GESTURE SENSING: BRIDGING REAL AND VIRTUAL WORLDS

Sowmya S R[1] , Prajna Acharya[2] , R Kavitha[2] , Raksha G[2] , Rakshitha Malnad P[2]

[1]*Assistant Professor, Department of Information Science and Engineering, Dayananda SagarAcademy of Technology and Management, Bengaluru, India*
[2]*B.E. Students, Department of Information Science and Engineering, Dayananda Sagar Academy ofTechnology and Management, Bengaluru, India*

Abstract

Approximately 5% of the global population, comprising deaf and mute individuals, often lack access to sign language, leading to feelingsof disconnection. A prototype assistive medium has been developed to address this gap by recognizing hand gestures and converting them into real-time text using image processing and deep learning techniques. Hand gestures play a vital role in facilitating human-machine interaction and are integral to vision-based gesture recognition technology. The system involves tracking, segmentation, gesture acquisition, feature extraction,gesture recognition, and text conversion, all crucial in bridging communication between deaf and mute individuals and those who can hear and speak.

Key Words: *Facial recognition using OpenCV andPython incorporates advanced algorithms such as LSTM, SVM, RNN, and ANN for accurate identification.*

---

1. INTRODUCTION

Sign language, reliant on hand movements and facial expressions, is primarily used by the hearing impaired for communication, but is underutilized by those without hearing impairments, hindering social interaction. Real-time interpretation is costly and not always available, necessitating an automatic translation system. Recent developments focus on translating sign language into OpenCV, recognizing personal signs, and convertingthem into standard text. Sign languages, naturaland complete, utilize manual communication, with distinct grammar and lexicon. While they vary, they share non-universal characteristics.

Sign language, not exclusive to the hearing impaired, is significant for overcoming communication barriers, particularly in noisy environments. Distinguishing it from body language, sign language relies on manual communication, not sound patterns. A proposedcharacter recognition system aims to enhance communication between sign language users and non- users, addressing various challenges like social interaction and education. Recognizing hand gestures efficiently, it fosters inclusivity and better communication. Body language, a type of non-verbal communication, differs from sign language, which relies on manual communication rather than acoustically transmitted sounds. Sign language employs hand gestures, facial expressions, and body movements like eye and leg gestures for conveying meaning. This paper introduces a character recognition and interpretation design aiming to address communication barriers between sign language users and non-users. Challenges faced by those with hearing or speech impairments in communication with non-sign language users include social interaction, education, behavioral issues, mental health, and safety concerns. Gestures encompass physical actions by the hand, eye, or body, with hand gestures being most easily understood. The proposed mark recognition system offers high accuracy and efficiency, bridging the communication gap andfostering inclusivity for improved communication.

2. MOTIVATION

The "Sign Language Detection Using LSTM" project is driven by the aim to bridge the communication barrier between sign languageusers and non-users. While sign language serves as the primary mode of communication for many individuals who are deaf or hardof hearing, its limited understanding poses challenges for effective interaction and inclusion. Thus, the project endeavors to create a system capableof accurately identifying and interpreting sign language gestures, facilitating smoother communication between sign language users and others.

Beyond enhancing inclusivity and accessibility for thosewith hearing impairments, the project seeks to empower sign language users by enabling them to communicate independently. With a dependable sign language detection system, individuals can assert greaterautonomy in their daily interactions. This tool allows them to articulate their thoughts, requirements, and emotions without relying solely on interpreters or written text. Such empowerment holds the potential to boost self-confidence, foster self-expression, andimprove overall quality of life.
Real-time communication is another key motivation for the project. The capability to interpret sign language gestures instantly holds immense value, particularly in situations demanding prompt communication. For instance, in emergencies, medical settings, or public interactions, a system adept at swiftly and accurately interpreting sign language can facilitate efficient communication, ensuring timely assistance, access toessential services, and equitable opportunities for sign language users.

Furthermore, the project aims to support the teachingand learning of sign language. By developing a sign language detection system using LSTM, it can serve as avaluable resource for individuals learning sign language. The system can offer visual feedback, guidance, andassistance in comprehending and practicing various sign gestures. It can be integrated into educational platforms, interactive applications, or virtual tutors to enrich the learning process, maintain consistency, and deliver personalized feedback to learners.

