS190** |
| **UDAY KUMAR G** | **1NT20CS200** |

Under the Guidance of

**Dr. Nirmala J Saunshimath**
**Assistant Professor**, Dept. of CS&E, NMIT


Department of Computer Science and Engineering
**(Accredited by NBA Tier-1)**

2023-2024

# NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM
, APPROVED BY AICTE & GOVT.OF KARNATAKA)

## Department of Computer Science and Engineering

**(Accredited by NBA Tier-1)**

## CERTIFICATE

This is to certify that the INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE is an authentic work carried out by **PARINITHA P (1NT20CS124), SHREYA (1NT20CS167), SWATHI K T (1NT20CS190), and UDAY KUMAR G (1NT20CS200)** bonafide students of **Nitte Meenakshi Institute of Technology**, Bangalore in partial fulfilment for the award of the degree of ***Bachelor of Engineering*** in COMPUTER SCIENCE AND ENGINEERING of Visvesvaraya Technological University, Belgavi during the academic year ***2023-2024.*** It is certified that all corrections and suggestions indicated during the internal assessment has been incorporated in the report. This project has been approved as it satisfies the academic requirement in respect of project work presented for the said degree.

| **Internal Guide** | **Signature of the HOD** | **Signature of Principal** |
|---|---|---|
| Dr. Nirmala J Saunshimath<br>Assistant Professor, Dept. CSE, NMIT Bangalore | Dr. Vijaya Shetty S<br>Professor, Head, Dept. CSE, NMIT Bangalore | Dr. H. C. Nagaraj<br>Principal, NMIT Bangalore |

**Signature of Examiner (**Internals**)**

1.________________

**Signature of Examiner (**SEE**)**

1. ______________________

# DECLARATION

We hereby declare that

(i) The project work is our original work

(ii) This Project work has not been submitted for the award of any degree or examination at any other university/College/Institute.

(iii) This Project Work does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

(iv) This Project Work does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a) their words have been re-written but the general information attributed to them has been referenced;

b) where their exact words have been used, their writing has been placed inside quotation marks, and referenced.

(v) This Project Work does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

| NAME | USN | Signature |
|---|---|---|
| **PARINITHA P** | **1NT20CS124** | |
| **SHREYA** | **1NT20CS167** | |
| **SWATHI K T** | **1NT20CS190** | |
| **UDAY KUMAR G** | **1NT20CS200** | |

Date: 25-03-2024

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success. I express my sincere gratitude to our Principal **Dr. H. C. Nagaraj**, Nitte Meenakshi Institute of Technology for providing facilities.

We wish to thank our HoD**, Dr. Vijaya Shetty S** for the excellent environment created to further educational growth in our college. We also thank her for the invaluable guidance provided which has helped in the creation of a better project.

I hereby like to thank our **Dr. Nirmala J Saunshimath**, Assistant Professor, Department of Computer Science & Engineering on her periodic inspection, time to time evaluation of the project and help to bring the project to the present form.

Thanks to our Departmental Project coordinators. We also thank all our friends, teaching and non-teaching staff at NMIT, Bangalore, for all the direct and indirect help provided in the completion of the project.


| NAME | USN | Signature |
| --- | --- | --- |
| **PARINITHA P** | **1NT20CS124** | |
| **SHREYA** | **1NT20CS167** | |
| **SWATHI K T** | **1NT20CS190** | |
| **UDAY KUMAR G** | **1NT20CS200** | |

Date: 25-03-2024

# ABSTRACT

The purpose of our project title as ‘INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE’ is to recognize hand gestures and build a vision-based recognition system that employs both manual and facial features, extracted from the same input image (of a video) and converting it to local language. The focus of this work is to create a vision-based application which offers sign language translation to text thus aiding communication between signers and non-signers. We want to implement this idea in real time. Output will be in the form of text and speech.

The utilization of hand signs as a potent means of human-to-human communication has been widely acknowledged, offering a natural mode of interaction particularly beneficial for individuals with speech impairments. In India, where approximately one percent of the population falls into this category, the incorporation of a system capable of understanding Indian Sign Language (ISL) stands to have profound positive impacts. This paper introduces a novel approach employing the Bag of Visual Words model (BOVW) to discern ISL alphabets (A-Z) and digits (0-9) within live video streams, with outputs translated into text, speech, and localized languages such as Kannada and Hindi. Segmentation techniques based on both skin color detection and background subtraction facilitate precise identification. Feature extraction is achieved through Speeded Up Robust Features (SURF), with resulting histograms facilitating the mapping of signs to their corresponding labels. Classification is carried out using both Support Vector Machine (SVM) and Convolutional Neural Networks (CNN). Additionally, an interactive Graphical User Interface (GUI) has been developed to ensure user-friendly accessibility to the system. Through this innovative approach, individuals utilizing ISL can seamlessly engage in communication, thereby enhancing their social inclusion and quality of life.

# TABLE OF CONTENTS

## List of tables



## List of Acronyms

SLR - Sign Language Recognition

CNN - Convolution Neural Network

RNN - Recurrent Neural Network

BLSTM-3D - 3Dimensional Residual ConvNet and bi-directional LSTM networks

MLP - multi layer perceptron

HMM - hidden Markov model

WLASL - Word Level American Sign Language

KNN - K-Nearest Neighbors

SVM - Support Vector Machine

# List of Figures

# CHAPTER 1: INTRODUCTION

## 1.1 Background

Sign language is the primary mode of communication between hearing and vocally impaired population. People with disabilities or differently able are often isolated from accessing proper health, education and other social interactions. Assistive technology can change the life of a differently able person to a great extent in all means.

However, researchers have been working on the problem for quite some time, and the results are promising. Although interesting technologies for voice recognition are becoming available, there is currently no commercial solution for sign recognition on the market.

A solution to this problem is an automated sign language translator which can translate sign language(includes facial expression, manual features) into natural language and outputs in text and speech and also the reverse, speech to signs.

So here is the computer-based method for regular people to understand what the differently-abled individual is trying to say.

## 1.2 Brief history of Technology/concept

- Machine learning was first conceived from the mathematical modeling of neural networks. A paper by logician Walter Pitts and neuroscientist Warren McCulloch, published in 1943, attempted to mathematically map out thought processes and decision making in human cognition
- In 1957, Frank Rosenblatt – at the Cornell Aeronautical Laboratory – combined Donald Hebb's model of brain cell interaction with Arthur Samuel's machine learning efforts and created the perceptron. The perceptron was initially planned as a machine, not a program. The software, originally designed for the IBM 704, was installed in a custom-built machine called the Mark 1 perceptron, which had been constructed for image recognition.
- Machine learning (ML) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so.
- Deep Learning (DL) is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain.
- Deep learning is based on the concept of artificial neural networks, or computational systems that mimic the way the human brain functions. And so, our brief history of deep learning must start with those neural networks. In 1943, Warren McCulloch and Walter Pitts create a computational model for neural networks based on mathematics and algorithms called threshold logic.
- Overall, it is expected that this study will facilitate knowledge accumulation and creation of intelligent-based SLR and provide readers, researchers, and practitioners a roadmap to guide future direction.

## 1.3 Application

The 'INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE' incorporates one of the latest DL application. This application helps for dumb and deaf people. Using this application regular people can understand the differently abled people's sign language.

Speech impaired people use hand signs and gestures to communicate. Normal people face difficulty in understanding their language. Hence there is a need of a system which recognizes the different signs, gestures and conveys the information to the normal people. It bridges the gap between physically challenged people and normal people.

## 1.4 Research motivation and Problem statement

### 1.4.1 Research Motivation

Hearing loss is the most common sensory deficit in humans today. As per WHO estimates in India, there are approx. 63 million people, who are suffering from Significant Auditory Impairment; this places the estimated prevalence at 6.3% in Indian population.

These statistics formed the motivation for our project. As these speech impairment and deaf people need a proper channel to communicate with normal people there is a need for a system. Not all normal people can understand sign language of impaired people. Our project hence is aimed at converting the sign language gestures into text that is readable for normal people.

We are going to recognize hand gestures and build a vision-based recognition system that employs both manual and facial features, extracted from the same input image (of a video) and converting it to local language.

### 1.4.2 Statement of the Problem

Speech impaired people use hand signs and gestures to communicate. Normal people face difficulty in understanding their language. Hence there is a need of a system which recognizes the different signs, gestures and conveys the information to the normal people. It bridges the gap between physically challenged people and normal people.

So our goal is to build sign language detection using video sequence.

Effective extension of this project to words and common expressions may not only make deaf and dumb people communicate faster and easier with outer world, but also provide a boost in developing autonomous systems for understanding and aiding them.

## 1.5 Research objectives and contributions

### 1.5.1 Objectives

**Objective1:** Collecting dataset for recognition of sign languages and preprocessing the dataset and selection of algorithms.

**Objective2:** To predict sign language from hand gesture and converting into letters and voice.

**Objective3:** Converting voice to sign language.

**Objective4:** Converting the sign of letters to words and converting to local language (kannada, Hindi).

### 1.5.2 Contribution

Following are the contributions to our work:

**Research and Understanding:** Examine the technology and techniques currently in use for sign language interpreting. Understand the linguistic facets of sign languages to guarantee precise translation.

**Data Collection:** To train your model, compile a large dataset of sign language expressions and movements. Work together with communities and sign language specialists to make sure regional and cultural variations are taken into account.

**Recognition of Gestures:** Use real-time gesture detection technologies to capture the subtleties of sign language. In order to precisely record and understand motions, think about using sensors, cameras, or wearable technology.

**Designing User Interfaces (UI):** Provide a simple and easy-to-use interface so that those who are deaf or mute can communicate with the system. Make sure the method is simple to incorporate into different settings and everyday life.

**Testing and Input:** To make sure the system is accurate and usable, thoroughly test it with people who are deaf or mute. Get input, then iteratively enhance the product based on user experiences.

## 1.6 Organization of the report

In chapter1, we have given the introduction of sign language recognition system, what technologies we are going to use to achieve the aim that we have set. We have given a brief description about the background and research objectives of the project.

## 1.7 Summary

In this introductory chapter, the focus is on the development of an automated sign language translator using machine learning and deep learning technologies. Highlighting the background, the chapter emphasizes the isolation faced by differently-abled individuals due to communication barriers and underscores the transformative potential of assistive technology. A historical overview traces the evolution of machine learning, from neural networks to deep learning, setting the stage for the proposed solution.

The application section introduces the 'INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE,' a deep learning application aimed at facilitating communication for speech-impaired individuals. Motivated by the prevalence of hearing loss, the research aims to bridge the communication gap by converting sign language gestures into readable text for the general population. The problem statement underscores the challenges faced by speech-impaired individuals and articulates the need for a system that recognizes signs and gestures.

