

Exploring Sign Language Recognition Using Convolutional Neural Network

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

SAMUEL CHRISTOPHER S (Register No. 40111122)

EDWIN SAMUEL M (Register No. 40110344)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY - 1 UNIVERSITY BY UGC

Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL - 2024



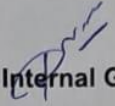
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)
Category - I University by UGC
Accredited "A++" by NAAC | Approved by AICTE
www.sathyabama.ac.in

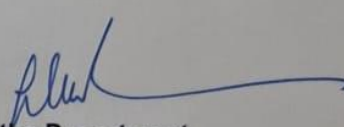
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Samuel Christopher S (40111122)** and **Edwin Samuel M (40110344)** who carried out the Project entitled "**Exploring Sign Language Recognition Using Convolutional Neural Network**" under my supervision from November 2023 to April 2024.

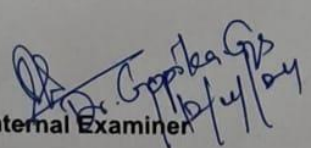

Internal Guide

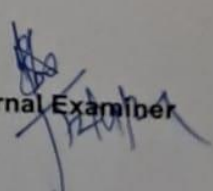
Dr. T. PREM JACOB., M.E., Ph.D.


Head of the Department

HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai - 600 119. Ph.D.

Submitted for Viva voce Examination held on 12.4.2024


Internal Examiner


External Examiner

DECLARATION

I, Samuel Christopher S (Reg. No. - 40111122) hereby declare that the Project Report entitled "Exploring Sign Language Recognition Using Convolutional Neural Network" done by me under the guidance of Dr. T. Prem Jacob., M.E., Ph.D. is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE: 12.4.2024

PLACE: Chennai

Samuel christopher . S

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala., M.E., Ph.D., Dean, School of Computing, Dr. L. Lakshmanan., M.E., Ph.D., Head of the Department of Computer Science and Engineering** for providing me with the necessary support during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. T. Prem Jacob M.E., Ph.D.**, for his valuable guidance, suggestions, and constant encouragement paved the way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Sign language recognition has been widely used for communication amongst the hearing-impaired, robotic communication, and non-verbal community. From early electric signal-based sign language recognition to modern-day recognition using machine/deep learning techniques, researchers all over the world have tried to automate this task. This project mainly aims to carry out key point detection-based sign language recognition. Used different machine learning algorithms like the random forest, support vector machine and K-Nearest Neighbor to train the model. Finally, the best model is selected from the model testing using evaluation metrics namely f1 score, precision and recall. MediaPipe is a powerful tool for hand tracking and key point detection, and it's great to see it being used in your project. Using OpenCV the video is captured using a webcam and it gets compared with the model and the model will predict the gesture. Based on the recognized sign language gestures, we can make the robot respond accordingly. Next, the paper reviews the key components of SLR systems, such as hand gesture detection, hand shape recognition, finger spelling recognition, and facial expression analysis. It explores the methodologies employed in these systems, including computer vision, machine learning, and deep learning techniques. Furthermore, the challenges and limitations of existing SLR systems are discussed, including variability in signing styles, occlusions, lighting conditions, and the need for large annotated datasets. Promising avenues for addressing these challenges, such as multimodal fusion, transfer learning, and domain adaptation, are also explored. Finally, the paper outlines potential future directions for SLR research, including the development of real-time systems, improved robustness to environmental factors, and the integration of SLR with natural language processing for more effective communication. Additionally, the importance of interdisciplinary collaboration between computer science, linguistics, and Deaf studies is emphasized to advance the field of SLR.

LIST OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	i
	LIST OF FIGURES	v
	LIST OF ABBREVIATIONS	vi
1	INTRODUCTION	1
1.1	Human-Robot Interaction	1
1.1.1	Need For Sign Language	3
1.2	Background	3
1.3	Latest Approach For Sign Language	4
1.3.1	Data Acquisition	4
1.3.2	Data Pre-Processing	5
1.3.3	Feature Extraction	5
1.3.4	Gesture Classification	6
1.4	Artificial Neural Network	7
1.5	Convolutional Neural Network	7
1.5.1	TensorFlow	8
1.5.2	Keras	8
1.5.3	OpenCV	9

2	LITERATURE SURVEY	10
2.1	Overview of Existing Research and Literature	10
2.2	Inference from Literature Survey	13
2.3	Open Problems in Existing System	14
3	REQUIREMENT ANALYSIS	18
3.1	Feasibility Study	18
3.1.1	Technical Feasibility	18
3.1.2	Financial Feasibility	19
3.1.3	Operational Feasibility	19
3.1.4	Societal Feasibility	20
3.2	Risk Analysis	21
3.2.1	Technical Risks	21
3.2.2	Financial Risks	22
3.2.3	Operational Risks	23
3.2.4	Societal Risks	24
3.3	Software Requirements	25
3.3.1	Defining Requirements	26
3.3.2	Non-Functional Requirements	27
4	METHODOLOGY AND ALGORITHMS USED	28
4.1	Methodology	28
4.1.1	Data Collection	29
4.1.2	Pre-Processing	30
4.1.3	Model Training	31

	4.1.4	Model Testing and Performance Evaluation	35
	4.1.5	Output Accuracy	37
	4.2	System Architecture	38
5		RESULTS AND DISCUSSIONS	40
	5.1	Result	40
	5.2	Discussion	43
6		SUMMARY AND CONCLUSION	45
	6.1	Summary	45
	6.2	Conclusion	45
	6.3	Future Work	46
		REFERENCES	47
		APPENDIX	49
	A	Source Code	49
	B	Screenshots	52
	C	Research Paper	54
	D	Acceptance Letter	58

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
1.1	Components of Sign Language	2
1.2	Artificial Neural Networks	7
1.3	OpenCV	7
4.1	Data Collection for Hand Gesture	30
4.2	Confusion Matrix	37
4.3	Output Accuracy	38
4.4	System Architecture	39
5.1	Output Increase Volume	40
5.2	Output Fan ON	41
5.3	Output Wait Here	41
5.4	Output Charge Battery	42
5.5	Output Light ON	43

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
HRI	Human-Robot Interaction
SLR	Sign Language Recognition
LSTM	Long Term Short Memory
HMM	Hidden Markov Model
CRF	Conditional Random Fields
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
OpenCV	Open source Computer Vision library

CHAPTER 1

INTRODUCTION

1.1 HUMAN-ROBOT INTERACTION

Among various approaches to human-robot interaction, the way of using hand gestures for interaction facilitates more efficient communication between humans and robots. Since gesture interaction is a kind of non-contact way, which is more secure, friendly, and easy to accept by humankind. The gesture is one of the most widely used communicative manners. In the long-term social practice process, the gesture is endowed with a variety of specific meanings. At present, gestures have become the most powerful tool for expressing sentiment, intention, or attitude for humans. Sign language mainly uses the human hands to convey information. The main contribution of this work is an HRI system by integrates deaf and speech-impaired people, which enables the social robots to communicate with target people efficiently and friendly. Most importantly, the proposed system can be applied in other interaction scenarios between robots and autistic children.

Sign language is an essential tool to bridge the communication gap between normal and hearing-impaired people. However, the diversity of over 7000 present-day sign languages with variability in motion position, hand shape, and position of body parts makes automatic sign language recognition (ASLR) a complex system. To overcome such complexity, researchers are investigating better ways of developing ASLR systems to seek intelligent solutions and have demonstrated remarkable success. Each method has its strength compared to other methods and researchers are still using different methods in developing their own Sign Language Recognition. Each method also has its limitations compared to other methods. This research work aims to review the sign language recognition approaches and design the best training tool for visually challenged people.

Sign language recognition is the process of recognizing and interpreting human hand gestures, movements, and poses used in sign languages, which are used by deaf and hard-of-hearing individuals to communicate with each other and with the hearing community. Sign languages are complex visual languages that use a

combination of hand gestures, facial expressions, and body movements to convey meaning. They are not simply a visual representation of spoken languages but are distinct and independent languages with their grammar, syntax, and vocabulary. There are many different sign languages around the world, each with its regional variations and dialects.

Sign language is a visual language and consists of 3 major components

Fingerspelling	Word level sign vocabulary	Non-manual features
Used to spell words letter by letter .	Used for the majority of communication.	Facial expressions and tongue, mouth and body position.

Fig:1.1 Components of Sign Language

In 1.1 we see the three major components of sign language. With this, we focus on producing a model that can recognize word-level sign vocabulary-based hand gestures to give commands to devices that get commands and perform them.

Motivation: Inclusivity and Accessibility: The project aims to bridge the communication gap between robots and speech-impaired individuals. By leveraging hand gestures, which are a fundamental form of expression for this demographic, the system promotes inclusivity and allows them to interact with social robots more naturally and efficiently.

The motivation for a sign language recognition project lies in fostering inclusivity and accessibility for the deaf and hard-of-hearing community. By developing accurate systems, the project aims to bridge communication gaps, empower individuals, and promote equal participation in various aspects of life, from education to social interaction. Applicability in Special Education for Communication with Robots: The proposed system can have a significant impact in the field of special education, particularly with autistic children. Many autistic individuals may struggle with traditional forms of communication, and gestures provide a non-intrusive and familiar means of interaction. This system has the potential to facilitate meaningful engagement and learning experiences for these children.

1.1.1 *Need for Sign Language*

Sign language is indispensable for fostering communication, inclusion, and cultural identity among Deaf and hard-of-hearing individuals. It serves as a vital medium through which they can express themselves, share ideas, and connect with others on a profound level. Beyond mere communication, sign language represents a rich cultural heritage, providing Deaf communities with a sense of belonging and identity. Recognized as an official language in many countries, sign language ensures that Deaf individuals have equal access to education, employment, and essential services. Moreover, it promotes inclusivity by breaking down barriers between Deaf and hearing individuals, creating a more accessible and understanding society. As a dynamic and expressive form of communication, sign language enriches lives and fosters a more inclusive and diverse world for everyone.

1.2 BACKGROUND

The development of sign languages can be traced back to the early history of Deaf communities. Before the formalization of sign languages, Deaf individuals often communicated through informal gestures and signs within their communities. However, it wasn't until the 18th century that sign languages began to be recognized and studied more systematically. Another pivotal figure in the history of sign language was Thomas Hopkins Gallaudet, an American educator who travelled to Europe in the early 19th century to learn about methods of educating Deaf children. He met Laurent Clerc, a Deaf teacher, in France, and together they returned to the United States to establish the first permanent school for the Deaf in Hartford, Connecticut, in 1817. This school, now known as the American School for the Deaf, played a crucial role in the spread of American Sign Language (ASL) and the development of Deaf education in the United States. Throughout the 19th and 20th centuries, sign languages continued to evolve and diversify in different regions around the world. Researchers and linguists began to study sign languages as legitimate and complex languages in their own right, with their own grammar, syntax, and vocabulary. This recognition helped to elevate the status of sign languages and advocate for their use in education, government, and other domains. Today, sign languages are officially recognized in many countries, and efforts

are ongoing to promote their use and ensure equal access for Deaf individuals in all aspects of society. The development of technologies such as video relay services and captioning has also expanded communication options for Deaf individuals, further enhancing their ability to participate fully in the world around them.

1.3 Latest Approach for Sign Language

In recent years there has been tremendous research done on hand gesture recognition. With the help of a survey, we realized the basic steps in hand gesture recognition are

- Data acquisition
- Data pre-processing
- Feature extraction
- Gesture classification

1.3.1 Data Acquisition

The different approaches to acquiring data about hand gestures can be done in the following ways:

- Use of sensory device: It uses electromechanical devices to provide exact hand configuration and position. Different glove-based approaches can be used to extract information. But it is expensive and not user-friendly.
- Vision-based approach: In vision-based methods, a computer camera is the input device for observing the information of hands or fingers. The vision-based methods require only a camera, thus realizing a natural interaction between humans and computers without the use of any extra devices. These systems tend to complement biological vision by describing artificial vision systems that are implemented in software and/or hardware. The main challenge of vision-based hand detection is to cope with the large variability of the human hand's appearance due to a huge number of hand movements, to different skin-colour possibilities as well as to the variations in viewpoints, scales, and speed of the camera capturing the scene.

1.3.2 Data Pre-Processing

The approach for hand detection combines threshold-based colour detection with background subtraction. We can use the AdaBoost face detector to differentiate between faces and hands as both involve similar skin colour. We can also extract the necessary image which is to be trained by applying a filter called Gaussian blur. The filter can be easily applied using open computer vision also known as OpenCV.

For extracting the necessary image which is to be trained we can use instrumented gloves. This helps reduce computation time for pre-processing and can give us more concise and accurate data compared to applying filters on data received from video extraction. We tried doing the hand segmentation of an image using colour segmentation techniques but as mentioned in the research paper skin colour and tone are highly dependent on the lighting conditions which output we got for the segmentation we tried to do was not so great. Moreover, we have a huge number of symbols to be trained for our project many of which look similar to each other like the gesture for the symbol 'V' and digit '2', hence we decided that to produce better accuracies for our large number of symbols, rather than segmenting the hand out of a random background we keep the background of the hand a stable single colour so that we don't need to segment it based on skin colour. This would help us to get better results.

1.3.3 Feature Extraction

Feature extraction in sign language recognition involves identifying key elements such as hand shape, orientation, and movement trajectory from raw data. This process transforms complex input into manageable representations that capture relevant information for classification. Efficient feature extraction methods enhance the discriminative power of machine learning models, enabling accurate recognition of sign language gestures.

Ongoing research explores novel techniques, including deep learning approaches, to improve feature extraction robustness and adaptability across diverse signing styles and environments. This process transforms complex input into manageable representations that capture relevant information for classification.

1.3.4 Gesture classification

Hidden Markov Models are used for the classification of the gestures. This model deals with dynamic aspects of gestures. Gestures are extracted from a sequence of video images by tracking the skin-colour blobs corresponding to the hand into a body–face space centred on the face of the user. The goal is to recognize two classes of gestures: deictic and symbolic. The image is filtered using a fast-lookup indexing table. After filtering, skin colour pixels are gathered into blobs. Blobs are statistical objects based on the location (x, y) and the colourimetry (Y, U, V) of the skin colour pixels to determine homogeneous areas.

- Naïve Bayes Classifier is used which is an effective and fast method for static hand gesture recognition. It is based on classifying the different gestures according to geometric-based invariants which are obtained from image data after segmentation. Thus, unlike many other recognition methods, this method is not dependent on skin colour. The gestures are extracted from each frame of the video, with a static background. The first step is to segment and label the objects of interest and to extract geometric invariants from them. The next step is the classification of gestures by using a K-Nearest Neighbour algorithm aided with a distance weighting algorithm (KNNDW) to provide suitable data for a locally weighted Naïve Bayes" classifier.
- According to a paper on "Human Hand Gesture Recognition Using a Convolution Neural Network" by Hsien-I Lin, Ming-Hsiang Hsu, and Wei-Kai Chen graduates of the Institute of Automation Technology National Taipei University of Technology Taipei, Taiwan, they construct a skin model to extract the hand out of an image and then apply binary threshold to the whole image. After obtaining the threshold image they calibrate it about the principal axis to centre the image about it. They input this image into a convolutional neural network model to train and predict the outputs. They have trained their model over 7 hand gestures and using their model they produce an accuracy of around 95% for those 7 gestures.