Finally, the project is in line with the advancement of technology and contributes to the field of deep learning and pattern recognition. Developing a sign language detection system using LSTM involves solving complex problems such as analyzing time dependencies and recognizing subtle hand movements.

By pushing the boundaries of research and innovation incomputer vision and machine learning, the project can contribute to the development of more robust and accurate systems for a variety of real-world applications.In summary, the project aims to provide an effective means of communication for people who use sign language, promote inclusivity and accessibility, empower individuals, support teaching and learning, and advance technology.

3. PROBLEM STATEMENT

Sign language serves as a crucial communicationmethod for many individuals with hearing impairments, yet its limited understanding poses significant challenges. Current solutions like human interpreters or written communication have drawbacks in availability, accessibility, and real-time responsiveness, hindering inclusivity and independence for sign language users. Additionally, sign language's complexity, involvingintricate hand movements, facial expressions, and body language, demands accurate and timely interpretationfor effective communication.

Hence, this project endeavors to develop a sign languagedetection system utilizing LSTM, a deep learning architecture adept at modeling sequential data. The system aims to precisely recognize and interpret sign language gestures from video inputs in real-time,facilitating effective communication between sign language users and non-users. Ultimately, the project strives to bridge the communication divide, foster inclusivity, empower sign language users, and bolster their autonomous communication across various domains like education, employment, healthcare, andsocial interactions.

In pursuit of this objective, the project must confront various obstacles, including accurately capturing and depicting temporal dependencies in sign language gestures, managing the diversity and subtleties of different sign language styles and situations, guaranteeing real-time responsiveness for smooth communication, and attaining high levels of precision and reliability in gesture recognition. Creating a sign language detection system with LSTM holds the potential to foster a more inclusive society, empowering individuals reliant on sign language to engage effectively and fully in diverse activities, thereby diminishing barriers and advocating for equal opportunities for all.

4. LITERATURE SURVEY

Marouane Benmoussa and his colleagues devised a method for recognizing human-computer gestures, reflecting the growing interest in gesture recognition, particularly with advancements in human-computer interaction (HCI) technology. The aim is to teach computers to communicate in human-like ways by understanding human speech, facial expressions, and gestures. Utilizing Kinect's Skeletal Tracking, the system identifies and interprets hand motions,leveraging Scale Invariant Features Transform (SIFT) and Speeded Up Robust Features (SURF), trained with K-means and SupportVector Machine (SVM) classifiers.

Initially, depth photos for 16 different movements were collected using the Microsoft Kinect sensor to construct the Hand Gesture Recognition (HGR) system, ensuring clarity by eliminating background uncertainty with depth data. While SIFT and SURF generate sets of key points to represent hand gesture training images, the varying numberof key points lacks a clear hierarchy. To address this, a bag-of-words strategy, a prominent approach in computer vision, is employed.

Moreover, SURF outperforms SIFT in defining gesture key points by a difference of 50 key points and demonstrates effectiveness against scale, rotation, and translation biases. The study introduces a machine learning technique to instantly identify 16 user hand motions using Kinect sensor data. A support vector machine model trained with hand depth data extracts a bag of words containing SIFT and SURF descriptors. Evaluation using the area under the ROC curve indicates a 98% performance for SURF and 91% for SIFT. This suggests SURF's suitability for real-time applications, being three times faster than SIFT.

Greeshma Pala et al. proposed a system for recognizing hand gestures, which has gained significance due to its communication benefits and versatile applications. The system offers three initial modes: Hand Sign to Text Recognition, Hand Gesture to Speech, and Hand Gesture Recognition. Utilizing a vision- based method, it doesn't require additional technology beyond a web camera for hand motion recognition. Images are gathered and processed into grayscale, then downsized to75x75 for uniformity. The K-Nearest- Neighbor (KNN) classification algorithmcategorizes images based on the majority class of neighboring points in Euclidean space. SVM is used for regression and classification, identifying the optimal separating line tocreate an ideal separation hyperplane. CNNs, specialized in image processing, are employed,with parameters such as weights and biaseslearned during training. The pickle model loads the dataset for processing. And test split iscarried out after the data set