The research objectives encompass the detection of sign language through video sequences, conversion to text using object recognition and natural language processing (NLP), and further transformation into voice in the local language. Contributions include technological examination, data collection collaboration, gesture recognition, user interface design, and comprehensive testing. The chapter sets the stage for subsequent exploration in the literature survey.

# CHAPTER 2: LITERATURE SURVEY

## 2.1 Introduction

An Extensive literature survey was conducted over several weeks that concluded with us learning about various other SLR applications created around the world. This yielded different applications for our project and exposed us to different technologies used in the process. We were also able to analyze problems in the existing SLR applications that we intend on fixing.

## 2.1 Related work

**PAPER 1**: **American Sign Language Recognition using Deep Learning and Computer Vision**

Kshitij Bantupalli (Dept of CS Kennesaw State University Kennesaw, USA), Ying Xie (Dept of CS Kennesaw State University Kennesaw, USA)

The focus of this work is to create a vision-based application which offers sign language translation to text thus aiding communication between signers and non-signers. The proposed model takes video sequences and extracts temporal and spatial features from them. They then use Inception, a CNN (Convolutional Neural Network) for recognizing spatial features. They then use a RNN (Recurrent Neural Network) to train on temporal features.

**LIMITATIONS:**

The model also performed poorly when there was variation in clothing. We will try to improve this model for all type of dress. The model also suffered from loss of accuracy with the inclusion of faces, as faces of signers vary.

**PAPER 2**: **An Efficient Two-Stream Network for Isolated Sign Language Recognition Using Accumulative Video Motion**

Hamzah Luqman

Sign language is the primary communication medium for persons with hearing impairments.

This language depends mainly on hand articulations accompanied by nonmanual gestures. We propose a hierarchical sign learning module that comprises three

networks: dynamic motion network (DMN), accumulative motion network (AMN), and sign recognition network (SRN). Additionally, we propose a technique to extract key postures for handling the variations in the sign samples performed by different signers.

**LIMITATIONS**:

It showed less accuracy with signer independent datasets.

**PAPER 3: AMERICAN SIGN LANGUAGE FINGERSPELLING RECOGNITION WITH PHONOLOGICAL FEATURE-BASED TANDEM MODELS**

Taehwan Kim, Karen Livescu, Gregory Shakhnarovich Toyota Technological Institute at Chicago, Chicago, IL, USA. They study the recognition of fingerspelling sequences in American Sign Language from video using tandem-style models, in which the outputs of multilayer perceptron (MLP) classifiers are used as observations in a hidden Markov model (HMM)-based recognizer. They focus on recognition of one constrained but important part of the language: fingerspelling, in which signers spell out a word as a sequence of handshapes or hand trajectories corresponding to individual letters.

**LIMITATIONS:**

For the more challenging visual conditions, more sophisticated visual analysis may be needed, for example explicitly accounting for motion, pose, or 3-D structure. From a linguistic perspective, they are interested in the coarticulation that occurs in fingerspelling at different rates and believe that phonological feature models may be particularly well-suited to handle such effects.

**PAPER-4**: **WORD-LEVEL DEEP SIGN LANGUAGE RECOGNITION FROM VIDEO: A NEW LARGESCALE DATASET AND METHODS COMPARISON**

Dongxu Li, Cristian Rodriguez Opazo, Xin Yu, Hongdong Li TheAustralian National University, Australian Centre for Robotic Vision (ACRV)

In this paper, they introduce a new large-scale Word-Level American Sign Language (WLASL) video dataset, containing more than 2000 words performed by over 100 signers. This dataset will be made publicly available to the research community. Based on this new large- scale dataset, we are able to experiment with several deep learning methods for word level sign recognition and evaluate their performances in large scale scenarios.

**LIMITATIONS:**

we at utilizing word-level annotations to facilitate sentence-level and story-level machine sign translations.

**PAPER-5** :**Signet: A Deep Learning based Indian Sign Language Recognition System**

Sruthi C. J and Lijiya A Mem be r, IEEE

In this paper, the authors proposed a signer independent deep learning based methodology for building an Indian Sign Language (ISL) static alphabet recognition system. Here, they review various existing methods in sign language recognition and implement a Convolutional Neural Network (CNN) architecture for ISL static alphabet recognition from the binary silhouette of signer hand region.
The proposed method was successfully implemented with an accuracy of 98.64% which is better than most of the currently existing methods.

**LIMITATIONS**:

In this work they used binary silhouette of signer hand region as input to the CNN. This can be extended to a method which can process videos or images taken directly from mobile camera or any external camera.

## 2.2 Study of Tools/Technology

When it comes to Machine Learning, Artificial Neural Networks perform really well. Artificial Neural Networks are used in various classification tasks like image, audio, words.
similarly for image classification we use Convolution Neural networks.

In traditional neural networks, all the inputs and outputs are independent of each other, but in cases like when it is required to predict the next word of a sentence, the previous words are required and hence there is a need to remember the previous words. Thus, RNN came into existence, which solved this issue with the help of a Hidden Layer. The main and most important feature of RNN is Hidden state, which remembers some information about a sequence. Hence this will be useful for deaf and dumb people.
CNN (Convolution neural network), RNN (Recurrent Neural Network), Active appearance model Tandem model, Edge orientation histogram, CNN, Cycle GAN ,CNN,BLSTM-3D,SLR based on CNNs and Attention model.

## 2.3 Summary

This table gives a summary of the papers literature survey is done so far.

Table 1: Summary of literature survey

| AUTHOR | TITLE | METHOD | ACCURACY | LIMITATION |
|---|---|---|---|---|
| K. Bantupalli and Y. Xie, 2018 | American Sign Language Recognition using Deep Learning and Computer Vision | CNN(Convolution neural network) RNN(Recurrent Neural Network) | 99% | Various type of dresses and varying face is not recognised. |
| U. von Agris, M. Knorr and K. -F. Kraiss, 2008 | The significance of facial features for automatic sign language recognition | Active appearance model | 88-92% | Facial feature extraction is applied only to the interested areas such as eyes and mouth. |
| T. Kim, K. Livescu and G. Shakhnarovich,2012 | American sign language fingerspelling recognition with phonological feature-based tandem models | Tandem model | 90% | For the more challenging visual conditions, more sophisticated visual analysis may be needed. |

| | | | | |
|---|---|---|---|---|
| D. Li, C. R. Opazo, X. Yu and H. Li,2020 | Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison | Edge orientation histogram | 82.1% | we at utilizing word-level annotations to facilitate sentence-level and story-level machine sign translations |

| | | | | |
|---|---|---|---|---|
| Shruti. C.J. and Lijiya. A 2019 | Signet: A Deep Learning based Indian Sign Language Recognition System | CNN | 98.64% | In this work they used binary silhouette of signer hand region as input to the CNN. This can be extended to a method which can process videos or images taken directly from mobile camera or any external camera. |
| Maria Parelli, Katerina Papadimitriou | Exploiting 3D Hand Pose Estimation in Deep Learning-Based Sign Language Recognition from RGB Videos | Cycle GAN ,CNN | 94.56% | They did not implement it for local language so we can make implement that. We can also convert it to voice |
| Y. Liao, P. Xiong, W. Min, W. Min and J. Lu 2019 | Dynamic Sign Language Recognition Based on Video Sequence With BLSTM-3D Residual Networks | BLSTM-3D | 89. 9% | The sign sentence recognition for real world video in complex environment will be considered for further |

| | | | | |
|---|---|---|---|---|
| | | | | research it could not be achieved in this paper. |
| J. Huang, W. Zhou, H. Li and W. Li | Attention-Based 3D-CNNs for Large-Vocabulary Sign Language Recognition | SLR based on CNNs Attention model | 95.3% | They want to pay attention to continuous SLR in future work, which translates a sign video into a semantic sentence. |

# CHAPTER 3: SYSTEM REQUIREMENTS SPECIFICATIONS

## 3.1 General Description

This research proposal outlines the development of a real-time Sign Language (SL) finger-spelling recognition system using convolutional neural networks (CNNs) from camera images. The project aims to collect diverse SL gesture data, preprocess it, and train a CNN to recognize hand gestures based on position and orientation. The proposed architecture will filter and classify hand images, allowing for real-time recognition. The system's potential applications range from aiding the deaf and hard of hearing in communication to enhancing human-computer interaction. Ethical considerations, data diversity, and future expansion into more complex SL signs are key elements of this promising research endeavor.

### 3.1.1 Product Perspective

The product perspective for an "Interpretation of Sign Language for Deaf and Dumb People" project is centered on creating a user-centric, real-time communication tool. This technology aims to bridge the gap between the deaf and hearing worlds by providing immediate sign language interpretation. The product's design prioritizes accessibility and ease of use, catering to a diverse range of user needs. It must be privacy- conscious, provide user support, and seamlessly integrate with other communication platforms. Ethical considerations and ongoing research and development are key aspects, ensuring the product respects cultural and linguistic nuances and remains at the forefront of sign language technology. Cost considerations and sustainability strategies are also vital to maintain affordability and accessibility.

## 3.2 System Requirements

### 3.2.1 Hardware Requirements

Webcam, Camera or Imaging Device, Computer or Processing Unit, microphone, Graphics Processing Unit (GPU), Display, Redundancy and Backup.

### 3.2.2 software Requirements.

Operating System, Development Environment, Machine Learning Frameworks, Image and Video, Processing Libraries, Speech Recognition Software (Optional), Web Development Tools (Optional), Database Management System (Optional), Version Control System, Accessibility Tools, Testing and Quality Assurance Tools.

### 3.2.2.1 Functional Requirements & Non-functional

**Requirements Functional Requirement.**

1. Sign Language Recognition: Accurate recognition of a wide range of sign language gestures for effective communication.
2. Real-Time Processing: Swift processing of sign language gestures in real-time to ensure immediate communication.
3. User Interface: Intuitive and user-friendly interface allowing interaction in sign language, text, and speech.
4. Multilingual Support: Support for multiple sign languages to cater to a diverse user base.
5. Accessibility: Adherence to accessibility standards to make the system usable for individuals with different abilities.

**Non – Functional Requirement.**

1. Accuracy: High accuracy in recognizing sign language gestures for effective communication.
2. Latency: Low-latency processing for real-time communication, minimizing delays.
3. Reliability: High reliability with minimal downtime and robust error handling.
4. Usability: An intuitive and user-friendly interface for a positive user experience.
5. Security: Strong security measures to protect user data and communication, including encryption and authentication.