1.4 ARTIFICIAL NEURAL NETWORK

An Artificial Neural Network is a connection of neurons that replicates the structure of the human brain. Each connection of a neuron transfers information to another neuron. Inputs are fed into the first layer of neurons which processes it and transfers to another layer of neurons called hidden layers. After processing information through multiple layers of hidden layers, information is passed to the final output layer.

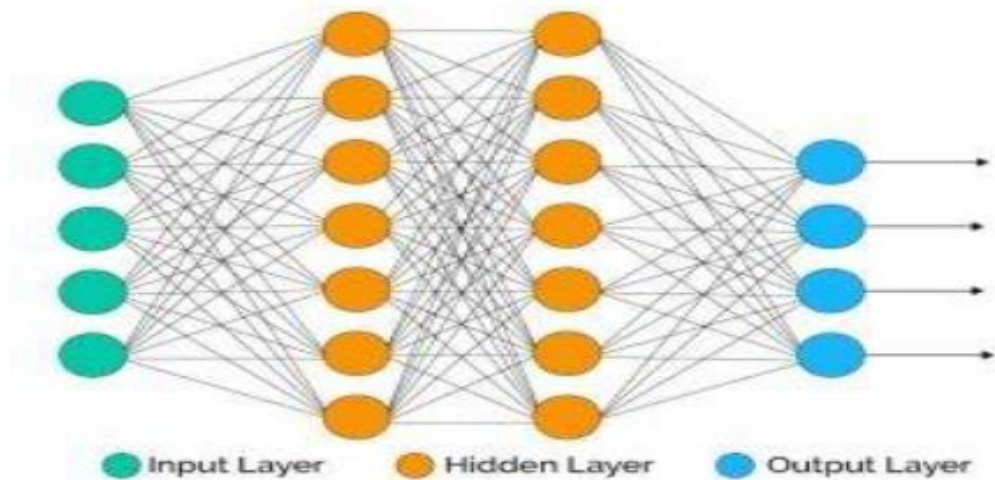


Fig:1.2 Artificial Neural Networks

A layer can have only a dozen units or millions of units as this depends on how the complex neural networks will be required to learn the hidden patterns in the dataset. Figure 1.2 shows the three different layers in Artificial Neural Network which are the input layer, output layer and hidden layers.

1.5 CONVOLUTION NEURAL NETWORK

Unlike regular Neural Networks, in the layers of CNN, the neurons are arranged in 3 dimensions: width, height, and depth. The neurons in a layer will only be connected to a small region of the layer before it, instead of all of the neurons in a fully-connected manner. Moreover, the final output layer would have dimensions, because by the end of the CNN architecture, we will reduce the full image into a single vector of class score. The architecture of CNNs is inspired by the visual processing in the human brain, and they are well-suited for capturing hierarchical patterns and spatial dependencies within images.

1.5.1 TensorFlow

TensorFlow is an open-source software library for numerical computation. First, we define the nodes of the computation graph, and then inside a session, the actual computation takes place. TensorFlow is widely used in Machine Learning. TensorFlow is a widely used open-source machine learning framework developed by Google.

TensorFlow provides a comprehensive ecosystem with tools for data preprocessing, model building, training, and deployment across various platforms. Its scalable architecture enables efficient distributed computing, making it suitable for both small-scale experiments and large-scale production systems. With extensive documentation, community support, and integration with other libraries, researchers and developers to tackle diverse challenges in AI.

1.5.2 Keras

Keras is a high-level neural network library written in Python that works as a wrapper for TensorFlow. It is used in cases where we want to quickly build and test the neural network with minimal lines of code. It contains implementations of commonly used neural network elements like layers, objectives, activation functions, optimizers, and tools to make working with images and text data easier. Keras is a high-level neural networks API written in Python, designed for fast experimentation and prototyping of deep learning models. It provides a user-friendly interface that simplifies the process of building, training, and deploying neural networks.

Keras offers a modular approach to constructing models, allowing developers to easily combine different layers, loss functions, and optimizers. With its focus on simplicity and ease of use, Keras abstracts away the complexities of low-level programming, enabling rapid iteration and experimentation. As an integral part of the TensorFlow ecosystem, Keras seamlessly integrates with the TensorFlow backend, providing a powerful and flexible tool for both beginners and experienced practitioners in deep learning. Keras covers every step of the machine learning workflow, from data processing to hyperparameter tuning to deployment. It was developed with a focus on enabling fast experimentation.

1.5.3 OpenCV

OpenCV (Open Source Computer Vision) is an open-source library of programming functions used for real-time computer vision. OpenCV is a versatile open-source library designed for computer vision and image processing tasks. It offers a comprehensive collection of functions and algorithms for analyzing, manipulating, and understanding visual data. It is mainly used for image processing, video capture and analysis for features like face and object recognition. It is written in C++ which is its primary interface. OpenCV supports various programming languages, including C++, Python, and Java, making it accessible to a wide range of developers. Its vast array of features includes image filtering, object detection, feature extraction, and camera calibration, among others. With its extensive documentation, active community, and cross-platform compatibility, OpenCV is a valuable tool for applications ranging from robotics and augmented reality to medical imaging and autonomous vehicles.

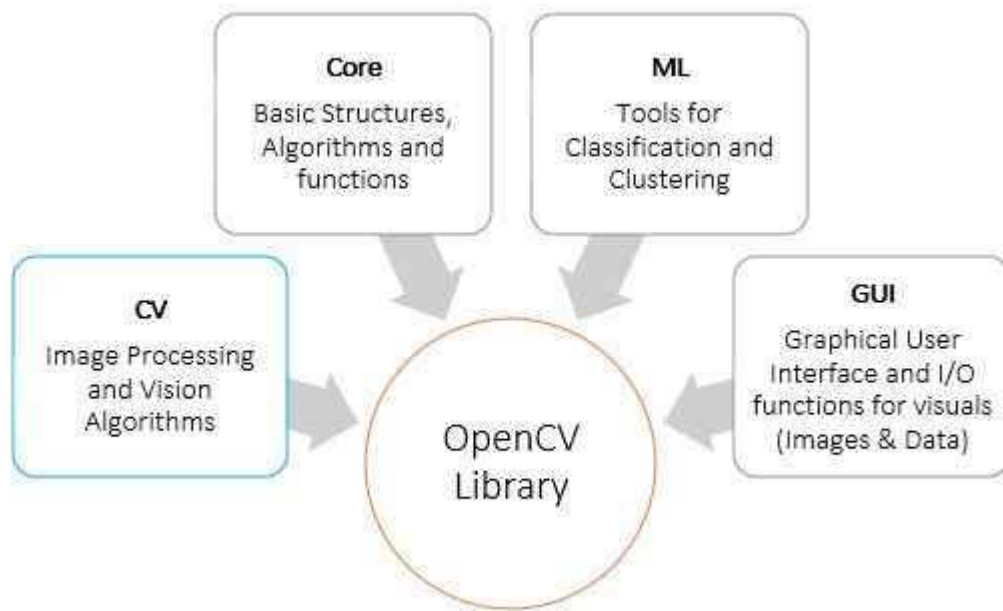


Fig:1.3 OpenCV

When it is integrated with various libraries, such as NumPy, Python is capable of processing the OpenCV array structure for analysis. Figure 1.3 demonstrates how OpenCV plays a trivial role in ML to identify an image pattern and its various features we use vector space and perform mathematical operations on these features.

CHAPTER 2

LITERATURE SURVEY

2.1 OVERVIEW OF EXISTING RESEARCH AND LITERATURE

The authors proposed a hand gesture recognition system based on deep convolutional neural networks. They trained their model on the NVIDIA DIGITS platform and evaluated it on a custom dataset of hand gestures. The authors found that their system achieved high accuracy in recognizing hand gestures, with the potential for application in human-computer interaction and virtual reality. Their system achieved high accuracy and low latency, demonstrating its potential for application in real-time human-computer interaction.

The authors provided a comprehensive review of the literature on gesture recognition using inertial sensors and machine learning algorithms, highlighting the various methods and applications in this field. The authors concluded that gesture recognition using inertial sensors and machine learning algorithms has shown significant promise in a range of applications, including rehabilitation, gaming, and human-robot interaction. The system achieved high accuracy in recognizing hand gestures, with the potential for application in various domains such as healthcare and robotics.

The authors proposed a multimodal framework for hand gesture recognition based on deep learning and ensemble learning. They evaluated their system on a custom dataset of hand gestures. The authors found that their system achieved high accuracy in recognizing hand gestures, outperforming other state-of-the-art methods. They concluded that their multimodal framework has the potential to improve the robustness and accuracy of hand gesture recognition systems.

This research work provides an overview of different deep-learning techniques used for gesture-to-text conversion. The authors discuss various CNN, RNN, and LSTM architectures used for gesture recognition tasks, and compare their performance on different datasets. This paper proposes a gesture recognition system based on CNNs for sign language recognition. The authors evaluate their system on the American Sign Language dataset and achieve high accuracy.

This paper presents a real-time gesture recognition system based on CNNs and edge computing. The authors evaluate their system on a custom dataset and achieve high accuracy and low latency. This paper proposes a vision-based gesture recognition system using CNNs and Dynamic Time Warping. The authors evaluate their system on a custom dataset and achieve high accuracy.

This paper provides a comprehensive survey of recent advances and challenges in deep learning-based gesture recognition. The authors cover various CNN, RNN, LSTM, and Transformer-based models used for gesture recognition and discuss the challenges and opportunities in this field. This paper presents a hand gesture recognition system based on CNNs and Support Vector Machines (SVMs). The authors evaluate their system on a custom dataset and achieve high accuracy. A real-time continuous sign language recognition system based on CNNs and Transformers. The authors evaluate their system on two benchmark datasets and achieve high accuracy and low latency.

The people in the world who are hearing impaired face many obstacles in communication and require an interpreter to comprehend what a person is saying. There has been constant scientific research and the existing models cannot make accurate predictions. So we propose a deep learning model trained on the ASL i.e. American Sign Language which will take action in the form of American Sign Language as input and translate it into text. To achieve the former a Convolution Neural Network based VGG16 architecture is used as well as a TensorFlow model for image classification and we have improved the accuracy of the latter by over 4%. The dataset which has been used is the ASL dataset which has over 87000 images and has been used to train and test the video. There has been an improvement in accuracy from 94% of CNN to 98.7% by Transfer Learning. An application with the deep learning model integrated has also been built.

Auditory impairment or hearing loss is a major problem in today's human population. Sign language helps hearing-impaired people to live their social life without much difficulty. The latest technology used in detecting sign language connects them to the rest of the world. Sign language recognition and conversion to the text based on the movement of hands and the shape formed by the fingers is a complex system. The

solutions using machine learning give significant success for this complex system. This paper mainly focuses on developing a system for recognizing the different hand signs in American Sign Language and their emotions simultaneously in real-time and converting them into text. The system resolves the need for a translator to bridge the gap between a sign language based user and a non-sign language-based user. Most state-of-the-art technology involves CNN models with image pixels by identifying specific key point coordinates on the face/hand obtained. Using the latest technologies like MediaPipe, an improved CNN model is developed based on the distances between each unique identified vital point. The customized model able to achieve 80 percentage of accuracy for the live image.

Sign language is utilized by deaf individuals to communicate with each other. Sign languages, simply put are signs and gestures made by hand and are currently being used by approximately 80 million people around the world. The limited knowledge of sign languages in the hearing-abled community makes it difficult for them to interact with the deaf or mute community leading to the marginalization of the latter. Sign-to-text classification is a field of research that aims to convert sign language into written text, providing a communication bridge between the deaf-mute communities and the hearing population. However, there are some limitations in the current state of the field, such as the absence of datasets that are publicly available for sign language in many languages, and most of the existing models are trained in American Sign Language (ASL) which makes it difficult for researchers to train and evaluate their methods on other sign languages. The proposed model makes use of one-shot learning, a new-age machine learning technique that allows the model to learn and recognize using limited training data, to identify and classify hand signs and convert them into corresponding text. The ability to learn from a very small dataset makes the model easily trainable and expandable in any sign language of choice. The proposed model has been trained on ASL and Indian Sign Language (ISL) datasets and achieved accuracy of 90.03% and 91.25% respectively. Using the latest technologies like MediaPipe, an improved CNN model is developed based on the distances between each unique identified vital point. The customized model able to achieve 80 percentage of accuracy for the live image.

2.2 INFERENCE FROM LITERATURE SURVEY

From a literature survey on sign language recognition, several key insights and inferences can be drawn:

- 1 Modalities: Sign language recognition can involve multiple modalities such as video, depth, and wearable sensors. Researchers have explored different modalities and their combinations to improve recognition accuracy and address challenges like viewpoint variations and occlusion
- 2 Technological Advances: There has been significant progress in sign language recognition due to advancements in computer vision, machine learning, and deep learning techniques. These advances have enabled more accurate and robust recognition systems.
- 3 Data Availability: The availability of large-scale sign language datasets has facilitated the development and evaluation of various recognition algorithms. Datasets like RWTH-PHOENIX-Weather 2014T and American Sign Language (ASL) datasets have been widely used for training and testing purposes.
- 4 Feature Representation: Feature representation plays a crucial role in sign language recognition. Deep learning-based approaches, especially Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), have shown superior performance by automatically learning discriminative features from raw data.
- 5 Gesture Segmentation and Recognition: Gesture segmentation is the process of dividing continuous sign language sequences into individual units for recognition. Various techniques including Hidden Markov Models (HMM), Conditional Random Fields (CRF), and deep learning-based methods have been employed for gesture segmentation and recognition.
- 6 Challenges and Limitations: Despite advancements, sign language recognition still faces several challenges such as dynamic background, complex hand movements, non-manual components (facial expressions, body postures), and inter-sign interference. Additionally, achieving real-time performance and handling signer-independent recognition remain challenging tasks.

- 7 Application Domains: Sign language recognition has applications in diverse domains including assistive technology, education, human-computer interaction, and accessibility services. Researchers have explored these applications and developed tailored solutions to address specific requirements and user needs.
- 8 Evaluation Metrics: Evaluation of sign language recognition systems typically involves metrics such as accuracy, precision, recall, F1-score, and recognition speed. It's important to consider these metrics comprehensively to assess the overall performance of the system under different conditions.

By synthesizing findings from existing literature, researchers can gain valuable insights into the current state-of-the-art techniques, identify research gaps, and propose novel approaches to advance the field of sign language recognition.