A human-computer hand gesture recognition and classification system for the deaf and mutewas proposed by Nitesh S et al. A gesture can be defined as any physical action made withthe hand, eye, or any other part of the body. The most humane and simple to understandmotions are hand gestures. The webcam input image is recorded and can be utilised as an input image for character recognition or to store as a training dataset. Images that are captured are RGB files. The photos that were obtained had very high pixel values and Nitesh S et al. proposed a human-computer hand gesturerecognition and classification system tailored for the deaf and mute community. Hand gestures, considered asany physical action made with the hand or other body parts, serve as a humane and easily understandable means of communication. The system records webcam input images for character recognition or training datasetstorage. As the captured RGB images have high pixel values and dimensions, they are converted to grayscale and then binary using the "rgb2gray" function.

Segmentation divides the image into background and foreground, focusing on the area of interest. Principle Component Analysis (PCA) is utilized for feature extraction, where eigenvalues and eigenvectors are determined by combining all images into a column matrix, averaging, and normalizing. Hand gestures are organized based on Euclidean distance, constructing eigenvectors for the dataset during training.

The system's accuracy during the training phase directly relates to the number of photos stored per character. During the test phase, input gestures are recognized, anda maximum score is calculated based on Euclidean distance. The accuracy table displays the percentage accuracy for each character. Character identification is facilitated by Euclidean distance, enabling direct interaction for individuals with disabilities without relying on sign language. However, the system's accuracy is influenced by lighting conditions.

A deep learning-based character action detection system was proposed by Shivanarayna Dhulipala et al. Cognitive issues, such as those pertaining to sign languages and their constraints, have become easier to tackle as a result of the rapid rise in computer use and artificial intelligence. The most important development in artificial intelligence is deep learning, which is used to teach computer systems how to recognise, decipher, and translate letters into written language. As a result, the focus of this dissertation will be on employing deep learning to

apply LSTM and CNN models to the detection of human activity and sign language. to bridge.

The communication divide between those who can hear and those who are persists. Within neural networks, a CNN model plays a vital role in detecting characters and faces within images. These models consists of neurons with biases and adjustable weights. Certain neurons process input data, computing weighted sums that , when triggered by stimuli, active particular functions and generate specific outputs. CNN models are commonly utilized in multi-channel modes.

5. TECHNOLOGIES USED

A. Python

Python, a high-level, versatile programming language, operates in an interpreted manner, allowing it to execute code line by line without compilation. Its notable feature is its focus on code readability, facilitated by significant whitespace characters. With its object-oriented programming approach and diverse language constructs, Python is adept at handling projects of all scales, facilitating the creation of clear and maintainable code by developers.

B. IDE (Jupyter)

The Jupyter Notebook provides a user-friendly, interactive environment for data science across multiple programming languages, functioning both as an integrated development environment (IDE) and a platform for presentations or instructional purposes.

C. Numpy (version 1.16.5)

NumPy, a Python package, is employed for array manipulation, offering functionalities for matrix operations, Fourier transforms, and linear algebra computations. Travis Oliphant is credited with developing NumPy in 2005, often referred to as Numerical Python.

D. OpenCV

OpenCV stands as a comprehensive open-source toolkit dedicated to image processing, machine learning, and computer vision tasks. Supporting a wide range of programming languages like Java, C++, Python, and others, OpenCV is capable of analyzing both movies and images for facial recognition, object detection, and handwriting identification. Its capabilities expand significantly when integrated with various other libraries, such as NumPy, which excels in numerical operations. This collaboration allows for seamless execution of NumPy operations within OpenCV. With its vast array of projects and programs, the OpenCV Tutorial guides users from fundamental image processing techniques to advanced operations, including both image and video manipulation.

E. Keras

Keras, a high-level neural network library, is built upon the foundations of TensorFlow, CNTK, and Theano. Employing Keras for deep learning facilitates rapid and straightforward prototyping, with seamless compatibility for both CPU and GPU operations. Developed with Python, a debugging-friendly and robust programming language, this framework offers versatility and ease of use in neural network development.

F. TensorFlow

TensorFlow is an extensive open-source machine learning platform known for its comprehensive ecosystem of resources, frameworks, and tools. It provides high-level APIs that facilitate workflows, allowing users to select from a range of abstraction levels for creating and deploying machine learning models.