**User Requirements.**

1. User Registration and Profile Management:
   Use Case: New users can register, providing personal information and preferences.
   Mode of Operation: Online registration with secure data storage.
   User Class: Deaf and hearing users.
   Object Class: User profiles and preferences.

2. Gesture Recognition and Interpretation:

Use Case: Users can select their preferred sign language for communication.

Mode of Operation: User-configured language settings.

User Class: Deaf and hearing users.

Object Class: Language settings.

3. Accessibility and Usability:

Use Case: Users can adjust interface settings for accessibility, such as font size and contrast.

Mode of Operation: User-configured accessibility settings.

User Class: Users with varying accessibility needs.

Object Class: Accessibility settings.

4. Security and Privacy:

Use Case: User data and communication are secure and private, following established privacy guidelines.

Mode of Operation: Secure data encryption and user authentication.

User Class: All users.

Object Class: User data and communication channels.

## 3.3 Summary.

The "Interpretation of Sign Language for Deaf and Dumb People" System Requirements Specification (SRS) defines the essential features and functionalities of this technology solution. It starts with user-centric aspects, enabling new users to register and manage their profiles. The core functionality involves real-time gesture recognition and interpretation, facilitating immediate and effective communication for the deaf community. Multilingual support ensures inclusivity, allowing users to select their preferred sign language. Accessibility is a paramount consideration, with adjustable interface settings catering to various accessibility needs. Security and privacy measures are rigorously implemented, ensuring the protection of user data and confidential communications. An optional real-time voice input and output feature extends the system's reach, accommodating users with diverse communication preferences. The system's ability to expand its sign language vocabulary empowers users to contribute and adapt the platform to their evolving needs. This comprehensive SRS underscores the project's ethical responsibility, addressing cultural nuances and emphasizing user privacy.

In summary, this SRS delineates a user-friendly, secure, and adaptable technology solution that fosters effective communication, inclusivity, and user empowerment within the deaf and hard of hearing community. It paves the way for a transformative platform bridging the communication divide and ensuring user dignity and privacy.

# CHAPTER 4: DESIGN

## 4.1 Architectural Design

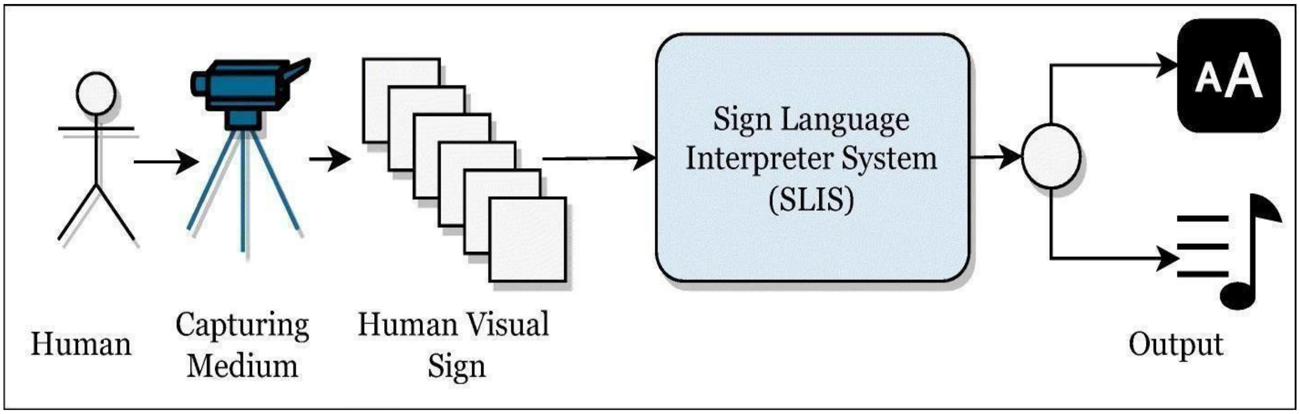



Fig. 4.1: Architectural design of the proposed system

The design depicts how the sign detection process takes place. When the signer shows signs, the camera captures video and converts it to frames and is passed to ISL interpreter and converts to text and voice.

## 4.2 Dataflow Diagram


Fig.4.2: DFD level 1

This DFD shows the user interaction with the interpreter.

## 4.3 Class Diagram

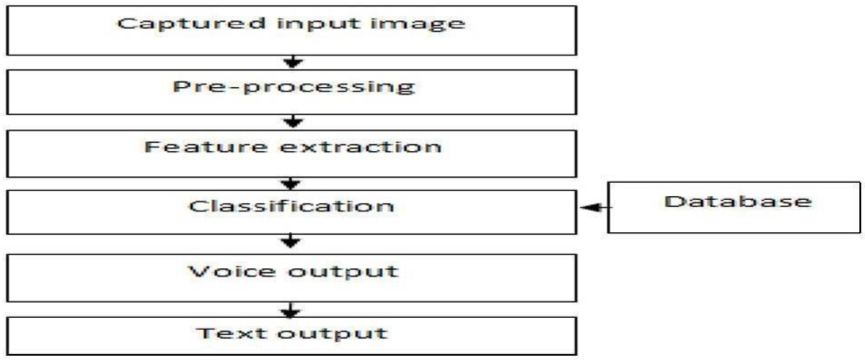



Fig.4.3: Class diagram of Proposed system

## 4.4 Use case Diagrams

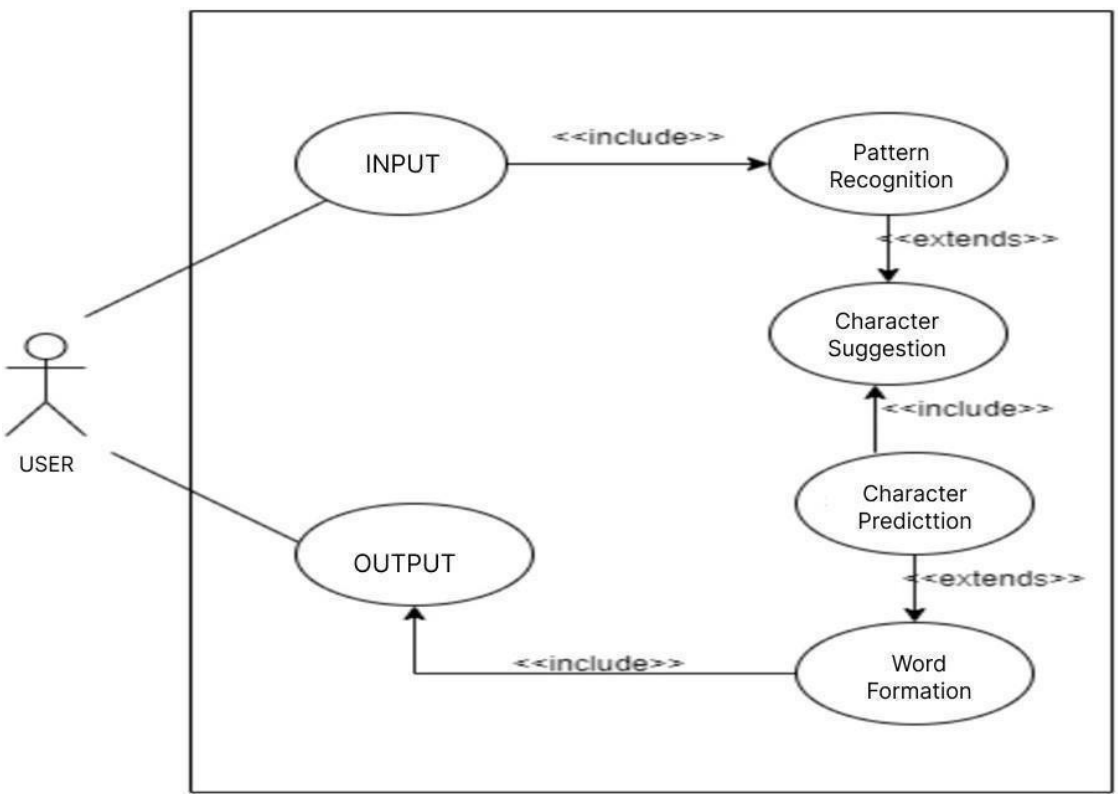



Fig. 4.4: Use case diagram

This diagram is a visual representation of the functional requirements and interactions between actors and the system.