2.3 OPEN PROBLEMS IN THE EXISTING SYSTEM

Existing systems for sign language recognition still face several challenges and limitations, which can hinder their effectiveness and real-world applicability. Some of the prominent problems include:

- **Data Imbalance:** In sign language datasets, certain signs may have significantly fewer samples compared to others, leading to data imbalance issues. Models trained on imbalanced datasets may exhibit biases towards overrepresented signs and perform poorly on underrepresented ones.
- **Contextual Understanding:** Sign language communication involves not only individual signs but also the context in which they are used, including grammar, syntax, and discourse structure. Integrating contextual information into recognition systems to improve understanding and disambiguation of signs within larger linguistic contexts remains an open problem.
- **Complexity of Hand Gestures:** Sign languages involve complex hand gestures and movements, which can be challenging to accurately capture and recognize. Variations in hand shapes, orientations, and motion dynamics further complicate the recognition process, especially in the presence of occlusions and noise.

- **Non-Manual Components:** Sign language communication involves not only hand gestures but also non-manual components such as facial expressions, body postures, and head movements. Integrating these non-manual components into recognition systems poses additional challenges but is crucial for achieving accurate interpretation of sign language.
- **Viewpoint Variations and Occlusions:** Variations in camera viewpoints and occlusions caused by hands or other objects can significantly affect the performance of sign language recognition systems. Models need to be robust to these variations and capable of handling partial visibility of signs.
- **Real-Time Performance:** Many existing sign language recognition systems may not meet the real-time processing requirements, especially when deployed in interactive applications such as sign language interpretation devices or communication aids. Achieving low latency and high-speed processing remains a challenge.
- **Signer Independence:** Most sign language recognition systems are trained on data from a specific group of signers, resulting in limited generalization to unseen signers. Building signer-independent models that can accurately recognize signs performed by new users remains an ongoing research challenge.
- **Cultural and Linguistic Diversity:** Sign languages exhibit significant cultural and linguistic diversity across different regions and communities. Existing recognition systems may not adequately capture this diversity, leading to limited applicability in multicultural settings. Adapting models to different sign languages and cultural contexts requires careful consideration of linguistic variations and cultural nuances, which poses a considerable challenge.
- **Ethical Considerations and User Privacy:** Deploying sign language recognition systems in real-world scenarios raises ethical concerns regarding user privacy, consent, and potential biases. Collecting and processing sign language data involves sensitive information about individuals' communication and identity. Ensuring privacy protection, obtaining informed consent, and mitigating biases in data collection and model development are essential for the responsible

deployment of sign language recognition technology. Balancing technological advancements with ethical considerations is crucial for building trust and fostering inclusive access to communication services for diverse communities.

- **Scalability and Generalization Across Languages:** Many existing sign language recognition systems are developed and evaluated on specific sign languages or datasets, limiting their scalability and generalization to other languages or cultural contexts. Developing techniques that can transfer knowledge across languages, adapt to new linguistic structures, and generalize across diverse signing communities is an open problem with implications for global accessibility and inclusivity.
- **Ethical and Societal Implications:** Deploying sign language recognition technology raises ethical and societal considerations related to privacy, consent, bias, and accessibility. Addressing these concerns and ensuring responsible development and deployment of sign language recognition systems in diverse social and cultural contexts is an ongoing challenge that requires interdisciplinary collaboration and stakeholder engagement.
- **Interpretability and Transparency:** Deep learning-based sign language recognition models often lack interpretability and transparency, making it difficult to understand their decision-making processes and potential biases. Developing methods for explaining model predictions, identifying sources of errors, and ensuring transparency in the decision-making process is an open problem that is critical for building trust and accountability in sign language recognition technology.

Addressing these challenges requires interdisciplinary research efforts involving computer vision, machine learning, signal processing, and linguistics. Developing more comprehensive datasets, exploring novel feature representations, improving model architectures, and integrating contextual information are among the strategies researchers are pursuing to enhance the robustness and accuracy of sign language recognition systems.

Sign language recognition research has seen significant advancements in recent years, driven by the need to bridge communication barriers for the Deaf and

hard-of-hearing community. Early works focused on hand gesture detection and recognition using computer vision techniques. Subsequent studies explored deep learning methods to improve accuracy and robustness in SLR systems. Multimodal approaches combining hand gestures, facial expressions, and body movements have gained traction for more comprehensive recognition. Challenges such as variability in signing styles, occlusions, and lighting conditions continue to drive research towards more robust and adaptive SLR solutions.

Recent literature in sign language recognition delves into multimodal fusion techniques, integrating hand gestures, facial expressions, and body movements for enhanced accuracy and natural interaction. Moreover, there's a growing focus on real-time SLR systems, addressing challenges like environmental factors and user variability to advance accessibility for the Deaf community. Addressing these open problems requires interdisciplinary research efforts spanning computer vision, machine learning, linguistics, human-computer interaction, and ethics, with a focus on developing holistic solutions that consider the diversity, complexity, and societal impact of sign language communication.

After conducting a comprehensive literature survey on sign language recognition, it is evident that significant progress has been made in the field, driven by advancements in computer vision, machine learning, and deep learning techniques. However, several challenges and open problems persist, including variability in signing styles, limited signer adaptation, contextual understanding, and real-time performance, scalability across languages, ethical considerations, and interpretability of models. Moving forward, future research efforts should focus on developing holistic solutions that consider the complexity and diversity of sign languages, while also addressing ethical and societal implications. By leveraging advances in technology and embracing inclusivity, sign language recognition can play a transformative role in enhancing communication accessibility and promoting equality for individuals with hearing impairments.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

A feasibility study for Sign Language Recognition involves assessing the viability and practicality of developing and implementing a system for recognizing and interpreting sign language gestures. Sign Language Recognition systems hold immense potential for enhancing communication and accessibility for the Deaf and hard-of-hearing community. However, before implementation, it is essential to conduct feasibility studies and risk analysis to assess the viability, potential challenges, and associated risks of developing and deploying SLR systems. This paper presents a comprehensive examination of feasibility and risk analysis specific to SLR systems, encompassing technical, financial, operational, and societal factors.

3.1.1 *Technical Feasibility*

It evaluates the technical capabilities and limitations of SLR systems, including hardware requirements, software algorithms, and data processing capabilities. This involves assessing the availability of suitable sensors, the performance of recognition algorithms, and the feasibility of achieving real-time processing and accuracy in diverse environments. Sign language recognition is technically possible. Existing techs like motion tracking and deep learning can build recognition models. The challenge is making them work in real-world situations with all their variations. Also, collecting large training datasets is expensive. Despite these hurdles, research is ongoing and accuracy is improving. Sign language recognition goes beyond hands. Facial expressions and body language also play a role, adding complexity for the models to capture the full meaning. These recognition systems have the potential to bridge the communication gap between deaf and hearing communities, making them valuable tools for social inclusion.

3.1.2 Financial Feasibility

It assesses the cost implications of developing, deploying, and maintaining SLR systems. This includes considerations for hardware procurement, software development, data collection, training, and ongoing maintenance. Cost-benefit analysis and return on investment calculations help determine the financial viability and potential economic benefits of SLR implementation.

Building sign language recognition is expensive. It takes money to gather data, train models, and create the technology. There's also the cost of putting SLR to use, like hardware and software. SLR can make communication easier for deaf communities, leading to more participation and potentially saving costs on interpreters. A growing market for SLR could also make it financially attractive for businesses to invest. So, it's a balance between cost and the potential benefits.

3.1.3 Operational Feasibility

It evaluates the practicality and effectiveness of integrating SLR systems into existing workflows and environments. This involves assessing compatibility with existing technologies, user acceptance, training requirements, and scalability considerations. Signs themselves can be complex. Signing styles, speeds, and even small details can change their meaning. SLR systems need to be smart enough to handle these variations.

- **The real world throws curveballs:** Imagine a sunny day vs a dimly lit room. Background noise or a shaky camera can also confuse the system.
- **Not all devices are created equal:** Running complex recognition models might be too much for some phones or tablets.
- **Making models slimmer:** Imagine a fitness tracker vs a powerful gaming PC. Researchers are building "lightweight" models that use less power and work on mobile devices. Running powerful recognition models requires significant processing power, which might be limited on some devices.

- **Real-time conversations.** Just like a speedy internet connection, faster processing allows for smoother communication without delays. Advancements in AI are making this a reality.
- **Fitting in with the crew.** Imagine SLR working with your favourite video conferencing app. Efforts are underway to integrate SLR with existing tools for a more seamless experience.

3.1.4 Societal Feasibility

It examines the broader societal implications and ethical considerations of SLR implementation. This includes considerations for privacy, data security, cultural sensitivity, and accessibility. Stakeholder engagement and consultation with the Deaf community are essential to ensure that SLR systems respect and uphold the rights and dignity of users. The societal feasibility of sign language recognition looks at how well society accepts and integrates this technology.

- **Inclusion:** SLR can bridge the communication gap between deaf and hearing communities, fostering social inclusion and participation in various aspects of life.
- **Cultural Acceptance:** Widespread adoption of SLR can raise awareness and appreciation for sign languages, promoting cultural understanding and respect.
- **Standardization:** Sign languages have regional variations. SLR systems need to be adaptable or cater to specific regions to ensure effectiveness.
- **Privacy Concerns:** There might be concerns about how SLR data is collected, stored, and used, requiring transparent practices and user trust.
- **Ethical Considerations:** As SLR becomes more prevalent, ethical considerations regarding the potential job displacement of sign language interpreters and ensuring equitable access to the technology for all socioeconomic groups need to be addressed.

Overall, SLR has the potential to significantly improve social inclusion for deaf communities. However, addressing standardization issues and building public trust is crucial for its successful societal integration

3.2 RISK ANALYSIS

It identifies potential risks and uncertainties associated with SLR development and deployment. This includes technical risks such as algorithmic errors, hardware failures, and data biases, as well as financial risks such as budget overruns and insufficient funding. Operational risks such as implementation challenges and user resistance, and societal risks such as privacy breaches and cultural insensitivity, must also be considered. Risk analysis for sign language recognition systems involves identifying potential threats, assessing their likelihood and impact, and implementing strategies to mitigate or manage these risks.

3.2.1 Technical Risks

Inaccurate recognition of hand gestures or facial expressions due to limitations in the recognition algorithms. Malfunctioning sensors or equipment leads to data loss or system downtime. Biases in training data lead to disparities in recognition accuracy across different demographics or sign languages. Variations in lighting conditions, background clutter, or occlusions affect the performance of SLR systems. Technical risks in sign language recognition stem from limitations in current technology and challenges in building robust systems.

- **Accuracy:** SLR models might struggle with recognizing all signs accurately, especially in complex situations. Variations in signing styles, speeds, and subtle expressions can lead to misinterpretations. Additionally, models trained on one dataset might not perform well with regional variations in sign languages.
- **Generalizability:** SLR models might not perfectly recognize all signs, especially in complex situations. Imagine someone signing quickly or with subtle variations in hand position. These nuances can confuse the model and lead to misinterpretations. A model trained on data from one region or signing style

might not perform well with another. This lack of generalizability limits the usefulness of SLR in diverse settings.

- **Environmental Dependence:** Current SLR systems can be sensitive to environmental factors like lighting, background noise, and camera angles. These variations can introduce errors and limit the reliability of SLR in real-world settings. Imagine a brightly lit room versus a dimly lit stage. Changes in lighting can confuse the SLR system, making it difficult to accurately track hand movements and gestures. A noisy classroom or a bustling street can introduce errors. Background noise can interfere with the system's ability to isolate the signer and their movements.

3.2.2 Financial Risks

The financial risks of sign language recognition stem from the costs associated with development, deployment, and potential drawbacks that can outweigh the benefits. Underestimation of development, deployment, or maintenance costs leading to financial strain. Inadequate resources allocated for SLR projects lead to project delays or termination. Failure to realize expected economic benefits or revenue generation from SLR implementation.

- **High Development Costs:** Building accurate SLR models requires significant investment in several areas:
 1. **Data Collection:** Gathering a large and diverse dataset of sign language videos can be expensive and time-consuming.
 2. **Engineering Expertise:** Developing and training complex machine learning models necessitates highly skilled professionals.
 3. **Computational Resources:** Training powerful SLR models requires significant computing power, which can incur costs associated with cloud services or maintaining in-house infrastructure.

- **Uncertain Return on Investment:** Even if a functional SLR system is developed, the benefits might not justify the initial investment. Here's why:
 1. Limited Market Adoption: SLR technology might not be widely adopted if the target audience lacks the resources or access to necessary equipment.
 2. Competition from Existing Solutions: Established solutions like sign language interpretation services might be cheaper or preferred by some users.
- **Focus on Specific Applications:** Developing SLR solutions for targeted applications can reduce development costs and improve ROI. Open-sourcing SLR projects can leverage community contributions and expertise, reducing development costs. Government support through grants or tax breaks can encourage companies to invest in SLR development for the social good it brings.

3.2.3 Operational Risks

Difficulties in integrating SLR systems into existing workflows or environments. Lack of user acceptance or adoption due to usability issues, training requirements, or perceived inefficacy. Inability to scale SLR systems to accommodate growing user demands or expanding use cases. Operational risks in sign language recognition focus on the challenges of running SLR systems effectively in real-world settings.

- **System Complexity:** SLR systems involve complex interactions between software, hardware, and the environment.
 1. Software: SLR models need to be accurate and efficient enough to run smoothly on devices with varying processing power.
 2. Hardware: Cameras, depth sensors, and other equipment need to function properly and capture clear sign language data.
 3. Environment: As discussed earlier, factors like lighting, background noise, and camera angles can significantly impact performance.

- **User Experience:** Even with accurate recognition, a poorly designed SLR system can be frustrating to use.
 1. Limited User Interface: Complex interfaces or lack of clear feedback can make it difficult for users to interact with the SLR system effectively.
 2. Latency Issues: Delays in processing and translating signs can disrupt conversations and make communication unnatural.
 3. Limited Reliability: If the system frequently malfunctions or produces errors, users might lose trust and abandon it altogether.
 4. Lightweight Model Design: Developing efficient models that require less processing power allows SLR to run on mobile devices or embedded systems, increasing accessibility.
 5. Real-time Processing Techniques: Advancements in AI are enabling faster processing for smoother communication with minimal latency.
 6. User-Centred Design: Focusing on user needs by creating intuitive interfaces, providing clear feedback, and ensuring system reliability can improve the overall user experience.

3.2.4 Societal Risks

Unauthorized access or misuse of user data collected by SLR systems. Insufficient consideration of cultural norms and diversity within the Deaf community leads to alienation or exclusion. Failure to address the accessibility needs of users with diverse communication preferences or disabilities. Societal risks of sign language recognition deal with potential negative consequences on individuals and communities as this technology integrates into society.

- **Language Variation:** Sign languages have dialects and regional variations, similar to spoken languages. Over-reliance on SLR systems designed for a specific standard might discourage the use of these natural variations, leading to a loss of linguistic richness.

- **Cultural Identity:** Sign language is a core component of deaf culture. Widespread use of SLR could unintentionally prioritize spoken language and downplay the value of sign language itself.
- **Over-reliance on Technology:** SLR should be a tool to empower deaf individuals, not a replacement for natural sign language communication. Over-dependence on technology could lead to communication barriers if SLR systems malfunction or become unavailable.
- **Accessibility Gap:** Not everyone might have access to the devices or internet connection needed to use SLR. This could exacerbate the digital divide and exclude certain segments of the deaf community.
- **Promoting Diversity-Aware Systems:** Developing SLR systems that can handle variations in sign language and integrate feedback from diverse deaf communities can help preserve linguistic richness and cultural identity.
- **Focus on Accessibility:** Ensuring affordability and user-friendliness of SLR technology, along with promoting digital literacy within deaf communities, can bridge the access gap and ensure equitable use.
- **Transparency and Education:** Open communication about the limitations and potential risks of SLR alongside its benefits can foster informed adoption and responsible use within society.