6. SYSTEM DESIGN

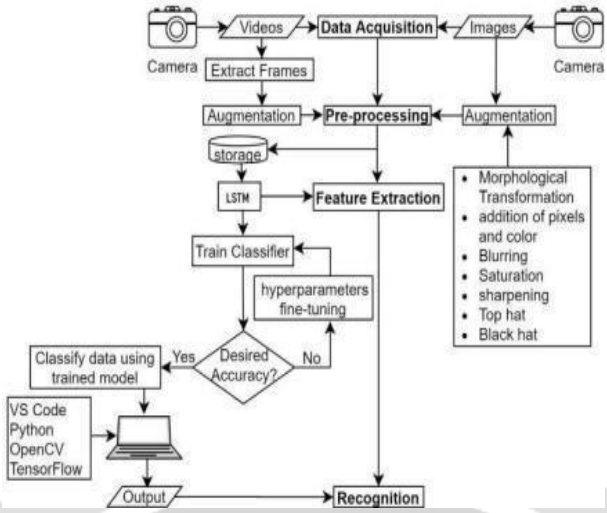

Fig-1: System Architecture

A. DATA FLOW

Initially, the image is analyzed by a classifier that takes into account preferences and factors in trajectory when motion occurs. Subsequently, keyframe extraction and feature analysis are conducted. Finally, results are generated and further classification based on hand shapes is performed.

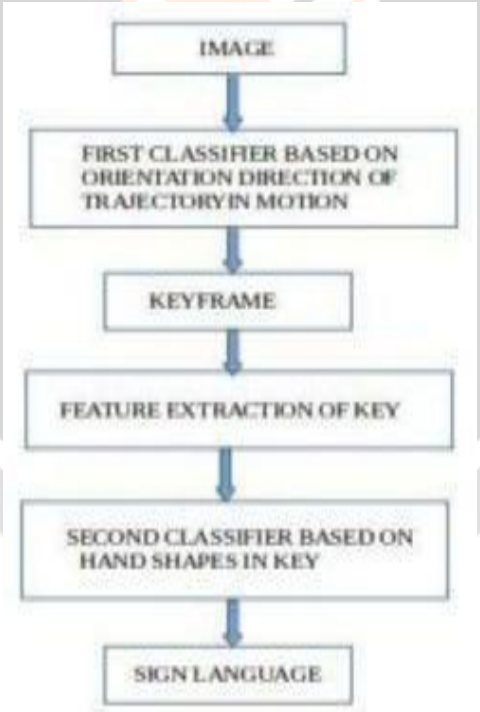

Fig-2: Data Flow Diagram

B. USE CASE DIAGRAM

In this scenario, two actors, the user and the system, are assigned various tasks. The user initiates the webcam, captures the motion in the live video stream, and receives the outcomes. Meanwhile, the system manages all other processes, including translating the gesture, extracting its features, comparing them to existing features, and executing hatching and gesture recognition. The user comprehends the gesture's significance once the results are presented.

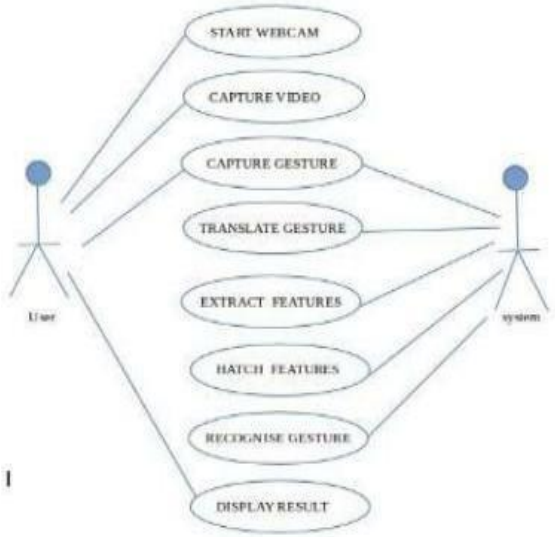

Fig-3: Use case diagram

C. SEQUENCE DIAGRAM

This illustrates the sequential interaction of multiple objects, depicting the order of events as they unfold. Referred to often in event diagrams or scenarios, this diagram showcases the interactions and sequential arrangement of the system's components. Presented below is the relevant sequence diagram for this project, comprising seven steps leading to the outcome. The initial step involves the user recording a video with a camera, followed by capturing a picture.