## 4.5 Sequence Diagram

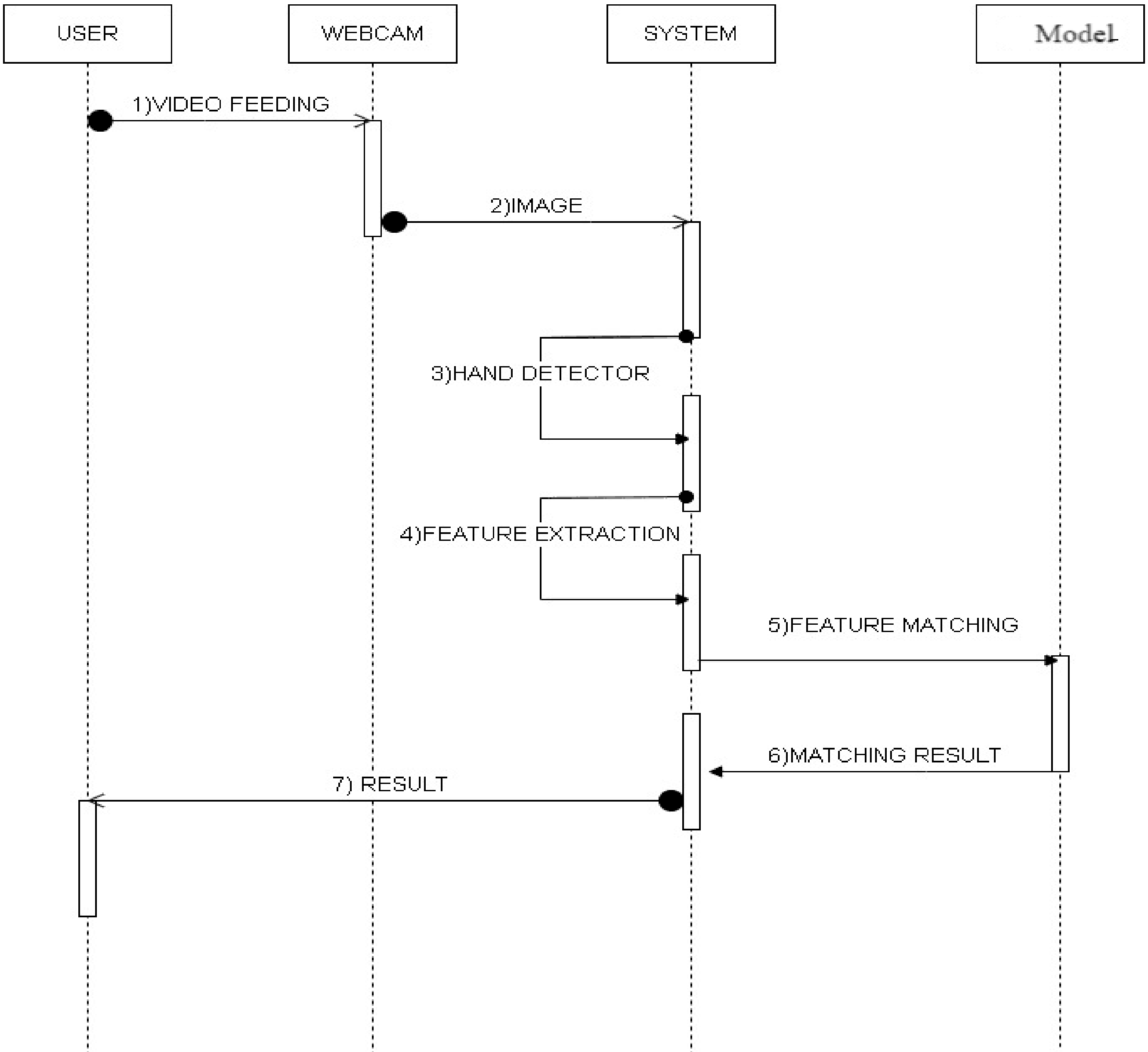



Fig .4.5: Sequence diagram

A sequence diagram for sign language represents the chronological order of events and interactions between elements, illustrating the dynamic exchange of sign language gestures and communication flow in a visual and sequential manner.

.

### 4.6 Activity Diagram

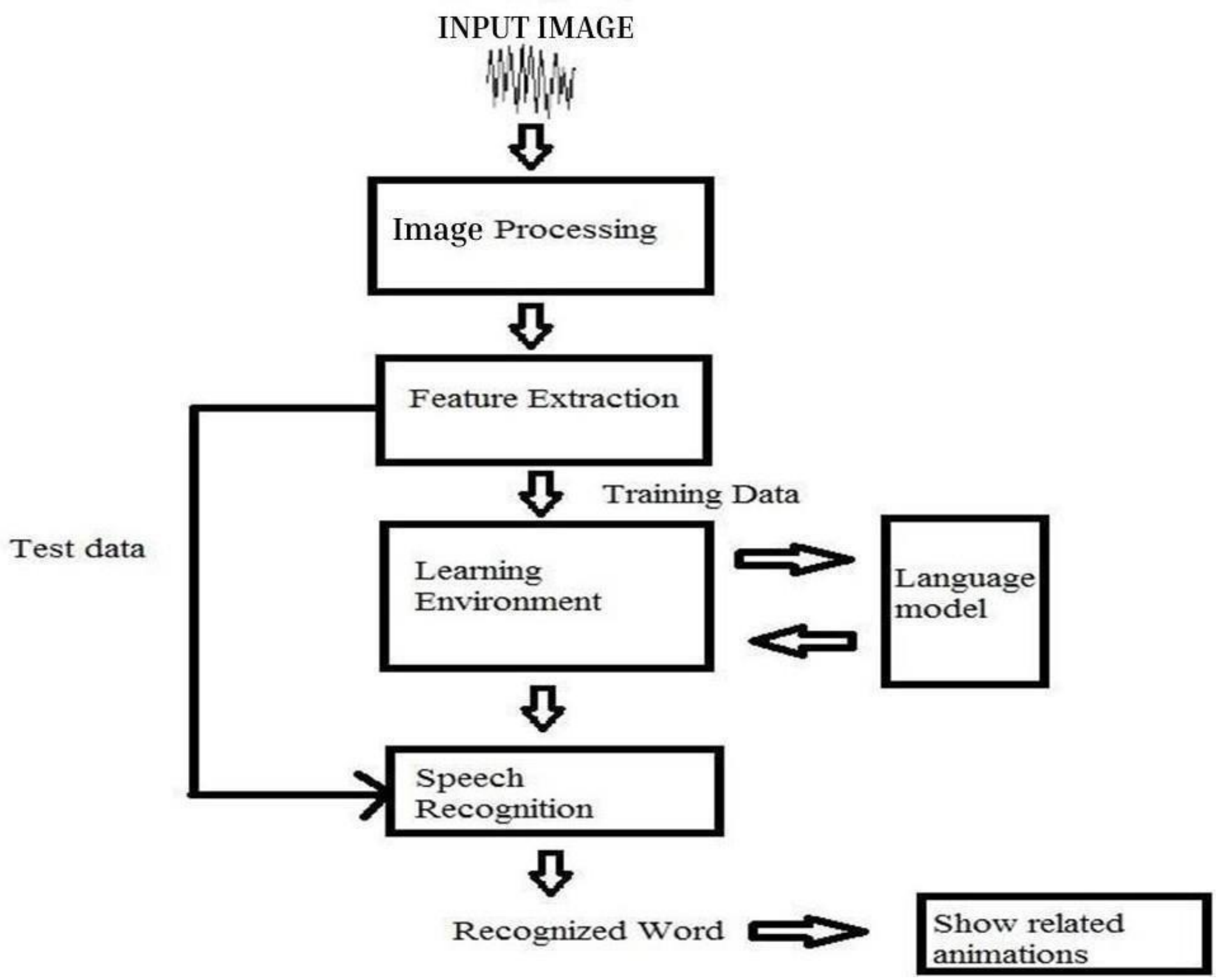



Fig. 4.6: Activity diagram

An activity diagram for sign language represents a visual depiction of the sequential and parallel activities involved in the interpretation and translation of sign language gestures into textual and spoken communication.

# CHAPTER 5: IMPLEMENTATION

## 5.1 Methodology

In order to build a highly accurate system that would be beneficial for real-time users, sign language recognition requires effective and reliable data. Here, the authors have addressed the sign detection and classification issue by utilizing the specially created dataset. Dataset, Image Acquisition, Data Pre-processing, Feature Extraction, and Sign Classification are the several steps at which the data flows for sign language recognition.

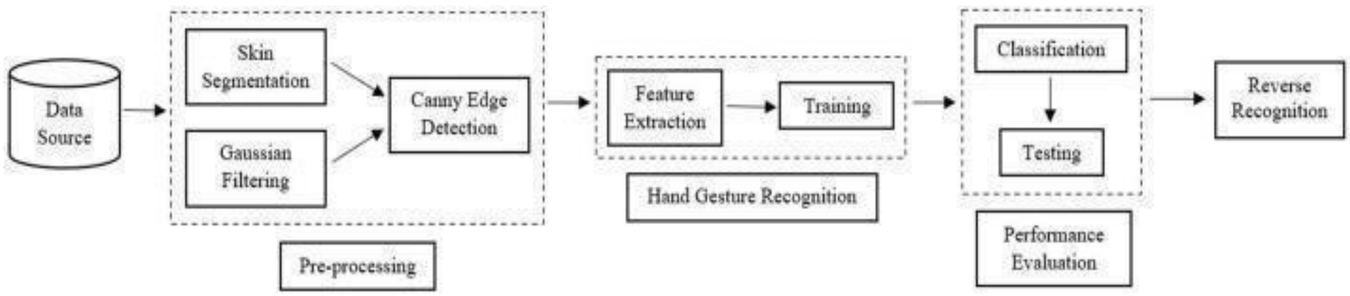



Fig.5.1 Flow diagram of the application

Here we are using different algorithms mainly Convolutional Neural Network, K-Nearest Neighbors .

**1. Convolutional Neural Network (CNN)**

Convolutional Neural Networks are used in the "Interpretation of Sign Language" project, taking use of their ability to recognize images. This breaks down obstacles to communication for the deaf by enabling real-time decoding of sophisticated sign language motions from video input. The research demonstrates how AI can promote inclusivity and increase global acceptance.

**2. Support Vector Machine (SVM)**

A key component of the "Interpretation of Sign Language for Deaf and Dumb People" project is Support Vector Machines (SVM). A key component of sign language communication is the ability to recognize and categorize complicated hand motions. SVM, a potent machine learning algorithm, helps with this. The deaf and dumb community may communicate easily since the SVM model is trained on a variety of sign language datasets, which teaches the system to reliably interpret and transform gestures into meaningful expressions. The algorithm is a useful tool for developing a dependable and efficient sign language interpretation system because of its capacity to manage non-linear relationships and categorize patterns.

## 5.2 Description Of Process

### 1. Data collection

Since data collection fosters the development of machine learning and deep learning models, it is a crucial part of research projects in all fields.The biggest challenge we faced during the data collection process was the absence of standardized databases for Indian sign language. Therefore, as part of this project, we attempted to manually create a dataset to aid in our problem-solving.

After the live video's signs were converted into frames, the frames were further extracted using a pixel value threshold. Pre-processing uses less computer power because the generated frames have a resolution of 250*250. In every folder, there were more than a thousand images of every sign. As a result, the dataset contained 36,000 images altogether from the two methods of image collection. To make the signs, one hand or both hands were needed. The images were captured at different angles and stored in grayscale, bearing the.jpg suffix.

### 2. Pre processing

The image is ready for feature extraction and detection in this step. In order to preserve scale homogeneity, every photo has the same dimensions.The video frame is captured and converted into the HSV color space for pictures taken in plain sight. The skin's tint can be easily removed because it is different from the background. After that, an experimental threshold is applied to the frame to determine its hue and eliminate skin-toned pixels. Furthermore, the image is binarized, noise is eliminated through blurring, and the output's maximum contour is taken, with the idea being that the contour with the biggest area represents the hands. Errors are further removed through the use of morphological techniques and the median filter.

### 3. Feature Extraction

This step includes developing a model codebook, creating a bag of visual words (BOVW), extracting features, and creating histograms.

Data retrieval and NLP's (Natural Language Processing) Bag of Words (BOW) provided the definition for the widely used image classification model known as the Bag of Visual Words (BOVW) [26]. This entails counting the occurrences of every word in a text, figuring out how frequently a word occurs to extract the keywords, and then using that data to create a frequency histogram. This idea is changed so that the image's characteristics are used in place of words. A language is created where each image is represented as a frequency histogram of acquired features using the image descriptors and key points. This frequency histogram can be used to predict the category of a subsequent image that is similar. SURF (Speeded Up Roburst Features), a feature detector and descriptor, is used for this.