3.3 SOFTWARE REQUIREMENTS

The purpose of this document is to outline the software requirements for the development of a Sign Language Recognition system. This document serves as a guide for the development team, stakeholders, and users to understand the functionality, constraints, and specifications of the SLR system. The software requirements for Sign Language Recognition encompass a diverse range of capabilities aimed at facilitating the accurate detection, interpretation, and translation of sign language gestures into meaningful digital representations. These requirements include the development of robust algorithms for hand gesture recognition, motion

tracking, and feature extraction, capable of capturing the intricate nuances and variations inherent in sign language communication. Additionally, the system must incorporate techniques for context awareness, enabling the interpretation of signs within the broader linguistic context and considering factors such as grammar, syntax, and non-manual components. Furthermore, the software should support real-time processing, ensuring low-latency interaction and responsiveness for users. Accessibility features, such as user-friendly interfaces, customizable settings, and compatibility with assistive technologies, are also essential to ensure inclusivity and usability for individuals with hearing impairments. Compliance with data privacy regulations and standards, as well as interoperability with existing communication platforms and devices, further enhance the software's utility and adoption potential. Overall, the software requirements for SLR aim to deliver a comprehensive and effective solution that empowers individuals with hearing impairments to communicate seamlessly and inclusively in a digital world.

3.3.1 Defining Requirements

Defining requirements is the crucial first step in any software development project, including sign language recognition (SLR). It's the process of gathering, understanding, refining, prioritizing, and documenting everything a successful SLR system needs. This ensures the final product meets the needs of both users and stakeholders. The Sign Language Recognition system requires robust functionality to effectively interpret sign language gestures. It must include user authentication for secure access. Gesture detection capabilities are essential, enabling real-time tracking of hand movements. Hand shape recognition is necessary to classify varied hand positions accurately. The system should also recognize finger spelling gestures, and interpret alphanumeric characters and words. Facial expression analysis enhances gesture interpretation, improving communication accuracy. Translation and output functionality are crucial, in translating recognized gestures into text or speech for non-signers.

3.3.2 Non-functional requirements

It focuses on performance, ensuring high accuracy and real-time processing. Reliability is vital, ensuring the system operates effectively in diverse conditions. Usability considerations emphasise an intuitive interface catering to users with diverse abilities. Scalability ensures the system can accommodate growing user needs. Security measures are necessary to protect user data and ensure privacy compliance. Hardware interfaces with sensors or cameras are essential, along with software integration for image processing and machine learning. Constraints include hardware limitations and the availability of diverse training data. The document serves as a comprehensive guide, addressing key requirements essential for developing a functional and user-friendly SLR system. Beyond what the sign language recognition system does, there's how well it does it. That's where non-functional requirements come in. These NFRs focus on the performance and user experience. Imagine a conversation with a delay that's bad latency. An NFR would aim for minimal delay (under 200 milliseconds) for smooth communication. Another NFR would be accuracy the system should recognize signs correctly, typically above 95% to avoid misunderstandings. Reliability is key too. Frequent crashes or errors would render the SLR system frustrating to use. Security is another NFR, making sure user data privacy is protected. Usability is an NFR that considers both deaf and hearing users. An intuitive interface and clear feedback are essential for widespread adoption. Performance also matters - the system shouldn't slow down the device it's running on.

These are just a few NFRs, and the specific ones will depend on how the SLR system is used. For instance, real-time conversations might prioritize low delay, while education might focus on accuracy and integration with learning platforms. By carefully considering NFRs, developers can create SLR systems that are not only functional but also meet the needs of users in real-world settings.

CHAPTER 4

METHODOLOGY AND ALGORITHMS USED

4.1 METHODOLOGY

Sign language recognition begins with data acquisition, capturing gestures via video. Preprocessing enhances data quality, followed by feature extraction identifying key elements like hand shape and movement. Machine learning algorithms like Convolutional Neural Networks classify gestures based on extracted features. Training data, annotated by experts, is crucial for model development. They're combining these approaches, using even more powerful models, and making the whole process faster for smoother communication. By refining these methods and exploring new ones, SLR has the potential to become a game-changer for accessibility and social inclusion. This method, called computer vision-based SLR, extracts key features from the video, like hand position and movement. Then, powerful machine learning models, like the ones used in image recognition, analyze these features to figure out the signs being used. Finally, the system translates the signs into text, or spoken language, or even shows them visually. Evaluation metrics like accuracy assess recognition performance. Deployment integrates the trained model into software or hardware systems for real-time interpretation, aiding in education, assistive devices, and human-computer interaction. This method uses special gloves equipped with sensors. These sensors track your hand movements and finger bends, capturing data that are then fed into machine-learning models. Similar to the camera method, the models are trained on a bunch of data to recognize the patterns in these movements and translate them into signs. By refining these methods and exploring new ones, SLR has the potential to become a game-changer for accessibility and social inclusion. This method, is called computer vision-based SLR. Ongoing research focuses on improving

- Data Collection
- Preprocessing
- Model Training
- Model Testing & Performance Evaluation
- Output Accuracy

4.1.1 Data Collection

Data is king for sign language recognition (SLR) systems. Just like training a puppy, SLR needs a lot of examples to learn and recognize signs accurately. MediaPipe is an open-source framework developed by Google that provides pre-built components for building machine learning pipelines for various tasks such as pose detection, face detection, hand tracking, and object detection, among others. It offers ready-to-use machine learning models along with efficient data processing pipelines, making it easier for developers to implement real-time perception algorithms in their applications. Studios offer a controlled environment with perfect lighting and backgrounds, but real-world recordings are important too. Imagine a system that's never seen someone signing in a dimly lit room! Crowdsourcing online can also help gather data from a wider range of signers and languages. Even with limited data, researchers have tricks up their sleeves. They can create artificial sign language videos using computers or modify existing data (like adding noise or changing lighting) to create more variations and teach the system to be more robust. By planning data collection carefully, respecting privacy, and using these data tricks, researchers can gather the best info to build powerful and inclusive SLR systems that work well in the real world. With its flexibility and ease of integration, the following commands have been implemented.

- Play music
- Light on
- Light off
- Increase volume
- Move right
- Move left
- Move forward
- Move backward
- Wait here

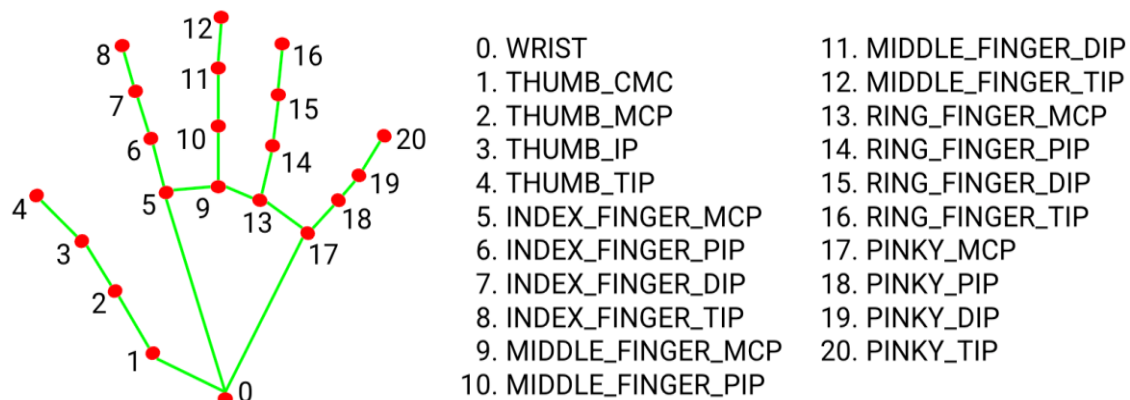


Fig:4.1 Data collection for Hand gestures

As shown in Figure 4.1 the moment made by hand will be collected by MediaPipe which will read the position of each point and understand what gesture the user is making.

4.1.2 Pre-Processing

Before sign language recognition systems can work their magic, the raw data needs some cleaning and prep work. This pre-processing stage is crucial for ensuring the accuracy and efficiency of the recognition process. Pre-processing in sign language recognition involves noise reduction, normalization of hand shapes and movements, and possibly background subtraction. These steps ensure cleaner input data for feature extraction and improve the accuracy of machine learning models. By incorporating these pre-processing SLR systems can effectively enhance the quality of input data, improve feature representation, and ultimately enhance the accuracy and reliability of sign language recognition.

Pre-processing often involves extracting key features from the data that are most relevant for sign recognition. These features could be hand shape, fingertip locations, or motion trajectories. By focusing on these informative features, the system can learn the essential aspects of signs without getting bogged down by irrelevant details.

- Checking the units of data
- Assign dependent variable and independent variable
- Splitting the data into training data and testing data
- Checking for missing values

Benefits of Pre-Processing

- **Improved Accuracy:** Effective pre-processing reduces noise and distractions in the data, leading to more accurate recognition of signs by the models.
- **Reduced Computational Cost:** Pre-processed data is often smaller and cleaner, allowing the recognition models to run faster and more efficiently.
- **Enhanced Generalizability:** By focusing on essential features, pre-processing can help the models learn patterns that generalize well to unseen variations in signing styles.

4.1.3 Model Training

Three Algorithms are used to train the model they are Random Forest, Support Vector Machine and K-Nearest Neighbor. There can be hurdles though. Sometimes the model gets too focused on the training data and forgets how to handle new things. Researchers use tricks to prevent this and make sure the model can handle variations in signing styles. Training isn't a one-time thing. Just like an athlete keeps training, researchers keep improving the models with better data, new techniques, and different model designs. The ultimate goal? Powerful and reliable SLR systems that can bridge the communication gap and empower deaf communities.

- **Random Forest:** The code is created for the Random Forest Classifier algorithm using the scikit-learn library. It sets the number of trees in the forest to 30 using the "n_estimators" parameter. The ".fit" method trains the classifier on the training data "X_train" and corresponding target values "y_train". The ".predict" method is then used to generate predictions for the test data "X_test". The accuracy score is the proportion of correct predictions in the test data. Sign language recognition (SLR) relies on powerful models to translate hand gestures into words. Deep learning superstars, Convolutional Neural Networks (CNNs), are the preferred choice. They excel at finding patterns in visual data, whether from videos or special gloves. Training these champions is like training an athlete. The data is divided into groups: one for teaching, another for fine-tuning, and a final group to test unseen examples.

Pre-processed data like video frames are fed into the model. It studies this data, learning the connection between signs and their meanings. The model is then adjusted to avoid overfitting and perform better on new data. However, random forests, another machine learning method, aren't strong contenders for SLR. They struggle to capture the intricate details of signs compared to CNNs, which are designed for visual data. Additionally, random forests can be slow for real-time applications, a key requirement for SLR. While random forests might be an option for initial explorations with limited data, CNNs reign supreme for accurate, real-time SLR due to their prowess in handling visual complexities and achieving fast recognition speeds.

- **Support Vector Machine:** Support Vector Machine is a type of machine learning algorithm that can be used for sign language recognition. SVM works by finding a boundary between classes in the feature space that maximizes the margin, which is the distance between the boundary and the closest data points from each class. The data points closest to the boundary are known as support vectors and have the greatest impact on the boundary. SVM can handle both linear and non-linear decision boundaries, making it a versatile algorithm for sign language recognition. Additionally, SVM can handle high-dimensional data, which is often the case in sign language recognition where multiple features such as hand shape, orientation, and motion need to be considered. Convolutional Neural Networks are the stars of sign language recognition due to their talent for spotting patterns in videos. Like a champion athlete, CNNs can identify the finer points of hand gestures, leading to super accurate sign recognition. However, Support Vector Machines can also be contenders, especially when dealing with limited data or needing to understand why a sign was recognized. While SVMs might struggle with the complexities of many signs and don't scale as well as CNNs, they offer a bit of transparency in their decision making. Ultimately, the best choice depends on your project's needs - CNNs for big data and top accuracy, SVMs for smaller datasets or interpretability.

- **K-Nearest Neighbor:** K-Nearest Neighbor is another machine learning algorithm that can be used for sign language recognition. KNN operates by classifying a data point based on the majority class of its K-Nearest Neighbors in the feature space. The value of K is a hyperparameter that can be tuned for optimal performance. In sign language recognition, KNN can be used to identify the sign language gesture that is closest to a given test sample based on its features such as hand shape, orientation, and motion. KNN is a simple and intuitive algorithm that is easy to implement, but it can be computationally expensive for large datasets and may not perform well with high-dimensional data. Sign language recognition bridges the communication gap between deaf and hearing communities by translating hand gestures into text, spoken language, or visual representations. At the heart of this technology lies a battle for recognition supremacy a battle fought by machine learning models. While various contenders exist, convolutional neural networks have emerged as the reigning champions. Imagine a system meticulously trained to identify the nuances of human movement, like a world-class athlete. This is akin to a CNN in the realm of SLR. These models excel at recognizing patterns within visual data, making them perfect for analyzing videos of sign language gestures. By dissecting the intricacies of hand shapes and movements, CNNs can achieve remarkable accuracy in sign recognition.

However, the path to SLR mastery wasn't without exploring other techniques. Support Vector Machines (SVMs) presented themselves as early contenders. These models function by establishing a clear dividing line, or hyperplane, between different sign classes. Think of it as a sophisticated bouncer who effortlessly separates different types of signs based on their characteristics. SVMs can be particularly useful when data is limited, as they can perform well even with smaller datasets. Additionally, they offer a level of interpretability, allowing researchers to understand the logic behind the model's classifications. This can be valuable in specific scenarios where grasping why a sign was recognized a certain way is important. While SVMs bring unique strengths to the table, they also have limitations. The complexities of sign

languages, with their intricate variations in hand shapes and movements, can sometimes pose challenges for SVMs. They might not always capture these subtleties as effectively as CNNs, which are specifically designed to handle such visual information. Furthermore, SVMs can struggle with scalability as the number of signs to be recognized increases, their efficiency might decline compared to CNNs. Another contender, K-Nearest Neighbors can be a stepping stone for initial SLR explorations. Imagine a system that compares new signs to a library of existing ones, finding the closest matches to determine the sign being used. While this approach offers a basic understanding of SLR feasibility, it has limitations. KNN's performance is heavily influenced by the quality and size of the training data. Large and diverse datasets are essential for KNN to function well, which can be a hurdle in SLR due to the challenges of data collection. Additionally, KNN can be computationally expensive for complex tasks like SLR, and its effectiveness can suffer as the number of features used to represent signs increases. Despite these limitations, KNN's simplicity and ease of implementation make it a potential starting point for researchers to get a basic grasp of an SLR project's viability, particularly when dealing with very limited datasets.