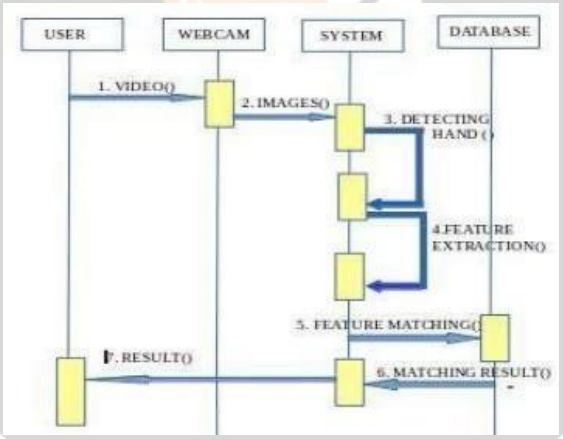

Fig-4: Sequence diagram

7. BASIC MODULES

A. Key points and extraction using MP Holistic Module

Utilizing the Media pipe Holistic module, key points are extracted effectively. This pipeline encompasses advanced components for face, hand, and pose tracking, facilitating holistic tracking. Consequently, the model can simultaneously recognize hand and body poses along with facial landmarks. Media Pipe proves invaluable for tasks such as face and hand detection and key point extraction, offering a comprehensive solution for feeding data into computer vision models.

Media Pipe, an open-source, cross-platform machine learning framework, empowers developers to construct comprehensive and multimodal applied machine learning pipelines. By leveraging Media Pipe, developers can focus more on model experimentation rather than system implementation details, as it efficiently handles model implementations across various platforms.

Utilizing the OpenCV framework, a feature is employed to access image input from the system webcam, along with hand and face landmark detection and key point extraction.

B. Collect Keypoint Values for Training and Testing Module

The Media Pipe Hand Land marker task facilitates the detection of hand landmarks within an image, utilizing functions like detect, detect_for_video, and detect_async. The process involves preprocessing the input data, detecting hands in the image, and identifying hand landmarks. To execute Hand Land marker in video or live mode, a timestamp for the input image is necessary. In live mode, the current thread remains unblocked, and results are promptly returned. After processing an input frame, the Hand Land marker task invokes its result listener with the detection outcome.

The output of HandLandmarkerResult comprises three arrays, each containing the results for one detected hand:

1. Handedness: indicating whether the detected hands are left or right-handed.
2. Landmarks: consisting of 21 landmarks with x, y, and z coordinates. The x and y values are normalized between 0.0 and 1.0 based on the image's width and height, while the z coordinate is relative to the wrist's depth. Smaller values denote landmarks closer to the camera, and the z scale is roughly equivalent to the x scale.

World Landmarks: similarly comprising 21 landmarks with actual 3D coordinates represented by x, y, and z values in meters, originating from the geometric center of the hand.

C. Build and Train LSTM Neural Network Module

Researchers have developed several sign languages
recognition (SLR) systems, yet they primarily excel at recognizing individual sign motions discretely. In this research, we introduce a continuous SLR model that identifies concatenated gestures by leveraging a modified long-short-term memory (LSTM) architecture tailored for continuous gesture sequences. The model operates by breaking down continuous gestures into smaller components and employing neural network modeling to analyze these segments. Consequently, training the model does not necessitate consideration of distinct combinations of sub-units.

D. Real-Time Detection Module

Using this approach, live hand gestures are interpreted into letters, then, words, and ultimately sentences. The process of detecting signs in real-time, with swift inference and a minimal margin of error, is referred to as real-time sign detection. According to one definition, a areal-time sign detection. According to one definition, a real-time sign system “manages an environment by receiving input data, processing it, and promptly delivering results to exert an immediate influence on the environment.”