IMage = {d1,d2,d3,….,dn}

Where di are the charectors like colors,shape etc. Obtained from SURF

The next step in the feature extraction process is to cluster all of the features that are generated after the SURF is applied. Related features are clustered together to use the core and cluster them as the visual keyword of the dictionary. We have decided to use micro batch K-means because of the size of the data, even though the K-means technique can be used for clustering as well. It is comparable to K-means, but requires less processing time and less memory. By employing small random batches of fixed-size data at a time, it does away with the requirement for all of the data to be in the memory at once. Up until convergence, the clusters are updated at each iteration using a new random sample drawn from the dataset.

## 4. Classification

After the feature detection stage we enter classifination stage, which is done using SVM(Support Vector Machine) and CNN(Convolutional Neural Network).

**i. SVM (Support Vector Machine)**

For classification and regression issues, the Support Vector Machine (SVM) is a supervised model that can resolve both linear and non-linear issues. It functions using the concept of decision planes, which define decision boundaries.

SVM with a linear kernel has been utilised for this classification. In order to classify and recognise ISL signs, we have fed the SVM the visual word histograms as feature vectors. 28,800 photos are used in total for the training. Following training, the classifier's performance is assessed using the testing set, which consists of 7236 images in total. The classifier's performance is measured using a number of criteria, including accuracy, precision, recall, and so on.

**ii. CNN (Convolutional Neural Network).**

CNNs are models of functional extraction inspired by the visual cortex of the human brain. When a filter map is applied to specific areas of a picture, CNNs analyse the images piece by piece. These segments are referred to as features, and they are used to compare two images by identifying nearly identical features at nearly same positions. When it comes to picture recognition and classification, CNNs outperform other neural networks. overall architecture, which consists of numerous convolutional and dense layers, is a fairly typical CNN architecture. Every CNN has three layers. The max-pool layer and dropout layer are placed after a set of two convolutional layers with a total of 32 filters and a 3 × 3 window size. A second set of two convolutional layers with 64 filters—a max pooling layer and a dropout layer—follows it. Lastly, there is a fully connected hidden layer with 512 neurons of the ReLU activation function and an output layer of the softmax activation function. There are also two more convolutional layers with 64 filters and a max pooling layer.

### 5. Reverse recognition

The reverse process is essential in a sign language recognition system to provide a dual mode of communication between the speech impaired and hearing majority. We have implemented this mode of communication in our system. Here text (English alphabets) is given as the input in the form of speech by the user, where it is mapped onto the labels and corresponding signs (images stored in database) are displayed to the user in a sequence. The speech recognition is done using the Google Speech API.

**6. Word formation and conversion to local language:** Detected signs were used to form the words using string operations. These words were then converted to Kannada using Google API.

## 5.3 Pseudo-Code

### 1.CNN Algorithm

```
function load_images(folder):
  train_data = []
  for label in list_of_labels_in_folder(folder):
    print(label, " Started!")
    path = folder + '/' + label
    for img_file in list_of_image_files_in_folder(path):
      img = read_image(path + '/' + img_file)
      new_img = resize_image(img, (100, 100))
    if new_img is not None:
    train_data.append([new_img, label])
    print(label, " ended!")
```

```
  return train_data
# Loading the train images and their corresponding labels
train_path = '/ISL Datasets/Train-Test/Train'
train_data = load_images(train_path)
# Mounting Google Drive
mount_google_drive('/content/drive')
# Loading the test images and their corresponding labels
test_path = 'ISL Datasets/Train-Test/Test'
test_data = load_images(test_path)
# Shuffling the data
shuffle(train_data)
shuffle(test_data)
# Separating features and labels
train_images = []
train_labels = []
test_images = []
test_labels = []
for feature, label in train_data:
  train_images.append(feature)
  train_labels.append(label)
for feature, label in test_data:
  test_images.append(feature)
  test_labels.append(label)
# Converting images list to numpy array
train_images = convert_to_numpy_array(train_images)
test_images =  convert_to_numpy_array(test_images)
# Reshaping images
train_images = reshape_images(train_images, (-1, 100, 100, 1))
test_images = reshape_images(test_images, (-1, 100, 100, 1))
# Changing the datatype and Normalizing the data
train_images = convert_and_normalize(train_images)
test_images = convert_and_normalize(test_images)
```

```
# Encoding the label values
label_encoder = create_label_encoder()
train_labels_encoded = label_encoder.fit_transform(train_labels)
test_labels_encoded = label_encoder.transform(test_labels)
# One hot encoding
train_labels_one_hot = one_hot_encode(train_labels_encoded)
test_labels_one_hot  = one_hot_encode(test_labels_encoded)
# Creating the Convolutional Neural Network Model
input_shape = (100, 100, 1)
num_classes = 36
model = create_cnn_model(input_shape, num_classes)
batch_size = 256
epochs = 100
# Compiling the model
compile_model(model, optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
# Displaying model summary
display_model_summary(model)
# Training the model
history = train_model(model, train_images, train_labels_one_hot, batch_size, epochs, test_images,
test_labels_one_hot)
# Evaluating the model on the test set
evaluate_model(model, test_images, test_labels_one_hot)
# Visualizing loss
plot_loss(history)
# Visualizing accuracy
plot_accuracy(history)

# Saving the trained model
save_model(model, '/Saved Files/CNN')
```

## 1. KNN Algorithm

```
# Loading and Preprocessing Data
data = read_csv('csv files/train.csv', low_memory=False)
test = read_csv('csv files/test.csv', low_memory=False)
x, y = extract_features_and_labels(data)
x_test, y_test = extract_features_and_labels(test)
label_encoder = LabelEncoder()
y = label_encoder.fit_transform(y)
y_test = label_encoder.transform(y_test)
# Choosing a K value
accuracy_rate = []
for i in range(1, 20):
    knn = KNeighborsClassifier(n_neighbors=i)
    score = cross_val_score(knn, x, y, cv=10)
    accuracy_rate.append(score.mean())
# Plotting accuracy with different values of k
plot_accuracy_vs_k(range(1, 20), accuracy_rate)
# Training and Predictions
k_value = 1
knn = KNeighborsClassifier(n_neighbors=k_value)
knn.fit(x, y
y_pred = knn.predict(x_test)
# Model Evaluation
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)
# Displaying Results
print("Accuracy Score:", accuracy)
print("Confusion Matrix:\n", conf_matrix)
print("Classification Report:\n", classification_rep)
# Plotting Confusion Matrix
```

```
plot_confusion_matrix(conf_matrix)
```

**2. SVM Algorithm**

```
# Loading and Preprocessing Data
data = read_csv('cnn files/train.csv', low_memory=False)
test = read_csv('cnn files/test.csv', low_memory=False)
x, y = extract_features_and_labels(data)
x_test, y_test = extract_features_and_labels(test)
label_encoder = LabelEncoder()
y = label_encoder.fit_transform(y)
y_test = label_encoder.transform(y_test)
# SVM with Linear Kernel
svm_classifier = SVMClassifier(kernel='linear')
svm_classifier.fit(x, y)
y_pred = svm_classifier.predict(x_test)
# Model Evaluation
accuracy = calculate_accuracy(y_test, y_pred)
classification_rep = generate_classification_report(y_test, y_pred)

# Displaying Results
print("Accuracy Score:", accuracy)
print("Classification Report:\n", classification_rep)
# Saving the trained SVM model
save_model(svm_classifier, '/content/drive/My Drive/Colab Notebooks/ISL Recognition/Saved
    Files/SVM')
# Confusion Matrix
conf_matrix = calculate_confusion_matrix(y_test, y_pred)
    plot_confusion_matrix(conf_matrix)
```

**3. Logistic Regression Pseudocode**

```
# Loading and Preprocessing Data
```

```
data = read_csv('cnn files/train.csv', low_memory=False)
test = read_csv('cnn files/test.csv', low_memory=False)
x, y = extract_features_and_labels(data)
x_test, y_test = extract_features_and_labels(test)
label_encoder = LabelEncoder()
y = label_encoder.fit_transform(y)
y_test = label_encoder.transform(y_test)

# Logistic Regression
logistic_regression = LogisticRegression(max_iter=100000)

# Displaying Data Head
display_data_head(x, x_test)

# Fitting the Model
logistic_regression.fit(x, y)

# Making Predictions
y_pred = logistic_regression.predict(x_test)

# Model Evaluation
accuracy = calculate_accuracy(y_test, y_pred)
classification_rep = generate_classification_report(y_test, y_pred)

# Displaying Results
print("Accuracy Score:", accuracy)
print("Classification Report:\n", classification_rep)

# Confusion Matrix
conf_matrix = calculate_confusion_matrix(y_test, y_pred)
```

```
plot_confusion_matrix(conf_matrix)
```

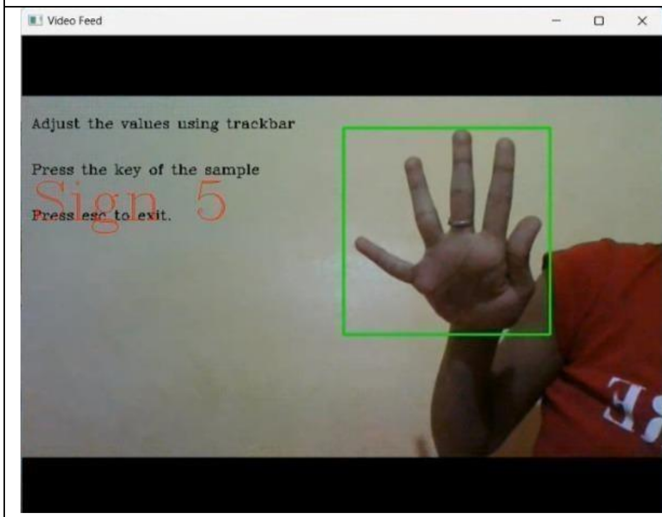
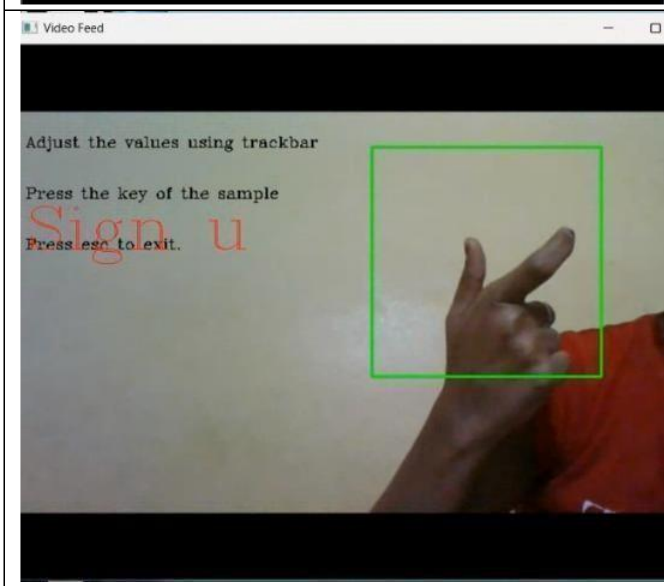
**4. Naive Bayes Algorithm**