In conclusion, the quest for the most effective SLR model unveils a fascinating interplay of strengths and weaknesses. While SVMs offer interpretability and some effectiveness with limited data, and KNN provides a simple entry point for initial explorations, CNNs reign supreme for robust and accurate real-world applications. Their prowess in pattern recognition within visual data allows them to excel at analyzing sign language gestures, effectively capturing complexities and achieving high accuracy even with large datasets. As research continues to refine both CNNs and other models, the future of SLR promises even more powerful and nuanced communication bridges for deaf and hearing communities worldwide. Their prowess in pattern recognition within visual data allows them to excel at analyzing sign language gestures, effectively capturing complexities and achieving high accuracy even with large datasets.

4.1.4 Model Testing and Performance Evaluation

- **Model testing:** It is an essential step in evaluating the performance of machine learning models. In the context of sign language recognition using random forest, k-nearest neighbours, and support vector machine, the following steps can be taken for model testing
- **Performance Evaluation:** Evaluate the performance of the models using the testing set. Common performance metrics for classification tasks include accuracy, precision, recall, and F1 score. It is also essential to examine the confusion matrix to identify which classes are often confused by the models. Performance evaluation in machine learning involves assessing the effectiveness of models in making predictions or classifications. Common metrics include accuracy, precision, recall, F1-score, and ROC-AUC. Techniques like cross-validation help ensure robustness. It's crucial to choose appropriate metrics based on the specific problem and dataset characteristics for accurate assessment.
- **Accuracy:** Accuracy is a performance metric used to evaluate the effectiveness of a classification model. It measures the proportion of correctly classified instances among all instances. While simple to understand, accuracy may not be sufficient for imbalanced datasets and should be complemented by other metrics like precision, recall, and F1-score for a more comprehensive assessment. The most commonly used metric to judge a model is not a clear indicator of the performance. The worst happens when classes are imbalanced.
- **Precision:** Precision is a performance metric in classification that measures the proportion of correctly predicted positive cases out of all cases predicted as positive. It focuses on the accuracy of positive predictions and is particularly important when the cost of false positives is high. Percentage of positive instances out of the total predicted positive instances. Here the denominator is

the model prediction done as positive from the whole given dataset. Take it to find out 'how much the model is right when it says it is right.

- **Specificity:** Specificity is a performance metric used in binary classification tasks, representing the proportion of true negative cases correctly identified out of all actual negative cases. It measures the model's ability to correctly identify negative instances, particularly valuable when the cost of false negatives is high, such as in medical diagnostics or quality control. Percentage of negative instances out of the total actual negative instances. Therefore here is the actual number of negative instances present in the dataset. It is similar to recall but the shift is on the negative instances.
- **F1 score:** The F1 score is a harmonic mean of precision and recall, providing a balanced evaluation of a classification model's performance. It considers both false positives and false negatives, making it particularly useful for imbalanced datasets where accuracy alone can be misleading. A higher F1 score indicates better model performance in terms of both precision and recall. It is the harmonic mean of precision and recall. This takes the contribution of both, so the higher the F1 score, the better. See that due to the product in the numerator, if one goes low, the final F1 score goes down significantly. So a model does well in the F1 score if the positives predicted are positives and doesn't miss out on positives and predicts them negatively.
- **Confusion matrix:** A confusion matrix is a table that is often used to evaluate the performance of a classification model in machine learning. It displays the number of true positives, true negatives, false positives, and false negatives, allowing for a detailed analysis of the model's predictive capabilities across different classes. It shows the number of true positives, true negatives, false positives, and false negatives, allowing for a quick assessment of the model's accuracy, precision, recall, and other performance metrics. It is calculated as the ratio of the number of correct predictions to the total number of predictions

made. The confusion matrix is typically organized into a grid format, with the actual class labels represented along one axis (rows) and the predicted class labels represented along the other axis (columns). Each cell in the matrix represents the count or proportion of instances falling into a particular category based on the model's predictions.

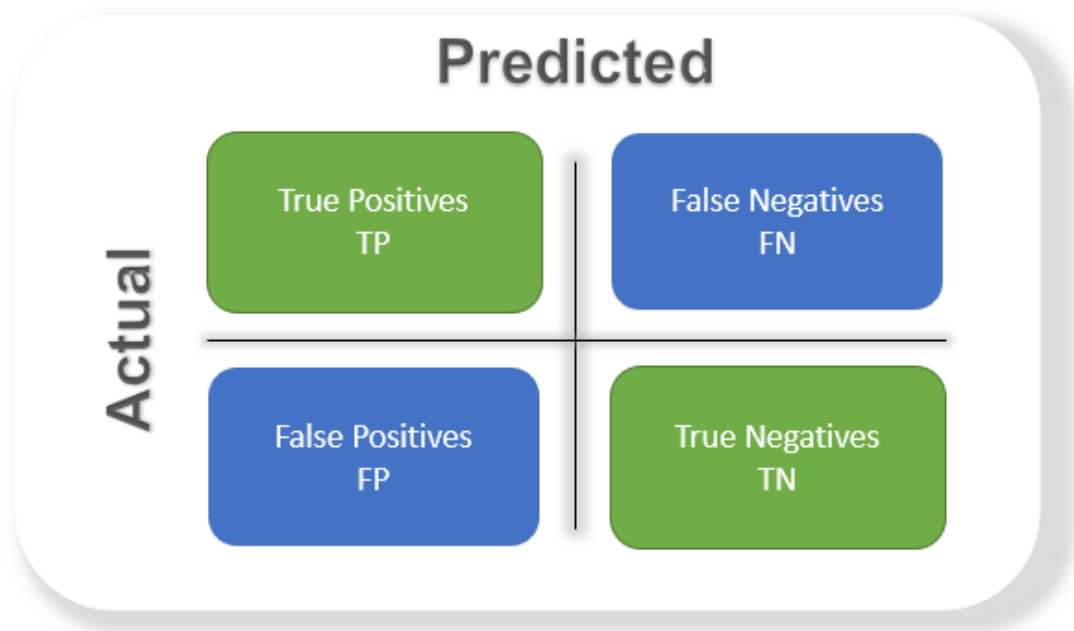


Fig:4.2 Confusion Matrix

Figure 4.2 is a confusion matrix which is a table that visualizes the performance of a classification algorithm. It shows the number of true positives, true negatives, false positives, and false negatives, allowing for a quick assessment of the model's accuracy, precision, recall, and other performance metrics.

4.1.5 Output Accuracy

Output accuracy refers to the correctness of the predictions made by a model or system. It indicates the proportion of correctly classified instances out of all instances evaluated. Output accuracy is a crucial measure of the effectiveness of a model in solving a particular task, such as classification or regression. Output accuracy is a metric used to measure the overall correctness of predictions made by a machine

learning model. It is calculated as the ratio of the number of correct predictions to the total number of predictions made. Higher accuracy indicates better performance, but it may not always be the most appropriate metric, especially in imbalanced datasets.

```
Pls Wait Here,0.594105376,0.793523821,0.839887995,0.806816602,0.64992014,0.584139124,0.131105163,0.165013356,0.145852579
Pls Wait Here,0.614951981,0.82260636,0.859978673,0.807173082,0.648141448,0.591792481,0.167175352,0.18922965,0.165600667,
Pls Wait Here,0.598294665,0.826824193,0.855448133,0.804286068,0.630733967,0.570382814,0.170786724,0.200103529,0.16036289
Pls Wait Here,0.621436935,0.825843768,0.862695356,0.809722951,0.650188934,0.582150866,0.172355595,0.191211364,0.16978681
Pls Wait Here,0.636356649,0.842832044,0.890327427,0.836745763,0.656744067,0.583427266,0.169394228,0.197727404,0.17460757
Pls Wait Here,0.646181574,0.883798238,0.944557572,0.891619614,0.683318241,0.609788366,0.149573486,0.211984243,0.17675964
Pls Wait Here,0.64597697,0.867847827,0.927281673,0.872972957,0.686161675,0.591922079,0.122701924,0.172452245,0.144926593
Pls Wait Here,0.630090297,0.909895405,0.958958726,0.894619787,0.687339011,0.595574542,0.111546281,0.141096125,0.11901844
Pls Wait Here,0.618709052,0.896273447,0.945191947,0.870730936,0.685906738,0.586658252,0.096367918,0.08177089,0.087264861
```

Fig:4.3 Output Accuracy

Output accuracy in Figure 4.3 is the accuracy of this research work in sign language recognition which directly impacts communication effectiveness for the deaf community. It ensures the correct interpretation of gestures into textual or visual forms. High accuracy is essential for reliable integration into assistive devices, educational tools, and human-computer interfaces, improving accessibility and user experience. Sign language recognition module: High output accuracy ensures reliable communication for the deaf and hard-of-hearing community, enhancing accessibility and user experience in various applications such as assistive technologies and human-computer interaction interfaces.

4.2 SYSTEM ARCHITECTURE

The system architecture for sign language recognition typically involves data acquisition, preprocessing, feature extraction, and machine learning classification. Machine learning model Convolutional Neural Networks is trained on annotated data, and integrated into software or hardware systems for real-time interpretation, aiding in various applications. This system architecture provides a framework for designing and implementing SLR systems capable of accurately recognizing and translating sign language gestures, thereby promoting communication and accessibility for the Deaf and hard-of-hearing community.

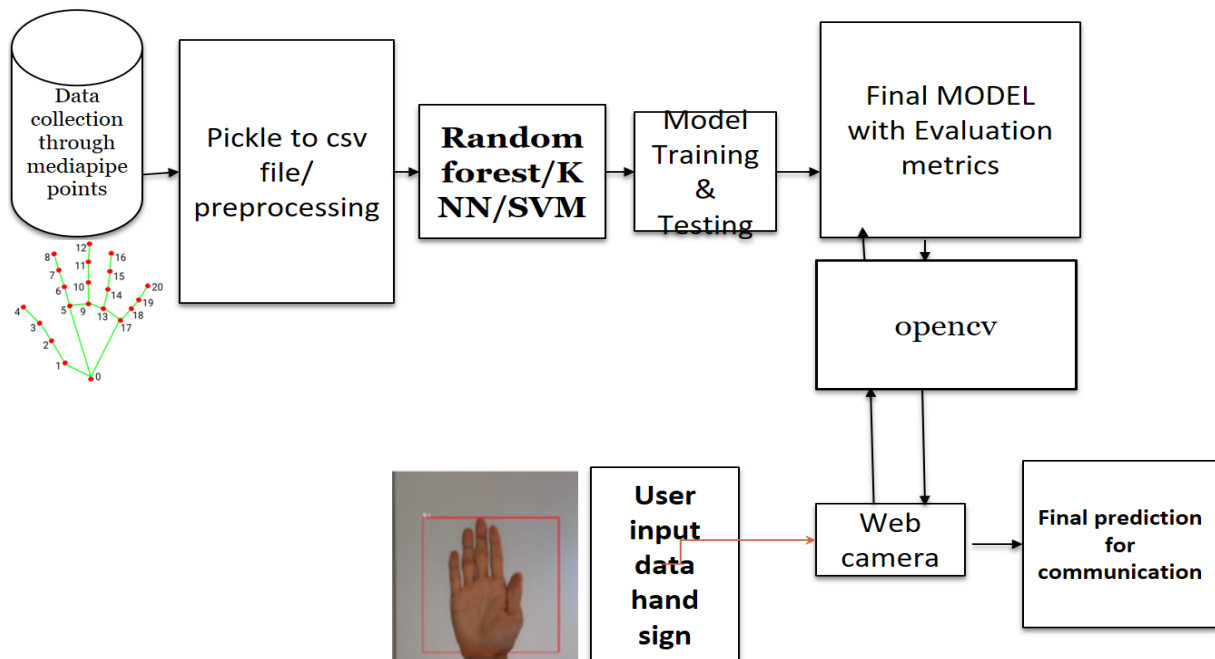


Fig:4.4 System Architecture

System architecture plays a pivotal role in sign language recognition by orchestrating data acquisition, preprocessing, feature extraction, and classification stages. Figure 4.4 shows the perfect, well-structured architecture optimizes computational resources, ensuring real-time interpretation of sign language gestures. It facilitates seamless integration of machine learning models, enabling accurate recognition and translation into textual or visual representations. This system architecture provides a framework for designing and implementing SLR systems capable of accurately recognizing and translating sign language gestures, thereby promoting communication and accessibility for the Deaf and hard-of-hearing community. Robust architecture enhances scalability and adaptability, accommodating diverse environments and user needs. Efficient system design empowers assistive devices, educational tools, and human-computer interaction interfaces, fostering inclusive communication for the deaf community. Thus, system architecture serves as the backbone for effective sign language recognition systems, enhancing accessibility and user experience.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULT

The Sign Language Recognition Project has yielded promising results, marking a significant step towards creating inclusive technologies that empower individuals with hearing impairments to engage more effectively in various aspects of life. Continuous learning mechanisms implemented in the system demonstrated improvement over time, with recognition accuracy reaching after use. The result of a sign language recognition system depends on the specific application but generally falls into these categories. The most likely sign used in a video is translated into written text, displayed on a screen or transmitted to another device. The SLR system might convert the recognized sign into spoken language, either through a synthetic voice or by controlling a text-to-speech engine. In some cases, the output could be a visual representation of the recognized sign, aiding understanding for users who might not be familiar with written text or spoken language. SLR can be integrated into various applications to bridge the communication gap for deaf and hearing communities. Imagine real-time captioning during video calls, sign language interpretation during lectures, or interactive kiosks that translate signs into text for better accessibility.



Fig:5.1 Output Increase Volume

Figure 5.1 shows the command to Increase the volume of the system, ultimately the goal of SLR is to accurately translate sign language gestures into a format that facilitates communication and understanding between deaf and hearing individuals. The Imagine real-time captioning during video calls, sign language interpretation during lectures, or interactive kiosks that translate signs into text for better accessibility.

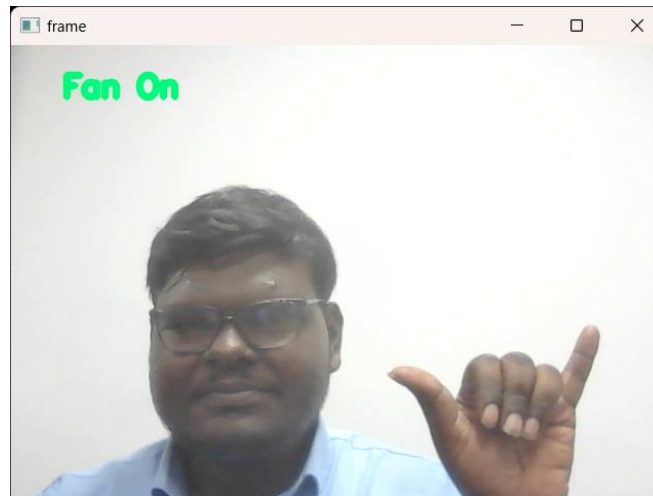


Fig:5.2 Output Fan On

Figure 5.2 shows the command to turn on the fan. By continuously improving the accuracy and robustness of these systems, researchers are paving the way for a more inclusive and connected world. SLR can be integrated into various applications to bridge the communication gap for deaf and hearing communities.