8. IMPLEMENTATION

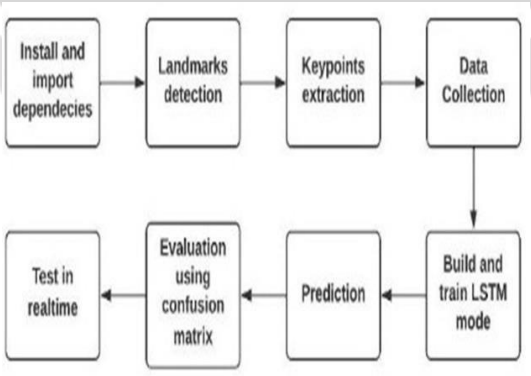

Fig-5: Methodology

We begin by gathering key points from media pipe holistic, accumulating ample data from various bodyparts, including hands, body, and face, stored as NumPy arrays, with each sequence comprising 30 frames; subsequently, we construct an LSTM model and train it on the stored data to identify actions depicted by a sequence of frames, adjusting the number of epochs to balance accuracy against runtime and model stability; upon training completion, the model is utilized for real-time hand gesture detection, with simultaneous conversion to text via OpenCV.

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 64)	442112
lstm_1 (LSTM)	(None, 30, 128)	98816
lstm_2 (LSTM)	(None, 64)	49408
dense (Dense)	(None, 64)	4160
dense_1 (Dense)	(None, 32)	2080
dense_2 (Dense)	(None, 3)	99

Total params: 596,675
Trainable params: 596,675
Non-trainable params: 0

Fig-6: Model Summary

9. RESULTS

The aim of this research was to employ forearm, hand, and finger kinematics models in conjunction with deep neural networks and Media pipe Holistic for signal prediction. The Media pipe LSTM, coupled with data augmentation, yielded the highest performance, achieving an average accuracy of 91.1% on test sets. This sign language detector is designed to understand signals, recognize and detect hand gestures, and generate corresponding coordinators. It willcontinuously update in real-time for all signs.

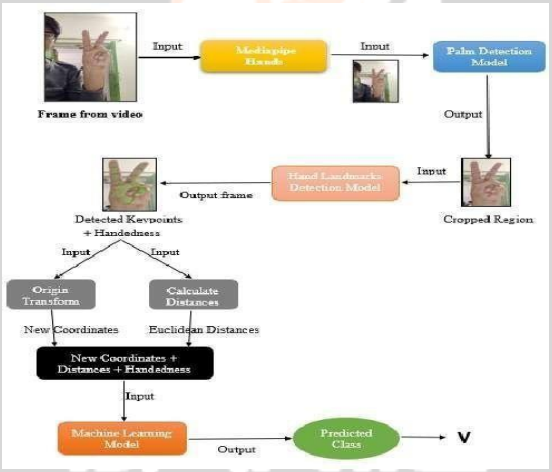

Fig-7: Real-time Workflow of Project

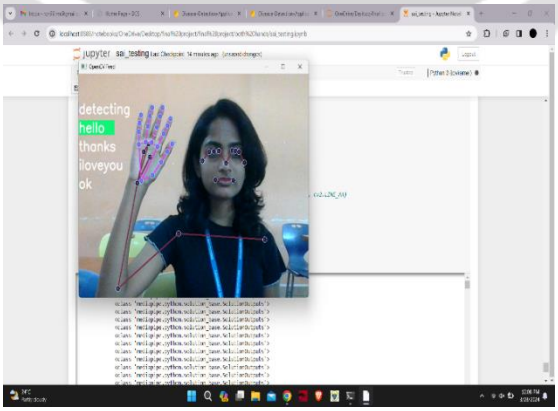

Fig-8: Detecting the sign for Hello

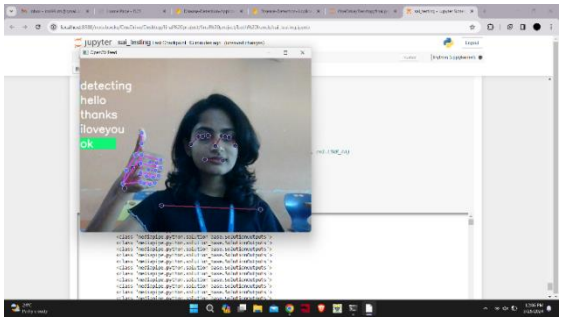

Fig-9: Detecting the sign for Ok

10. LIMITATIONS

While our "Sign Language Detection" project holds considerable promise, it also presents several limitations that warrant consideration. Firstly, the quality, size, and diversity of the training dataset significantly influence the performance of an LSTM model. Ensuring representationof all sign language styles, dialects, and situations while collecting a large and diverse set of sign language video data can pose challenges. A small and homogeneous dataset may lead to reduced generalization and accuracy of the model.