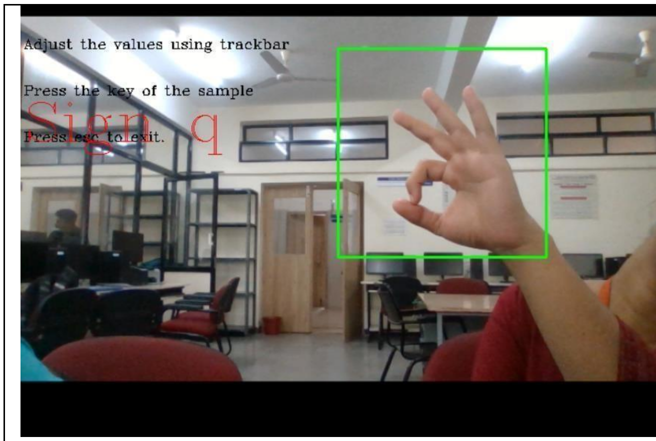
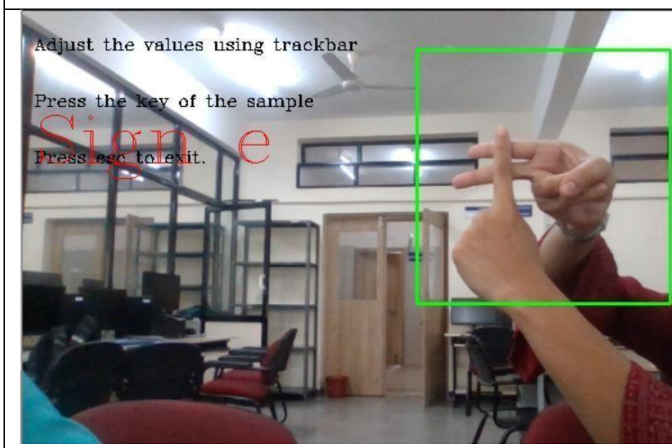
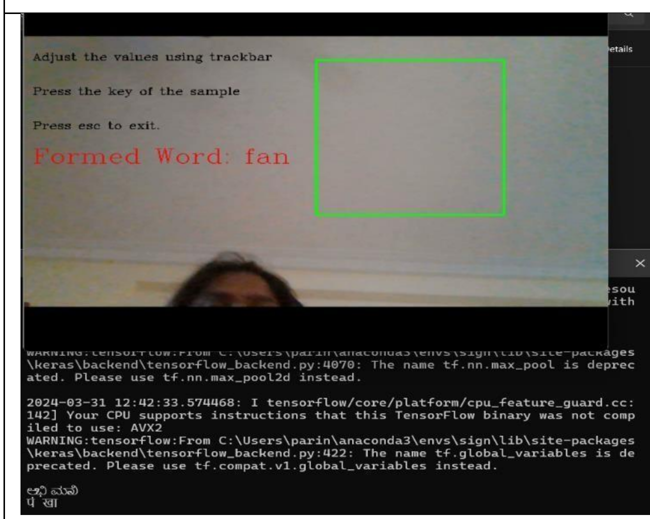
```
# Loading and Preprocessing Training Data
data = read_csv('cnn files/train.csv', low_memory=False)
x, y = extract_features_and_labels(data)

label_encoder = LabelEncoder()
y = label_encoder.fit_transform(y)
# Gaussian Naive Bayes
GaussNB = GaussianNaiveBayes()
print("Gaussian Naive Bayes Started!")
GaussNB.fit(x, y)
print("Gaussian Naive Bayes Finished!")
# Loading and Preprocessing Test Data
test = read_csv('cnn files/test.csv', low_memory=False)
x_test, y_test = extract_features_and_labels(test)
y_test = label_encoder.transform(y_test)
# Gaussian Naive Bayes Evaluation
print("Evaluating Gaussian Naive Bayes:")
y_pred_gauss = GaussNB.predict(x_test)
accuracy_gauss = calculate_accuracy(y_test,  y_pred_gauss)
conf_matrix_gauss = calculate_confusion_matrix(y_test, y_pred_gauss)
classification_rep_gauss = generate_classification_report(y_test, y_pred_gauss)
# Displaying Gaussian Naive Bayes Results
print("Accuracy Score (Gaussian Naive Bayes):", accuracy_gauss)
print("Confusion Matrix (Gaussian Naive Bayes):\n", conf_matrix_gauss)
print("Classification Report (Gaussian Naive Bayes):\n", classification_rep_gauss)
# Multinomial Naive Bayes
MultiNB = MultinomialNaiveBayes()
print("Multinomial Naive Bayes Started!")
```

```
MultiNB.fit(x, y)
print("Multinomial Naive Bayes Finished!")

# Multinomial Naive Bayes Evaluation
print("Evaluating Multinomial Naive Bayes:")
y_pred_multi = MultiNB.predict(x_test)
accuracy_multi = calculate_accuracy(y_test, y_pred_multi)
conf_matrix_multi = calculate_confusion_matrix(y_test, y_pred_multi)
classification_rep_multi = generate_classification_report(y_test, y_pred_multi)
# Displaying Multinomial Naive Bayes Results
print("Accuracy Score (Multinomial Naive Bayes):", accuracy_multi)
print("Confusion Matrix (Multinomial Naive Bayes):\n", conf_matrix_multi)
print("Classification Report (Multinomial Naive Bayes):\n", classification_rep_multi)

# Bernoulli Naive Bayes
BernNB = BernoulliNaiveBayes(binarize=True)
print("Bernoulli Naive Bayes Started!")
BernNB.fit(x, y)
print("Bernoulli Naive Bayes Finished!")

# Bernoulli Naive Bayes Evaluation
print("Evaluating Bernoulli Naive Bayes:")
y_pred_bern = BernNB.predict(x_test)
accuracy_bern = calculate_accuracy(y_test, y_pred_bern)
conf_matrix_bern = calculate_confusion_matrix(y_test, y_pred_bern)
classification_rep_bern = generate_classification_report(y_test, y_pred_bern)

# Displaying Bernoulli Naive Bayes Results
print("Accuracy Score (Bernoulli Naive Bayes):", accuracy_bern)
print("Confusion Matrix (Bernoulli Naive Bayes):\n", conf_matrix_bern)
```

# CHAPTER 6: TEST CASES

Table 2: Test Cases

| Input image | Testing output | Success/failure |
|---|---|---|
|  | 5 | Success |
|  | U | Success |

<table>
<tr>
<td>
</td>
<td>8</td>
<td>Failure</td>
</tr>
<tr>
<td>
</td>
<td>f</td>
<td>Failure</td>
</tr>
<tr>
<td>
</td>
<td>fan<br>ಅಭಿಮಾನಿ<br>पंखा</td>
<td>Success</td>
</tr>
</table>

This table gives the summary of various test cases of system developed. The table contains 2 success and 2 failure cases. Attempts to improve the results are being done.

# CHAPTER 7: RESULTS

a) Performance of SVM and CNN

SVM reported a 99.14% accuracy rate on the test set. An overall accuracy of 99% can be seen in the computed precision and recall values of the alphabets and digits identified using SVM.

On the training set during the most recent epoch, we have seen an overall accuracy of 94% using CNN, while a testing accuracy of more than 99%. Accuracy of CNN is calculated for 50 epochs for which a graph is drawn as shown in figure 8.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$$

Where, TP= true positives (TP)
TN= true negatives (TN)
FP= false positives (FP)
FN=false negatives (FN)

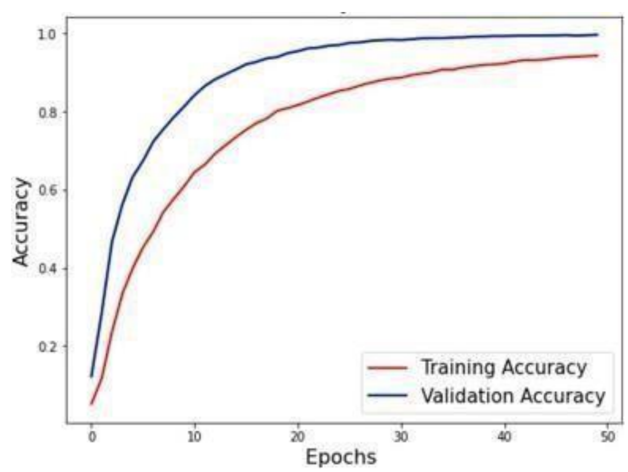



Fig. 7.1: Accuracy graph

Table 3: Accuracy measure of SVM and CNN

| Algorithm | Accuracy(in %) |
|---|---|
| SVM | 99.17 |
| CNN | 99.64 |

b) Workflow:

Begin with logging on to the application with your name and password, then getting access to the user panel where you will get the options like predict sign, translate speech, creating signs etc.