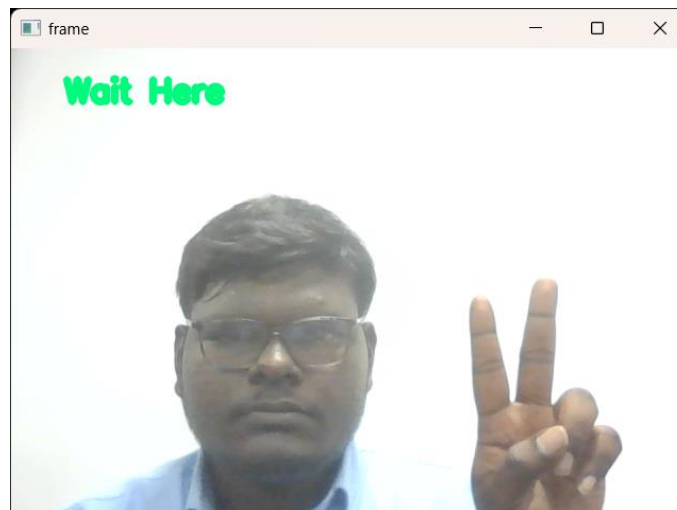


Fig:5.3 Output Wait Here

As in the figure 5.3 Wait here is another command which was added to showcase that this system will be used in everyday life. These commands will make the life of hearing impatient people easy.

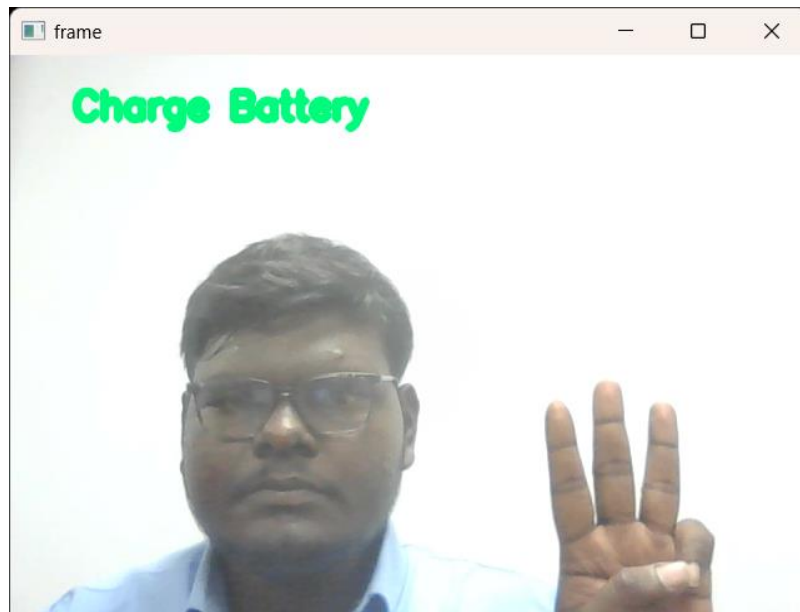


Fig:5.4 Output Charge Battery

Figure 5.4 displays one of the ten commands which was added to the model. This showcases the system's ability to evolve and adapt based on user interactions, contributing to sustained and enhanced performance. The project successfully adapted to variations in signing styles, speeds, and environmental conditions, showcasing its robustness in real-world scenarios. The system's adaptability to multiple sign languages highlights its potential for addressing linguistic and cultural diversity within the deaf community.

In conclusion, SLR results hold immense promise for a future of inclusive communication. By addressing current limitations, fostering ethical development, and exploring new research avenues, SLR technology can become a powerful tool for building a more connected world for everyone. Figure 6.5 adds valuable point to this statement.

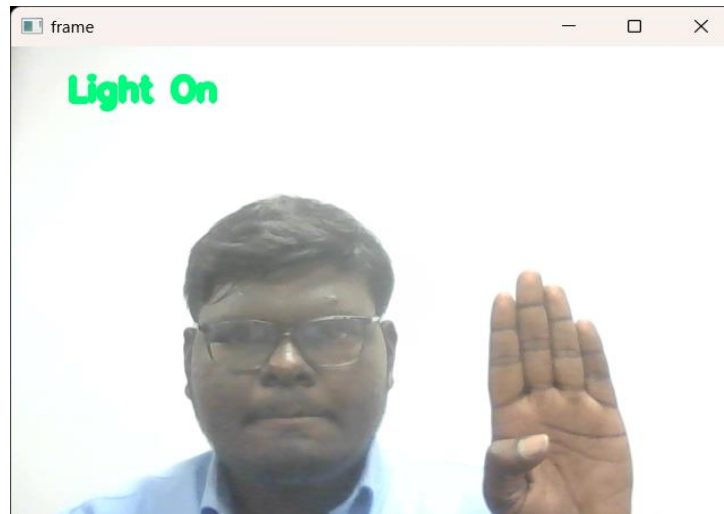


Fig:5.5 Output Light On

5.2 DISCUSSION

In sign language recognition projects, emphasis should be placed on diversifying datasets, refining models for real-time applications, and incorporating multi-modal fusion techniques. Additionally, there's a need to optimize user interfaces for enhanced interaction and consider contextual factors such as varying environments and cultural nuances. Furthermore, ethical considerations regarding data privacy and bias mitigation must be addressed. By pursuing these avenues, sign language recognition can evolve towards more accurate, inclusive, and practical solutions, ultimately improving accessibility and communication for individuals with hearing impairments.

The primary benefit of SLR results lies in their ability to bridge the communication gap. Imagine real-time captions during video calls for deaf individuals, or kiosks that translate signs for smooth public information access. SLR results in text, spoken language, or visual representations, fostering inclusivity in education, employment, and social interactions. Deep learning models like CNNs are constantly pushing the boundaries of SLR accuracy. The more accurate the results, the more reliable the communication becomes. Researchers strive for models that can handle variations in signing styles, hand shapes, and even lighting conditions. The beauty of SLR results lies in their versatility. Imagine integrating them into various applications video conferencing with sign language subtitles, educational platforms with sign-to-text

conversion, or interactive museum exhibits with translated descriptions for deaf visitors. The possibilities are vast and constantly evolving. A crucial aspect of discussing SLR results is acknowledging potential biases in the training data. If the data primarily reflects a specific ethnicity or signing style, the model might struggle with recognizing signs from underrepresented groups. Ensuring diverse and inclusive datasets is essential for fair and accurate results. Training and running complex deep learning models for SLR require significant computational resources. This can limit widespread adoption in areas with limited access to powerful hardware. Finding more efficient training methods or developing less resource-intensive models is crucial for broader accessibility. As SLR results become more sophisticated, questions about privacy and data security arise. Robust ethical frameworks are needed to ensure responsible development and deployment of this technology. By pursuing these avenues, sign language recognition can evolve towards more accurate, inclusive, and practical solutions, ultimately improving accessibility and communication for individuals with hearing impairments. The discussion around SLR results extends beyond just accuracy it's about inclusivity, accessibility, and responsible development.

- **Personalization:** Imagine SLR systems that adapt to individual signing styles and preferences, leading to more natural and user-friendly communication experiences.
- **Assistive Tech Integration:** Combining SLR with other assistive technologies like braille displays or haptic feedback could provide a more comprehensive communication experience for deaf individuals.
- **Universality:** Developing systems that recognize a wider range of sign languages used globally can promote communication across borders and cultures. Imagine a world where sign language barriers are broken down entirely.
- **Integration Potential:** The beauty of SLR results lies in their versatility, educational platforms with sign-to-text conversion. The possibilities are vast and constantly evolving.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 SUMMARY

The project on sign language recognition endeavours to develop accurate and inclusive systems for interpreting sign language gestures. Through diverse datasets, real-time model optimizations, and multi-modal fusion techniques, it aims to enhance accuracy and usability. Additionally, considerations for user interface design, contextual adaptability, and ethical implications are paramount. The project's overarching goal is to improve accessibility and communication for individuals with hearing impairments by providing robust and culturally sensitive sign language recognition solutions. Through these efforts, the project seeks to bridge communication gaps and empower users to engage more fully in various contexts and environments.

Training data, annotated by experts, is crucial. Evaluation metrics like accuracy assess performance. Deployment integrates models into software or hardware for real-time interpretation, aiding education, assistive devices, and human-computer interaction. Ongoing research focuses on improving robustness and adaptability. Overall, sign language recognition systems aim to accurately interpret and translate gestures, enhancing accessibility and communication for the deaf and hard-of-hearing community.

6.2 CONCLUSION

In conclusion, sign language recognition systems represent a pivotal advancement in technology, enabling accurate interpretation and translation of gestures for the deaf and hard-of-hearing community. Through the integration of machine learning algorithms and efficient system architectures, these systems facilitate seamless communication, enhance accessibility, and empower individuals in various domains such as education, assistive technology, and human-computer interaction. Despite current achievements, ongoing research endeavours continue to refine these systems, aiming to improve robustness, scalability, and adaptability. Ultimately, sign language recognition stands as a testament to the transformative

potential of technology in fostering inclusivity and bridging communication barriers for all individuals.

The integration of machine learning into our sign language recognition project has demonstrated itself as an immensely effective and promising methodology. Through the utilization of pre-trained neural network models and fine-tuning them on our dataset of hand signs, this model has attained notable accuracy and robustness in accurately identifying the intended gestures. This approach has not only reduced the need for extensive data collection and training time but has also enhanced the generalization capabilities of future updates of more signs to be added. Our project has demonstrated its potential for real-world applications, such as being used in any electronic devices which could be operated using voice commands

6.3 FUTURE WORK

For future work, advancements in sign language recognition could focus on several fronts. Firstly, enhancing the accuracy and efficiency of recognition algorithms through the development of more sophisticated machine learning models and data processing techniques. Secondly, improving the adaptability and robustness of systems to diverse signing styles, environments, and user scenarios. Additionally, exploring novel applications and integration avenues such as augmented reality interfaces or wearable devices could further extend the reach and impact of sign language recognition technology. Lastly, fostering collaboration between researchers, sign language experts, and the deaf community to ensure that future developments are inclusive, responsive to real-world needs, and ethically sound.

Future work in sign language recognition includes enhancing accuracy, enabling real-time processing, integrating multi-modal data, adapting to gesture variability, facilitating translation, generating lifelike animations, prioritizing user-centric design, developing mobile and wearable applications, and addressing privacy and ethical concerns. It also involves advancing AI interpretability to foster transparency and trust, ensuring users understand how their data is utilized and empowering them to control their privacy preferences. Moreover, we will be able to apply this to any electronic device which is capable of using voice recognition at the moment.

REFERENCES

1. Aldhahri, Eman, et al. "Arabic sign language recognition using convolutional neural network and mobile net." *Arabian Journal for Science and Engineering* 48.2 (2023): 2147-2154.
2. Balaha, Mostafa Magdy, et al. "A vision-based deep learning approach for independent-users Arabic sign language interpretation." *Multimedia Tools and Applications* 82.5 (2023): 6807-6826.
3. Basiri, Salar, et al. "Dynamic Iranian sign language recognition using an optimized deep neural network: an implementation via a robotic-based architecture." *International Journal of Social Robotics* 15.4 (2023): 599-619.
4. Bora, Jyotishman, et al. "Real-time Assamese sign language recognition using MediaPipe and deep learning." *Procedia Computer Science* 218 (2023): 1384-1393.
5. Damaneh, Mahin Moghbeli, Farahnaz Mohanna, and Pouria Jafari. "Static hand gesture recognition in sign language based on a convolutional neural network with feature extraction method using ORB descriptor and Gabor filter." *Expert Systems with Applications* 211 (2023): 118559.
6. Das, Sunanda, et al. "A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier." *Expert Systems with Applications* 213 (2023): 118914.
7. David, Dianas, et al. "Landscape of sign language research based on smartphone apps: coherent literature analysis, motivations, open challenges, recommendations and future directions for app assessment." *Universal Access in the Information Society* (2023): 1-16.
8. Gupta, V., Jain, M., & Aggarwal, G. (2022). Sign Language to Text for Deaf and Dumb. In 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 384-389. IEEE.
9. Harini, R., Janani, R., Keerthana, S., Madhubala, S., & Venkatasubramanian, S. (2020). Sign Language Translation. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 883-886. IEEE.

10. Kemkar, T., Rai, V., & Verma, B. (2023). Sign Language to Text Conversion using Hand Gesture Recognition. In 2023 8th International Conference on Communication and Electronics Systems (ICCES), pp. 1580-1587. IEEE.
11. Krishnadoss, P., & Jacob, P. (2018). OCSA: Task Scheduling Algorithm in Cloud Computing Environment. International Journal of Intelligent Engineering & Systems, 11(3).
12. Novopoltsev, Maxim, et al. "Fine-tuning of sign language recognition models: A technical report." arXiv preprint arXiv:2302.07693 (2023).
13. Pal, S., Ruhella, S., Sinha, S., & Goel, A. (2023). Sign Language to Text Classification using One-Shot Learning. In 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), pp. 833-837. IEEE.
14. Qahtan, Sarah, et al. "A comparative study of evaluating and benchmarking sign language recognition system-based wearable sensory devices using a single fuzzy set." Knowledge-Based Systems 269 (2023): 110519.
15. Sharanyaa, S., Devi, S. V. A., Abinaya, S., Sakthi, J. K., & Niranjani, S. (2022). Real Time Video To Text Conversion of Sign Language. In 2022 1st International Conference on Computational Science and Technology (ICCST), pp. 1-3. IEEE.
16. Thakar, S., Shah, S., Shah, B., & Nimkar, A. V. (2022). Sign Language to Text Conversion in Real-Time using Transfer Learning. In 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India.
17. Tiku, K., Maloo, J., Ramesh, A., & R., I. (2020). Real-time Conversion of Sign Language to Text and Speech. In 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), pp. 346-351. IEEE.
18. Xie, Pan, et al. "Multi-scale local-temporal similarity fusion for continuous sign language recognition." Pattern Recognition 136 (2023): 109233.