Additionally, sign language gestures exhibit a wide rangeof shapes, sizes, and complexities, each with its own variants. The subtle differences between comparable movements can pose difficulties for an LSTM model, potentially impacting its ability to accurately capture minute details. Addressing the diversity and subtleties of various sign language dialects, styles, and individual differences presents a complex task that may compromise accuracy.

Furthermore, achieving real-time processing for seamless communication demands efficient and prompt processing.

However, the computational demands of LSTM models may resultin latency issues. Ensuring accuracy and real-time performance, especially on resource-constrained devices like embedded systemsor smartphones, can be challenging.

Moreover, an LSTM model trained on a specific dataset may struggle to generalize to new users or environments not encountereddduring training. Variations in signing styles and environmental factors such as lighting and background clutter can interfere with gesture detection. Adaptations or adjustments may be necessary to optimize performance for new users and contexts.

Considering deployment in practical settings, accessibility and user interface design are critical factors. The system must be user- friendly, intuitive, and adaptable across various devices or platformsto ensure widespread applicability and acceptance. Addressing accessibility challenges for individuals with diverse technical knowledge levels and disabilities is essential for maximizing usability.

Despite these limitations, ongoing research, continuous improvement, and collaboration with sign language users and experts can help mitigate these challenges. Refining the system's accuracy, robustness, and usability in real-world settings is crucialfor enhancing its effectiveness and impact.

An LSTM model built on a particular dataset could have trouble generalising to new users or surroundings that weren'tencountered during training. Everybody signs differently, and environmental elements like lighting, camera angles, and background clutter can interfere with gesture detection. To ensure optimal performance, the system can need additional tweaking or adaption to new users and circumstances.

While the project has these limitations, they can be mitigated through ongoing research, continuous improvement, and collaboration with sign language users and experts to refine the system's accuracy, robustness, and usability in real-world settings.

11. CONCLUSION AND FUTUREENHANCEMENT

An LSTM model trained on a specific dataset may encounter difficulty in generalizing to unfamiliar users or environments not encountered during training. Variations in signing styles among individuals and environmental factors such as lighting, camera angles, and background clutter can disrupt gesture detection. Toachieve optimal performance, the system may require additional adjustments or adaptations to accommodate new users and

circumstances.

Our work aimed to develop an automatic real- time sign language gesture recognition system using various tools, but there is still room for further improvements.

Future studies could potentially develop a web or mobile application that can classify complete word symbols using facial emotions and relative hand movements from the face, which could be available on Android and Apple platforms. The future of sign language detection lies in the use of LSTM. With the increasing availability of sensors and wearable technology, there is growing interest in developing sign language recognition systems that can be used in real- world applications. For instance, such systems could be used to facilitate communication between deaf or hard of hearing individuals and those who do not know sign language.

12. REFERENCES:

[1] Machine Learning for Hand Gesture Recognition Using Bag-of-words Marouane Benmoussa, Abdelhak Mahmoudi, LIMIARF, EcoleNormale Superieure, Mohammed V University, Rabat, Morocco, 2018.

[2] Machine Learning-based Hand Sign Recognition Ms. Greeshma Pala, Ms. Jagruti Bhagwan Jethwani, Mr. Satish Shivaji Kumbhar, Ms. Shruti Dilip Patil, Department of Computer Engineering and Information Technology, College of Engineering Pune, Pune, 2021.

[3] Online Hand Gesture Recognition & Classification for Deaf & Dumb Nitesh S. Soni, Prof. Dr. M. S. Nagmode, Mr. R. D. Komati, Department of Electronics and Telecommunication, MIT College of Engineering, Pune, 2016.

[4] Sign and Human Action Detection Using Deep Learning Shivanarayna Dhulipala, Festus Fatai Adedoyin and Alessandro Bruno, Department of Computing and Informatics, Bournemouth University, Poole, 2022.

[5] Sign language Recognition Using Machine Learning Algorithm Radha S. Shirbhate, Vedant D. Shinde, Sanam A. Metkari, Pooja U. Borkar, Mayuri A. Khandge, JSPM's BSIOTR – Wagholi, Pune, IRJET, 2020