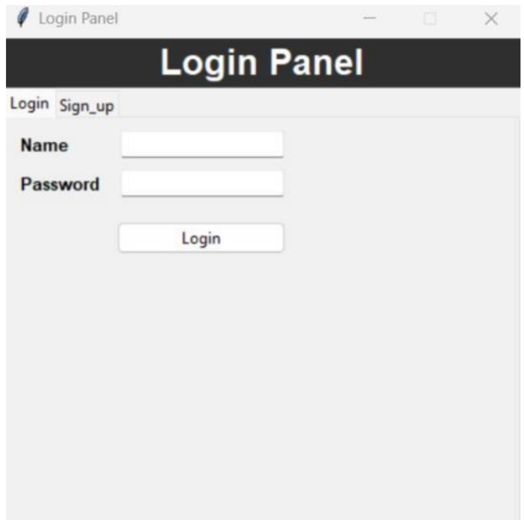



Fig. 7.2: Login page


Fig. 7.3: GUI of the Application

## Sign Prediction:

The following diagram will show sign prediction of **sign 3.**

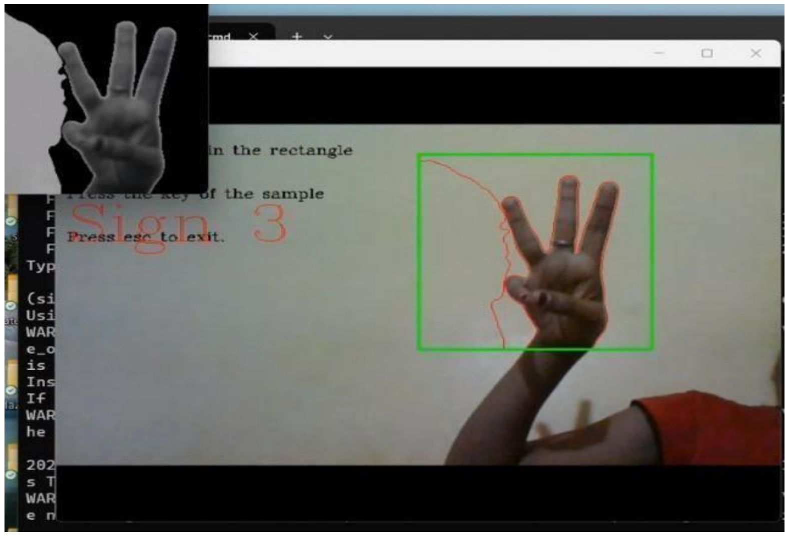



Fig .7.4: Sign predicting number 3

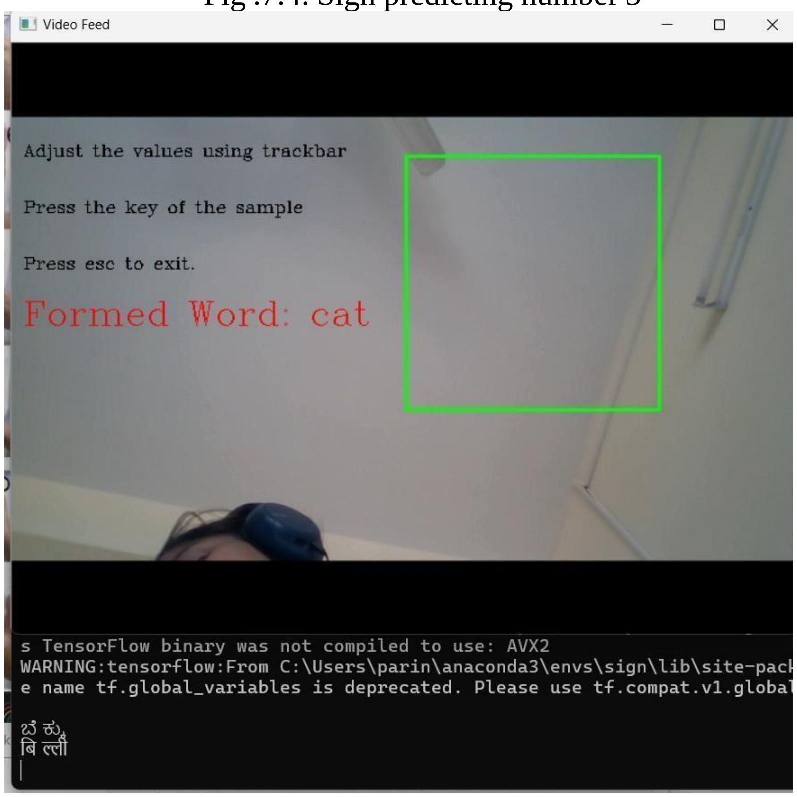



Fig. 7.5 Word formation and conversion to local language (kannada, hindi)

## Translate Speech:

The following diagram shows the "**Good Morning"** sign

Step 1: Select **Translate Speech** form user panel


Fig.7.6: Click ok after translate speech

Step 2: Say the word "Good Morning" so that the system will recognize it


Fig. 7.7: Click ok for recognizing the audio input

Step 3: Output display

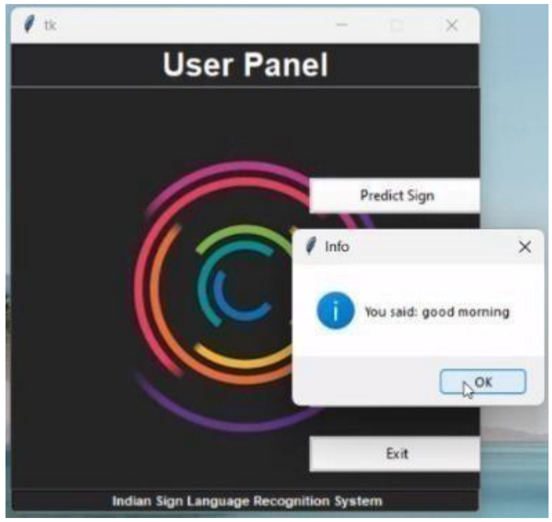



Fig .7.8: Dialogue box with the word

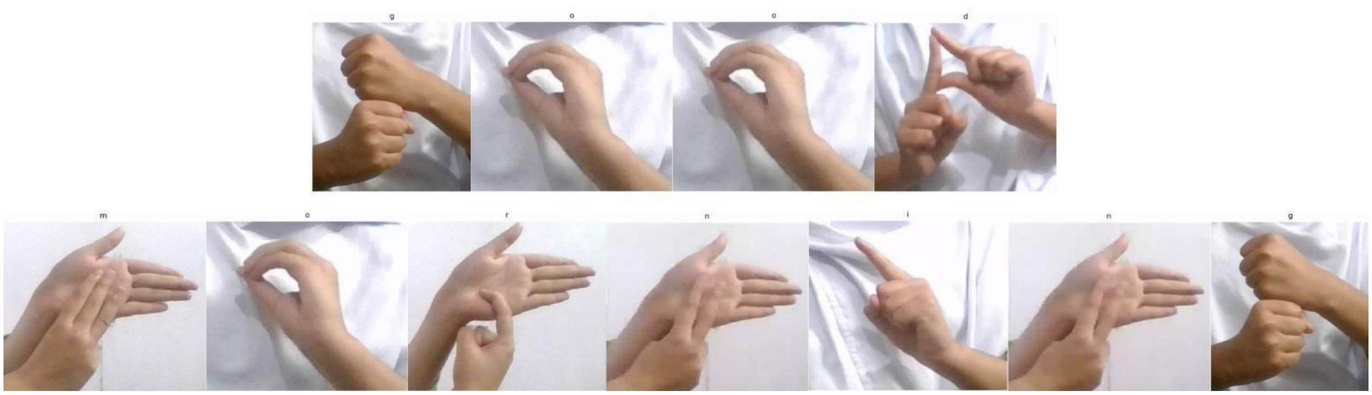


Fig. 7.9: Output signs for good morning

# CHAPTER 8: IMPACT OF OUR PROJECT TOWARDS OUR SOCIETY

The "Interpretation of Sign Language for Deaf and Dumb People" project is a powerful force that has profound effects on the environment and society. The project breaks down a long-standing barrier by facilitating real-time communication between the hearing and deaf communities, encouraging inclusivity and understanding. Deaf people can now actively engage in more facets of society, such as work and education, making the world more accessible and equitable.

Due to its promotion of empathy and dismantling of stereotypes related to communication difficulties, this project has a significant social impact. It improves the deaf community's quality of life by enabling them to participate meaningfully in social situations that were previously difficult for them. Families, workplaces, and public areas all benefit from the positive knock-on effect, which promotes a more diverse and accepting society.

Furthermore, through demonstrating the beneficial application of technology, the project is in line with environmental sustainability. The potential of technological innovation for social good is demonstrated by the application of artificial intelligence to enhance communication for the hard of hearing. People's general well-being increases in inclusive and supportive societies, which leads to a more peaceful and sustainable environment where technology advances humankind.

# CHAPTER 9: CONCLUSION

This work approach uses CNN and SVM to classify and recognize Indian sign language signs (0–9) and A–Z. Our primary objective is to develop a more real-time recognition utility, enabling the system to be utilized in any location. It is accomplished by creating a unique data set, solving the background dependency issue, and making the system rotationally invariant. With 99% accuracy, the system has been successfully trained on all 36 ISL static alphabets and digits. In the future, the dataset can be enlarged to include more signs in other languages spoken in different nations, creating a more functional framework for real-time applications. For both continuous and isolated recognition tasks, the technique can be expanded to form simple words and expressions. Accelerating the response time is the key to real-time applications.

# REFERENCES


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

[2] U. von Agris, M. Knorr and K. -F. Kraiss, "The significance of facial features for automatic sign language recognition," *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*, 2008, pp. 1-6, doi: 10.1109/AFGR.2008.4813472.

[3] T. Kim, K. Livescu and G. Shakhnarovich, "American sign language fingerspelling recognition with phonological feature-based tandem models," *2012 IEEE Spoken Language Technology Workshop (SLT)*, 2012, pp. 119-124, doi: 10.1109/SLT.2012.6424208.

[4] D. Li, C. R. Opazo, X. Yu and H. Li, "Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison," *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 1448-1458, doi: 10.1109/WACV45572.2020.9093512.

[5] S. C.J. and L. A., "Signet: A Deep Learning based Indian Sign Language Recognition System," *2019 International Conference on Communication and Signal Processing (ICCSP)*, 2019, pp. 0596-0600, doi: 10.1109/ICCSP.2019.8698006.

[6] Maria Parelli, Katerina Papadimitriou, Gerasimos Potamianos, Georgios Pavlakos, and Petros Maragos ,“Exploiting 3D Hand Pose Estimation in Deep Learning-Based Sign Language Recognition from RGB Videos”,

[7] Y. Liao, P. Xiong, W. Min, W. Min and J. Lu, "Dynamic Sign Language Recognition Based on Video Sequence With BLSTM-3D Residual Networks," in *IEEE Access*, vol. 7, pp. 38044-38054, 2019, doi: 10.1109/ACCESS.2019.2904749.