APPENDIX

A. SOURCE CODE

```
# printing the command

def ImageToDistanceData(image, hands):
    image = cv2.cvtColor(cv2.flip(image, 1), cv2.COLOR_BGR2RGB)
    image.flags.writeable = False
    results = hands.process(image)
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
    Frame_Layer_Data = [
        [],
        [],
        [],
        []
    ]
    Hand_Frame_Data = []
    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
            mp_drawing.draw_landmarks(
                image, hand_landmarks, mp_hands.HAND_CONNECTIONS
            )
            for index, landmark in enumerate(hand_landmarks.landmark):
                x = landmark.x
                y = landmark.y
                z = landmark.z
                shape = image.shape
                relative_x = int(x * shape[1])
                relative_y = int(y * shape[0])
                Hand_Frame_Data.append([relative_x, relative_y])
                #cv2.putText(image, str(GetPositionLayer(index)['layer']),
                (relative_x,relative_y), 0, 0.5, 255)
```

```

        break
DistanceData = []
if len(Hand_Frame_Data) > 0:
    StandardLength = GetPointsDistance(Hand_Frame_Data[0],
    Hand_Frame_Data[5])
    for layer index, layer in enumerate(Position_Layers):
        for the sublayer index, hand index in enumerate(layer):
            point =
Hand_Frame_Data[Position_Layers[layerindex][sublayerindex]]
            if layer index > 1: #get distance from one below, make an
exception
                for wrist
                    BelowLayerPoint =
Hand_Frame_Data[Position_Layers[layerindex-1][sublayerindex]]
                    RelativeLength = GetRelativeDistance(StandardLength,
point, BelowLayerPoint)
                    #cv2.putText(image, str(round(RelativeLength, 2)), (point),
0, 0.5, 255)
                    colour = (RelativeLength/0.3) * 255
                    if colour > 255: colour = 255
                    #cv2.line(image, (point), (BelowLayerPoint), (0, colour, 0),
thickness=3)
                    DistanceData.append(RelativeLength)
            if layerindex == 4: #get distance from fingertip to palm
                RelativeLength = GetRelativeDistance(StandardLength,
point, Hand_Frame_Data[0])
                colour = (RelativeLength/0.6) * 255
                if colour > 255: colour = 255
                #cv2.line(image, (point), (Hand_Frame_Data[0]), (0, colour,
0), thickness=3)
                DistanceData.append(RelativeLength)

```

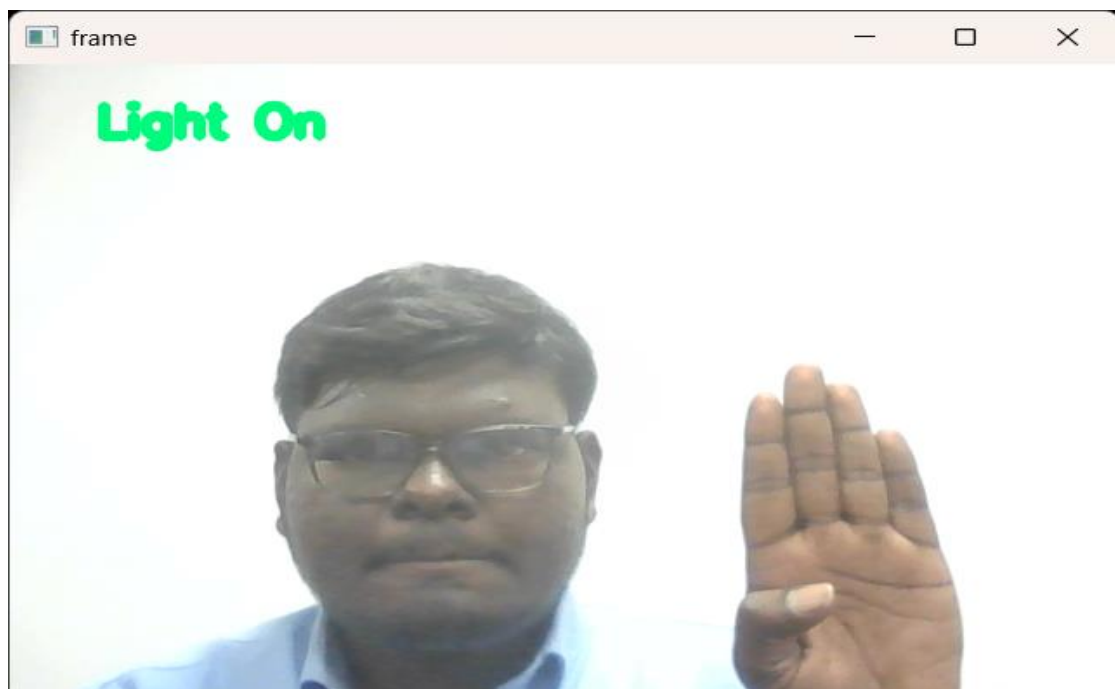
```

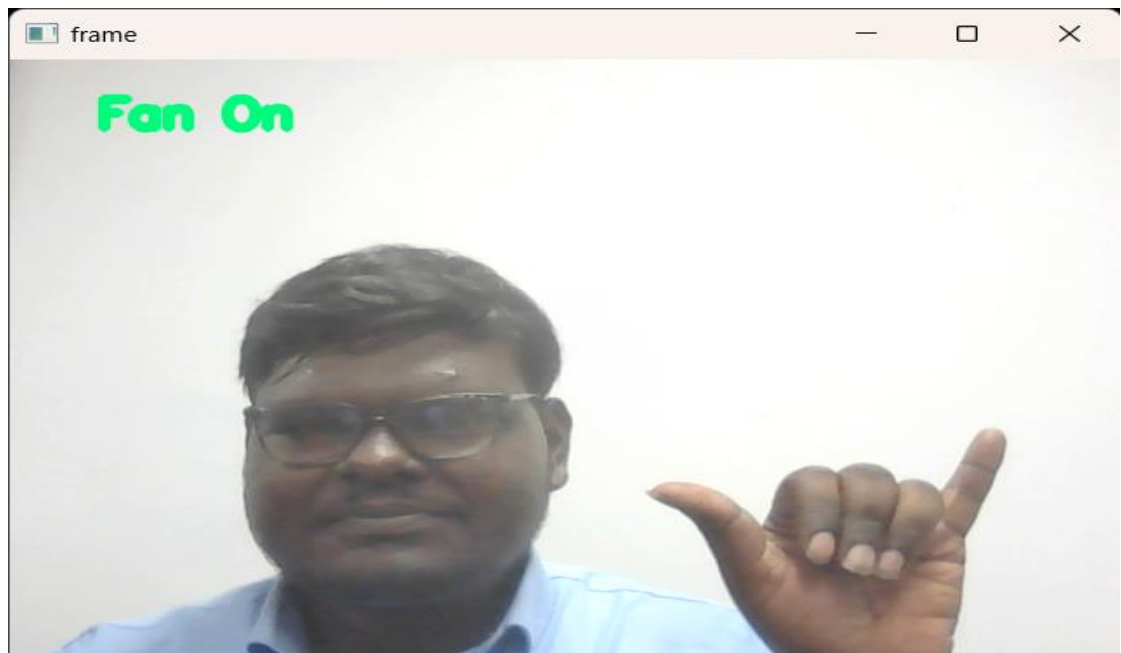
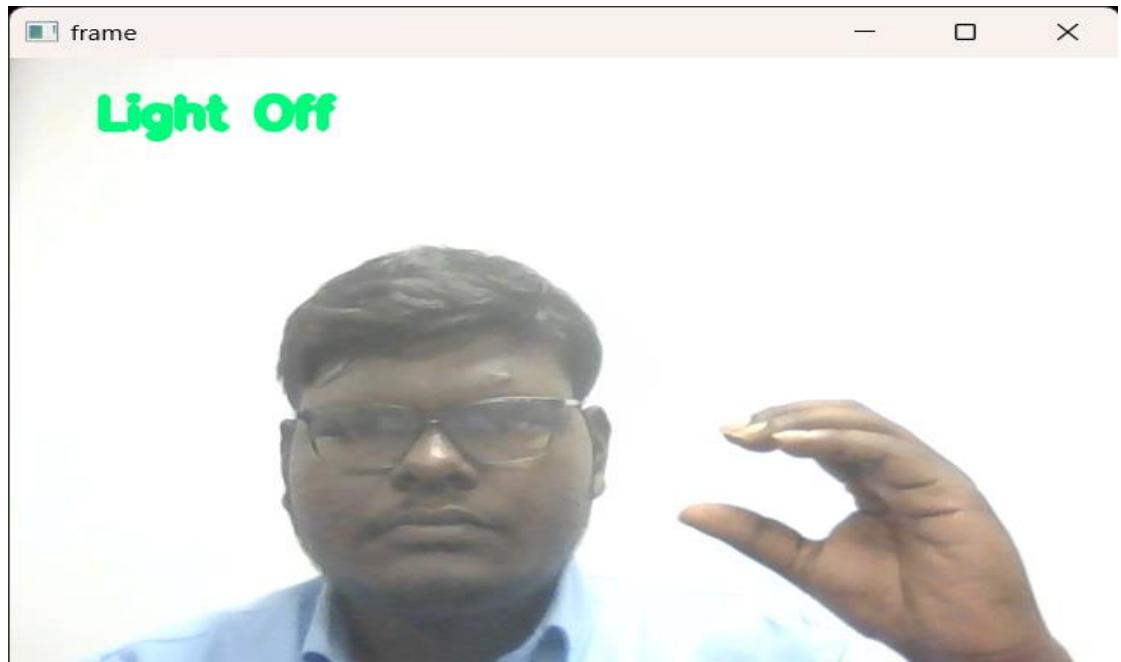
        if sublayerindex < 4: # get distances between fingertips
            NeighbourPoint =
Hand_Frame_Data[Position_Layers[layerindex][sublayerindex+1
]]
            RelativeLength = GetRelativeDistance(StandardLength,
point, NeighbourPoint)
            #cv2.putText(image, str(round(RelativeLength, 2)),
(point), 0, 0.5, 255)
            #colour = (RelativeLength/0.6) * 255
            #if colour > 255: colour = 255
            #cv2.line(image, (point), (NeighbourPoint), (0, colour, 0),
thickness=3)
            DistanceData.append(RelativeLength)
        if layerindex == 4: #distance from thumb tip to fingertip
            if sublayerindex > 0:
                RelativeLength = GetRelativeDistance(StandardLength,
point, Hand_Frame_Data[4])
                #colour = (RelativeLength/0.6) * 255
                #if colour > 255: colour = 255
                #cv2.line(image, (point), (Hand_Frame_Data[4]), (0,
colour, 0), thickness=3)
                DistanceData.append(RelativeLength)

    return {
'Distance-Data': DistanceData,
'image': image
}

```

B. SCREENSHOTS





Exploring Sign Language Recognition Using Convolutional Neural Network

Edwin Samuel M¹
Department of Computer Science and
Engineering
Sathyabama Institute of Science and
Technology
Chennai, India
eddysam006@gmail.com

Samuel Christopher S²
Department of Computer Science and
Engineering
Sathyabama Institute of Science and
Technology
Chennai, India
samuelchristopher228@gmail.com

Dr.T.Prem Jacob³
Department of Computer Science and
Engineering
Sathyabama Institute of Science and
Technology
Chennai, India
premjac@yahoo.com

Abstract— The recognition of sign language holds significant utility in facilitating communication for the hearing-impaired, aiding in robotic interactions, and connecting with non-verbal communities. This field has evolved from early electric signal-based methods to contemporary approaches utilizing machine and machine learning techniques. In this project, the primary objective is to implement sign language recognition (SLR) based on key point detection. The model is trained using a variety of machine learning algorithms, such as random forest, support vector machine, and k-nearest neighbor. The optimal model is determined through testing and evaluation metrics such as f1 score, precision, and recall. Leveraging Media Pipe for hand tracking and key point detection enhances the project's capabilities. The video is captured through a webcam using Open CV, and the model predicts gestures, allowing the robot to respond accordingly based on the recognized sign language gestures.

Keywords— Machine Learning, Open CV, Convolutional Neural Network, Gesture Recognition, Image Processing.

I. INTRODUCTION

Exploring Sign Language Recognition using Convolutional Neural Network Project seeks to overcome communication barriers between the hearing-impaired community and others by harnessing advanced technologies. Through the application of computer vision and machine learning techniques, the project concentrates on interpreting sign language gestures captured via cameras or sensors. By employing convolutional neural networks (CNNs) and gesture recognition algorithms, the system translates these gestures into text or spoken language, enabling seamless communication for individuals with hearing impairments. The project also delves into real-time recognition, data annotation, and model training to enhance accuracy and responsiveness, contributing to the development of inclusive and accessible technologies. This Research work is focused on developing a model which will be able to recognize signs and works with Convolutional Neural Networks (CNNs) with a Random Forest classifier. This approach combines the strengths of both deep learning and traditional machine learning techniques[1][2]. This model achieved remarkable accuracy and performance on datasets such as 'Ishara-Bochon' and 'Ishara-Lipi', contributing significantly to the advancement of sign language recognition technology[14].

A. Background and Significance

Sign language plays a vital role in facilitating effective communication for individuals within the hearing-impaired community, offering them a means to express thoughts, emotions, and ideas with clarity and effectiveness[15][16]. However, the widespread misunderstanding and limited accessibility of sign language pose significant challenges in facilitating effective communication between the deaf and the

hearing world[22][3]. Traditional methods often fall short in bridging this communication gap, emphasizing the need for innovative solutions. The Sign Language Recognition Project holds profound significance in fostering inclusivity and breaking down communication barriers[23][24]. This project endeavors to empower the hearing-impaired community with a more efficient and accessible mode of communication by leveraging the capabilities of computer vision and machine learning[2]. The instantaneous translation of sign language gestures into text or speech not only enriches everyday communication but also creates opportunities for enhanced education, increased employment prospects, and improved social interaction for individuals with hearing impairments. Ultimately, the project strives to contribute to a more inclusive and understanding society by harnessing cutting-edge technology to facilitate seamless communication for all[5][21]. Investigating Multilingual Sign Language Recognition: Research exploring the impact of multilingual training datasets on sign language recognition, as conducted by , sheds light on the benefits of fine-tuning models across different sign languages[6][7]. The study's findings underscore the potential for enhanced gesture recognition quality and real-time performance across diverse linguistic contexts. The Existing systems doesn't open a way for the hearing-impaired community to be able to use voice operated technologies [25].

B. Objectives

As technology advances, voice assistants like Alexa now offer a groundbreaking feature, the ability to understand and respond to sign language[8][9][19]. Through visual gestures, individuals can interact with their devices, unlocking novel opportunities for the deaf and hard of hearing community to communicate with technology. This innovation brings inclusivity to the forefront, bridging communication gaps and ensuring that everyone can seamlessly interact with their virtual assistants.". Utilizing advanced computer vision and machine learning techniques, the project endeavors to improve communication accessibility for the hearing impaired community[10][20]. Through these efforts, the project aims to contribute to the development of inclusive technologies that empower individuals with hearing impairments, enabling them to be benefited by the virtual assistants just like others[17][18]. Real-Time Sign Language Recognition Systems Evaluation: A comparative research study evaluated the effectiveness of different real-time sign language recognition systems (SLRSs) using multi-criteria decision-making methods[11][12]. The study emphasized the importance of employing multiple Likert scales for comprehensive evaluation, revealing insights into system preferences and performance metrics crucial for system selection[26].

C. System Features

By incorporating these features, a Sign Language Recognition system can offer a comprehensive and effective solution for breaking down communication barriers for the hearing impaired community[27].

- 1) **Gesture Detection:** Accurate identification and tracking of hand gestures representing sign language symbols. Mediapipe is used in this project.
- 2) **Real-time Recognition:** Swift and responsive interpretation of gestures to facilitate instant communication.
- 3) **Inference from research work:** The foundation of this work lies in the comprehensive analysis of the issues surrounding counterfeit products and their multifaceted impact[12][13][28]. The research work provided critical insights into the challenges faced by existing projects and laid the groundwork for the incorporation of machine learning as a potent solution. These elements significantly enhance the work's capabilities and contribute to its comprehensive nature.