**[8] J. Huang, W. Zhou, H. Li and W. Li, "Attention-Based 3D-CNNs for Large-Vocabulary Sign Language Recognition," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 9, pp. 2822-2832, Sept. 2019, doi: 10.1109/TCSVT.2018.2870740.**

**[9] V. Slavuj , B. Kovacic and I. Jugo Department of Informatics of Rijeka, Croatia – "Intelligent Tutoring System for Language Learning"-May 2015**

**[10]G. Ananth Rao, P.V.V. Kishore Department of Electronics and communication Engineering- "Selfie video based continuous Indian sign language recognition system" – February 2017**

**[11] Satheesh Kumar Raju, Anil Kumar G S, Sunny Arokia Swamy Department of electrical and electronics Engineering-"Double Handed Indian Sign Language to Speech and Text"-2015**

**[12] Oscar Koller, Simon Hadfield, and Richard Bowden-"Multi channel Transformers for Multi-articulatory Sign Language translation."-2020**

**[13]V.V. Kishore, M.V.D. Prasad, Ch. Raghava prasad, R Rahul-" 4-Camera Model for sign language recognition using Elliptical Fourier descriptors and ANN"-2015**

**[14] Tripathi, Neha, G. C. Nandi-"Continuous Dynamic Indian Sign language Gesture Recognition with Invariant Backgrounds"-2015**

**[15] Daleesha M. Viswanathan, Sumam Mary Idicula "Recent developments in Indian sign language recognition: an analysis"**

**[16] Anuja V. Nair, V. Bindu "A review on Indian sign language recognition"**

**[17] K.K. Dutta, S.A.S. Bellary "Machine learning techniques for Indian sign language recognition"**

**[18] Garima Joshi, Vig Renu, Sukhwinder Singh "DCA-based unimodal feature-level fusion of orthogonal moments for Indian sign language dataset"**

**[19] K. Manikandan, Ayush Patidar, Pallav Walia, Aneek Barman Roy "Hand gesture detection and conversion to speech and text"**

**[20] J. Sivic, A. Zisserman "Video Google: a text retrieval approach to object matching in videos"**

**[21] J. Singha, K. Das: Recognition of Indian sign language in live video**

**[22] Shanmukha Swamy, M.P. Chethan, Mahantesh Gatwadi: Indian sign language interpreter with android implementation**

**[23] S.C. Agrawal, A.S. Jalal, C. Bhatnagar, Ieee: Recognition of Indian sign language using feature Fusion.**

# APPENDIX 1: PAPER/PATENT WITH CERTIFICATES

4/1/24, 11:43 AM Conference Management Toolkit – Submission Summary

## Submission Summary

**Conference Name**
IEEE MYSORE SUBSECTION FLAGSHIP CONFERENCE MYSURUCON 2024

**Track Name**
Track-5: Technology for Education and Entertainment

**Paper ID**
6

**Paper Title**
Sign Language Detection For Deaf And Dumb People

**Abstract**
People with speech and hearing impairments rely heavily on sign language in their daily lives. This group comprises approximately 1% of the Indian population. For these reasons alone, incorporating a framework that is capable of understanding Indian Sign Language would be extremely advantageous to these people. Our method in this study recognizes Indian sign language alphabets (A-Z) and numerals (0-9) in a live video stream using the Bag of Visual Words model (BOVW). It then outputs the expected labels as both text and audio. Segmentation is done based on skin colour as well as background subtraction. SURF (Speeded Up Robust Features) features have been extracted from the images and histograms are generated to map the signs with corresponding labels. The Support Vector Machine (SVM) and Convolution Neural Networks (CNN) are used for classification. An interactive Graphical User Interface (GUI) is also developed for easy access.

**Created**
4/1/2024, 11:38:46 AM

**Last Modified**
4/1/2024, 11:38:46 AM

**Authors**
Parinitha P (Nitte Meenakshi Institute Of Technology Yelahanka Bengaluru) <parinithap10@gmail.com>
Shreya Shreya (Nitte Meenakshi Institute Of Technology Yelahanka Bengaluru) <shreyamadivala68@gmail.com>
Swathi K T (Nitte Meenakshi Institute Of Technology Yelahanka Bengaluru) <1nt20cs190.swathi@nmit.ac.in>
Uday Kumar G (Nitte Meenakshi Institute Of Technology Yelahanka Bengaluru) <1nt20cs200.uday@nmti.ac.in>
**Nirmala S** (Nitte Meenakshi Institute Of Technology) <nirmala.saunshimath@nmit.ac.in>

**Submission Files**
INTERPRETATION OF SIGN LANGUAGE FOR DEAF AND DUMB PEOPLE (modified).pdf (334.9 Kb, 4/1/2024, 11:29:11 AM)

https://cmt3.research.microsoft.com/MYSURUCON2024/Submission/Summary/6 1/1

In order to demonstrate our contributions to the field's advancement, our team wrote a paper that was submitted to the IEEE MYSORE SUBSECTION FLAGSHIP CONFERENCE MYSURUCON 2024 on Technology for Education and Entertainment.

# APPENDIX 2: PLAGIARISM REPORT

**DrillBit**

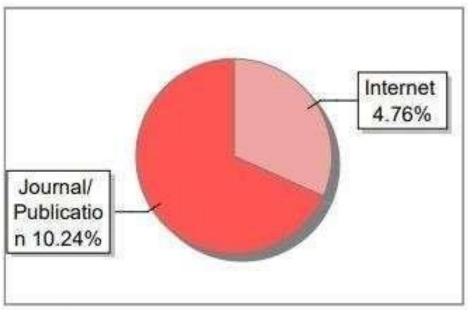
The Report is Generated by DrillBit Plagiarism Detection Software

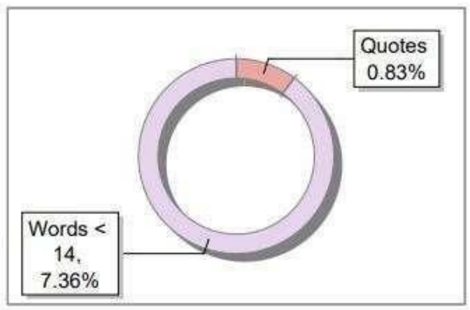
***Submission Information***

| | |
|---|---|
| Author Name | 1nt20cs160.shilpha@nmit.ac.in |
| Title | Cyber Security Tutorial 5 13/10/2023 |
| Paper/Submission ID | 1207708 |
| Submitted by | hod-library@nmit.ac.in |
| Submission Date | 2023-12-13 13:13:20 |
| Total Pages | 40 |
| Document type | Assignment |

***Result Information***

Similarity **15 %**




***Exclude Information***

| | |
|---|---|
| Quotes | Excluded |
| References/Bibliography | Not Excluded |
| Sources: Less than 14 Words % | Not Excluded |
| Excluded Source | **0 %** |
| Excluded Phrases | Not Excluded |

***Database Selection***

| | |
|---|---|
| Language | English |
| Student Papers | Yes |
| Journals & publishers | Yes |
| Internet or Web | Yes |
| Institution Repository | Yes |

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

**DrillBit Similarity Report**

| SIMILARITY % | MATCHED SOURCES | GRADE | |
|---|---|---|---|
| 15 | 54 | B | A-Satisfactory (0-10%)<br>B-Upgrade (11-40%)<br>C-Poor (41-60%)<br>D-Unacceptable (61-100%) |

| LOCATION | MATCHED DOMAIN | % | SOURCE TYPE |
|---|---|---|---|
| 1 | INFORMATICSwww.aensiweb.net | 2 | Publication |
| 2 | www.dx.doi.org | 1 | Publication |
| 3 | theemcoe.org | 1 | Publication |
| 4 | www.dataversity.net | 1 | Internet Data |
| 5 | www.socpros2023.iitr.ac.in | 1 | Publication |
| 6 | arxiv.org | 1 | Publication |
| 7 | www.mdpi.com | <1 | Internet Data |
| 8 | arxiv.org | <1 | Publication |
| 9 | www.mdpi.com | <1 | Internet Data |
| 10 | etd.aau.edu.et | <1 | Publication |
| 11 | DEVELOPMENT AND TESTING OF A JUNIOR HIGH SCHOOL ORAL HYGIENE EDUCATION PROGRAM by Josep-1970 | <1 | Publication |
| 12 | turcomat.org | <1 | Publication |
| 13 | www.frontiersin.org | <1 | Internet Data |
| 14 | Content-based analysis to detect Arabic web spam by Al-Kabi-2012 | <1 | Publication |

## APPENDIX 3:SELF ASSESSMENT OF PO-PSO ATTAINMENT

| **Program Outcomes (PO)** | **Justification** |
|---|---|
| PO1. Engineering Knowledge: | Demonstrated comprehensive engineering knowledge in developing a real-time system for Indian Sign Language recognition, encompassing data collection, pre-processing, feature extraction using Bag of Visual Words model, and classification employing Support Vector Machine (SVM) and Convolutional Neural Network (CNN), resulting in accurate identification of sign language alphabets and numerals. |
| PO2. Problem Analysis: | Understanding the challenges faced by individuals with speech and hearing impairments, the study thoroughly investigates the lack of accessible and affordable sign language recognition systems, particularly for Indian Sign Language (ISL). |
| PO3. Design Development of solutions: | Employing advanced techniques like Bag of Visual Words (BOVW), Support Vector Machine (SVM), and Convolutional Neural Networks (CNN), the system devises a robust framework capable of real-time recognition of ISL alphabets and numerals from live video streams. |
| PO5. Modern Tools: | Utilizing state-of-the-art tools and methodologies such as SURF feature extraction, SVM, CNN architectures, and Python-based libraries like OpenCV and Pyttsx3, the project harnesses the power of modern technology to tackle the intricate challenges of Indian Sign Language recognition. |
| PO6. The Engineer and society: | The project underscores the crucial role of engineers in addressing societal needs, particularly for marginalized communities like individuals with speech and hearing impairments. |
| PO8. Ethics: | Ethical considerations permeate every aspect of the research, from data collection and model training to system deployment. The project prioritizes privacy and consent, ensuring that individuals' rights and dignity are respected throughout the process. |
| PO9. Individual and Teamwork: | The project demonstrates a harmonious blend of individual expertise and collaborative teamwork, with pooling the diverse skills in computer vision, machine learning, and software development. |
| PO10. Communication: | Effective communication lies at the heart of the team endeavor, as evidenced by the development of an interactive Graphical User Interface (GUI) for easy accessibility. |

| | |
|---|---|
| PO11: Project management and finance: | The project demonstrates adept project management skills by meticulously outlining the methodology, allocating resources effectively, and adhering to timelines. |
| PO12. Life-long learning: | Embracing a spirit of continuous learning and innovation, the project reflects the ethos of life-long learning inherent to engineering professions. |

| **Program Specific Outcomes (PSO)** | **Justification** |
|---|---|
| | |
| PSO1. Professional Skills: | this paper introduces a novel approach that combines advanced techniques in computer vision and machine learning to address the communication needs of individuals using Indian Sign Language (ISL). By leveraging the Bag of Visual Words model (BOVW), segmentation techniques, and feature extraction methods such as Speeded Up Robust Features (SURF), the system can accurately discern ISL alphabets and digits in live video streams. |
| PSO2. Problem Solving Skills: | The research showcases adept problem-solving skills by identifying and addressing key challenges in Indian Sign Language recognition, including data scarcity, regional variations, and real-time processing constraints. |
| PSO3. Ethics and career development: | This project not only advances technical expertise but also underscores ethical responsibility and fosters career development by addressing communication barriers for individuals using Indian Sign Language. |