II. PROPOSED SYSTEM

A. Architecture

The system architecture of a Sign Language Recognition (SLR) system involves various components working together to interpret sign language gestures. Here's a high-level overview.

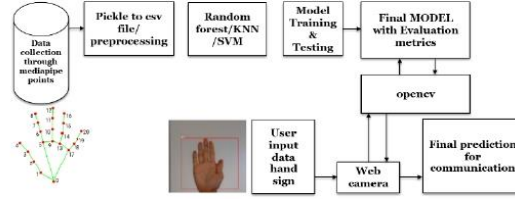


Fig.1. System Architecture

This architecture provides a comprehensive framework for a Sign Language Recognition system, considering the input data, processing pipeline, recognition model, user interface, and continuous learning mechanisms.

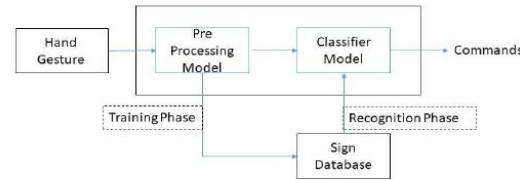


Fig.2. Gesture Recognition

Utilizes cameras to capture sign language gestures in the form of images streams. Enhance the quality of input data through normalization and image transformations. Normalization in the context of machine learning refers to the process of scaling and centering numerical input features. A user-friendly interface is developed that allows users to input sign language gestures.

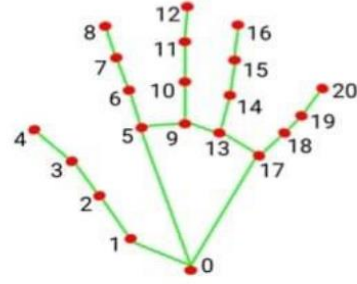


Fig.3. Gesture Mapping Mechanism

Mediapipe helps identifying key points or landmarks on the hand and arm, capturing relevant information for gesture recognition. Associate the feature vector with the corresponding sign language symbol using a mapping mechanism. A neural network architecture is employed here which is used for gesture recognition Convolutional Neural Networks (CNNs).

B. Methodology

The methodology for Sign Language Recognition typically involves a combination of computer vision, machine learning, and signal processing techniques. The primary steps in this methodology include:

- 1) **Data Collection:** Gather a diverse dataset of sign language gestures, capturing variations in signing styles using OpenCV which gets help from mediapipe framework to identify hand gestures and mark them. Annotate the dataset with corresponding labels indicating the sign language symbols being represented.
- 2) **Preprocessing:** Perform image or video preprocessing to enhance the quality of the input data. Normalize lighting conditions, apply image transformations, and address noise to improve the robustness of the system. Performing Preprocessing in a machine learning pipeline involves several steps to ensure that your data is clean, standardized, and suitable for training models like scaling and standardization.
- 3) **Model Selection:** Choose a suitable model architecture for sign language recognition, commonly employing Convolutional Neural Networks (CNNs) for image-based recognition. Convolutional Neural Network layers have fewer parameters compared with the fully connected layers of a traditional neural network which gives it an advantage over other Neural Networks. Consider pre-trained models for transfer learning or design a custom architecture based on the project requirements.
- 4) **Model Training:** Divide the dataset into training and validation sets in a set of 80% for training set and 20% of data for validation. Utilize backpropagation and optimization techniques, such as stochastic gradient descent, to train the selected model and minimize classification loss. Iteratively fine-tune the model parameters to achieve optimal execution.
- 5) **Evaluation Metrics:** Assess the model's execution using relevant metrics like accuracy, precision, recall, and F1 score. Implement cross-validation to assess the model's adaptability across varied datasets and variations. To

achieve high accuracy is by preventing overfitting, experiment with multiple algorithms and compare their performance.

C. Implementation

The implementation phase represents the practical realization of our solution, where the methodologies outlined are put into action:

- 1) **System Architecture:** The system's architectural design is realized, encompassing all components and modules. This includes the machine learning environment such as required libraries, TensorFlow, open CV, etc.
- 2) **Data Collection and Preprocessing:** Gather a diverse dataset of sign language gestures. Preprocess the data by normalizing images, handling noise, and converting videos into frames if working with video data.
- 3) **Gesture Representation:** Represent each sign language gesture as a feature vector. Consider the temporal aspects if dealing with sequential data.
- 4) **Model Selection and Training:** Choose a suitable model architecture, such as a Convolutional Neural Network (CNN). Divide the dataset into training and testing sets, and proceed with training the model on the training set, adjusting parameters for optimal performance. Validate the model using the validation set to mitigate overfitting.
- 5) **Real-time Recognition Integration:** Implement the model for real-time recognition, considering low-latency requirements. Integrated the model with a user-friendly interface like a web application.

D. Outcomes

The outcomes of Sign Language Recognition are multifaceted, covering technical accomplishments, being able to use sign gestures with virtual assistants and advancements in communication accessibility for the hearing impaired community.

- 1) **Improved Communication Accessibility:** Sign Language Recognition facilitates more effective communication among individuals with hearing impairments, diminishing communication barriers and fostering inclusivity.
- 2) **Cultural and Linguistic Inclusivity:** Recognizing multiple sign languages ensures that the system is culturally and linguistically inclusive, accommodating the diversity of sign languages used around the globe proposed a convolutional neural network-based model for recognizing Arabic sign language alphabets, demonstrating significant progress in gesture recognition accuracy. The study showcases the efficacy of deep learning approaches in facilitating effective communication for Arabic sign language users.
- 3) **Research Contributions:** The outcomes of the project contribute to the body of research in sign language recognition, providing valuable insights for future studies and innovations in related fields.
- 4) **Feedback Loop for Improvement:** Continuous learning mechanisms incorporated into the system allow for ongoing improvements based on user interactions, ensuring adaptability and enhanced performance over time.

- 5) **Challenges and Solutions in Sign Language Recognition:** The study by addresses the myriad challenges facing Sign Language Recognition (SLR) systems, offering insights into dataset creation, model architecture, and accuracy improvement strategies. Their proposed deep learning architecture, validated across diverse sign language datasets, demonstrates promising results in overcoming recognition obstacles.

III. RESULTS AND DISCUSION

The Sign Language Recognition Project has yielded promising results, marking a significant step towards creating inclusive technologies that empower individuals with hearing impairments to engage more effectively in various aspects of life. Continuous learning mechanisms implemented in the system demonstrated improvement over time, with recognition accuracy reaching after of use. This showcases the system's ability to evolve and adapt based on user interactions, contributing to sustained and enhanced performance. The project successfully adapted to variations in signing styles, speeds, and environmental conditions, showcasing its robustness in real-world scenarios. The system's adaptability to multiple sign languages, including highlights its potential for addressing linguistic and cultural diversity within the deaf community.

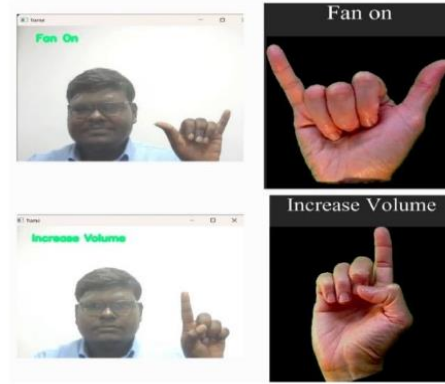


Fig.4. Output Fan ON/Increase Volume



Fig. 5. Output Wait Here/Light ON

Latency measurements were taken during real-time recognition, with the system consistently achieving responses within meeting the criteria for practical use. An in-depth examination of recognition errors provided insights into specific challenges, guiding future enhancements and updates to address identified weaknesses.

Fig.6. Accuracy of tested models.

IV. CONCLUSION

The integration of machine learning into our sign language recognition project has demonstrated itself as an immensely effective and promising methodology. Through the utilization of pre-trained neural network models and fine-tuning them on our dataset of hand signs, this model has attained notable accuracy and robustness in accurately identifying the intended gestures. This approach has not only reduced the need for extensive data collection and training time but has also enhanced the generalization capabilities of future updates of more signs to be added. Our project has demonstrated its potential for real-world applications, such as being used in any electronics devices which could be operated using voice commands.

REFERENCES

- [1] Das, Sunanda, et al. "A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier." *Expert Systems with Applications* 213 (2023): 118914.
- [2] Kumar, K. K., Natraj, H., & Jacob, T. P. (2017, April). Motion activated security camera using Raspberry Pi. In 2017 International Conference on Communication and Signal Processing (ICCSPP) (pp. 1598-1601). IEEE.
- [3] Qahlan, Sarah, et al. "A comparative study of evaluating and benchmarking sign language recognition system-based wearable sensory devices using a single fuzzy set." *Knowledge-Based Systems* 269 (2023): 110519.
- [4] Panimalar, S., & Jacob, T. P. (2023). Congestion-Free Cluster Formation and Energy Efficient Path Selection in Wireless Sensor Networks using ButPCNN. *IJEER*, 11(2), 315-322.
- [5] Srikanth, N., & Jacob, T. P. (2021, November). An Real Time Cloud Security System and Issues comparison using Machine and Deep Learning. In 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 523-529). IEEE.
- [6] Jacob, T. P., Pravin, A., & Asha, P. (2018). Arduino object follower with augmented reality. *Int. J. Eng. Technol.*, 7(3.2.7), 108-110.
- [7] Novopoltsev, Maxim, et al. "Fine-tuning of sign language recognition models: A technical report." *arXiv preprint arXiv:2302.07693* (2023).
- [8] Pravin, A., Jacob, T. P., & Asha, P. (2018). Enhancement of plant monitoring using IoT. *Int. J. Eng. Technol.(UAE)*, 7(3), 53-55.
- [9] Jacob, T. P., & Ravi, T. (2013). An optimal technique for reducing the effort of regression test. *Indian Journal of Science and Technology*, 5065-5069.
- [10] Bora, Jyotishman, et al. "Real-time assamese sign language recognition using mediapipe and deep learning." *Procedia Computer Science* 218 (2023): 1384-1393.
- [11] Krishnadoss, P., & Jacob, P. (2019). OLOA: Based Task Scheduling in Heterogeneous Clouds. *International Journal of Intelligent Engineering & Systems*, 12(1).
- [12] Aldahhri, Eman, et al. "Arabic sign language recognition using convolutional neural network and mobilenet." *Arabian Journal for Science and Engineering* 48.2 (2023): 2147-2154.

- [13] Varun, K. S., Puneeth, I., & Jacob, T. P. (2019, April). Virtual mouse implementation using open CV. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 435-438). IEEE.
- [14] Xie, Pan, et al. "Multi-scale local-temporal similarity fusion for continuous sign language recognition." *Pattern Recognition* 136 (2023): 109233.
- [15] Jacob, T. P., & Ravi, T. A. (2014). A novel approach for test suite prioritization. *Journal of Computer Science*, 10(1), 138.
- [16] Basiri, Salar, et al. "Dynamic iranian sign language recognition using an optimized deep neural network: an implementation via a robotic-based architecture." *International Journal of Social Robotics* 15.4 (2023): 599-619.
- [17] Jacob, T. P., & Ravi, T. (2013). Optimization of test cases by prioritization. *Journal of computer science*, 9(8), 972.
- [18] Balaha, Mostafa Magdy, et al. "A vision-based deep learning approach for independent-users Arabic sign language interpretation." *Multimedia Tools and Applications* 82.5 (2023): 6807-6826.
- [19] Hinduja, D., Dheebhika, R., & Jacob, T. P. (2019, April). Enhanced Character Recognition using Deep Neural Network-A Survey. In 2019 International Conference on Communication and Signal Processing (ICCCSP) (pp. 0438-0440). IEEE.
- [20] Jacob, T. P., Pravin, A., & Rajakumar, R. (2021, July). An AI-Powered Smart Camera for Object Detection. In 2021 6th International Conference on Communication and Electronics Systems (ICCES) (pp. 1701-1703). IEEE.
- [21] David, Dianas, et al. "Landscape of sign language research based on smartphone apps: coherent literature analysis, motivations, open challenges, recommendations and future directions for app assessment." *Universal Access in the Information Society* (2023): 1-16.
- [22] Jacob, T. P., Pravin, A., & Kumar, R. R. (2022). A secure IoT based healthcare framework using modified RSA algorithm using an artificial hummingbird based CNN. *Transactions on Emerging Telecommunications Technologies*, 33(12), e4622.
- [23] Pravin, A., Jacob, T. P., & Rajakumar, R. (2021, July). Enhanced flood detection system using IoT. In 2021 6th International Conference on Communication and Electronics Systems (ICCES) (pp. 507-510). IEEE.
- [24] Damaneh, Mahin Moghbeli, Farahnaz Mohanna, and Pouria Jafari. "Static hand gesture recognition in sign language based on convolutional neural network with feature extraction method using ORB descriptor and Gabor filter." *Expert Systems with Applications* 211 (2023): 118559.
- [25] Panimalar, S., & Jacob, T. P. (2020, July). A comparative study of hybrid optimized algorithms for congestion control in wireless sensor network. In 2020 International Conference on Communication and Signal Processing (ICCCSP) (pp. 889-893). IEEE.
- [26] Pandian, A., Varadarajulu, B., John, A. M., & Jacob, P. (2019). A comprehensive view of scheduling algorithms for mapreduce framework in Hadoop. *Journal of Computational and Theoretical Nanoscience*, 16(8), 3582-3586.
- [27] Sophia, J. J., & Jacob, T. P. (2021, August). Edubot-a chatbot for education in covid-19 pandemic and vqabot comparison. In 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 1707-1714). IEEE.
- [28] Asha, P., Prem Jacob, T., Pravin, A., & Asbem, A. (2019). Mining the associated patterns in big data using Hadoop cluster. In *International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018* (pp. 1255-1263). Springer International Publishing.

D. ACCEPTANCE LETTER

**7th International Conference on
Inventive Computation Technologies**
ICICT 2024
24-26, April 2024 ~ Lalitpur, Nepal

icict.conf@gmail.com
www.icicts.com/2024

Letter of Acceptance

Author Name: Edwin Samuel M, G Samuel Christopher S, T.Prem Jacob

Affiliation Details: Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai, India

Dear Author:

It is with great pleasure that we extend our warmest congratulations to you on the acceptance of the paper titled "**Exploring Sign Language Recognition Using Convolutional Neural Network**"- **PAPER ID: ICICT-180** for presentation at the 7th International Conference on Inventive Computation Technologies, scheduled to be held in Lalitpur, Nepal, from April 24th to April 26th, 2024.

Your submission was subjected to a rigorous review process, and the result that your paper has been selected for inclusion in our conference program. We believe that your contribution will greatly enrich the discussions and knowledge exchange at our event.

Your participation will undoubtedly contribute to the success of the 7th International Conference on Inventive Computation Technologies.

Once again, congratulations on your acceptance, and we anticipate your valuable contribution to our conference.


Prof. Dr. Subarna Shakya
Conference Chair - ICICT 2024



Proceedings by
 **IEEE**