Sign Language to Text Generation Using CNN and LSTM

Mariyam Mirza[1], Subramanian K.M[2], Sridhar Gummalla[3]
[1]PG Scholar, Department of Computer Science and Engineering, Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086
Mirzamariyam2000@gmail.com
[2]Professor, Department of Computer Science and Engineering, Shadan College of Engineering and Technology,Hyderabad, Telangana, India – 500086. Email: kmsubbu.phd@gmail.com
[3]Professor, Department of Computer Science and Engineering, Shadan College of Engineering and Technology,Hyderabad, Telangana, India - 500086
Sridhar_gummalla@yahoo.com

ABSTRACT
Language barriers remain a significant challenge, particularly in the realm of sign language, which has not yet been fully addressed by translation technologies. This project aims to develop an end-to-end custom object detection system for real-time sign language translation. The system will utilize hand gesture recognition to detect, interpret, and translate sign language through advanced computer vision techniques. The core of the proposed solution involves a deep, multi-layered Convolutional Neural Network (CNN) designed to handle variations in hand gestures such as pose, orientation, location, and scale. The methodology includes capturing images using OpenCV and a webcam, annotating these images for object detection, training a TensorFlow model for sign language recognition, and implementing real-time gesture detection. Unlike traditional face detection methods, such as Haar-based classifiers that struggle with occlusions or variations in pose, the CNN-based approach offers greater flexibility and accuracy, benefiting from its ability to adapt through extensive training data.

I. INTRODUCTION

Addressing the significant communication barriers faced by the hearing-impaired community, this project focuses on the development of an innovative Sign Language Recognition System. Sign language, particularly American Sign Language (ASL), is a rich and expressive mode of communication that often encounters challenges dueto limited awareness and engagement from those without hearing impairments. This initiative seeks to bridge this gap by designing a user-friendly computer system capable of translating complex sign language gestures into clear, comprehensible text.

In contrast to existing research that often relies on specialized equipment such as gloves or Kinect sensors, our project adopts a more accessible approach by utilizing standard webcams to capture detailed images of ASL gestures. This method aims to make sign language recognition more practical and inclusive. By applying advanced computer vision and machine learning techniques, the system will precisely identify and classify the nuanced hand movements characteristic of sign language, ensuring accurate and efficient translation.

The project's core objective extends beyond gesture recognition; it emphasizes practicality by using readily available technologies, thereby facilitating seamless communication between the hearing-impaired and the broader community. This approach not only enhances the accessibility of sign language tools but also promotes widespread adoption, fostering better understanding and inclusivity.

Through the integration of cutting-edge computer vision and machine learning algorithms, the project aspires to achieve a breakthrough in the precise identification and classification of ASL gestures. This advancement has the potential to revolutionize communication for the hearing- impaired, transforming societal interactions by breaking down barriers and enhancing mutual understanding.

Ultimately, this project aims to merge technological innovation with a deep appreciation of the social and cultural dimensions of sign language. The anticipated outcome is not only a technically robust solution but also a significant social impact, advancing towards a more connected and empathetic global community where communication is effortless and inclusive for everyone.

II. RELATED WORK

A Convolutional Neural Network (CNN) is the most used technique for ASL recognition. Hsien-I Lin et al. employed image segmentation to isolate the hand from the image. They modeled the skin and adjusted the threshold to center the image around itsprimary axis. This processed image was then used totrain and predict outcomes with their CNN model.

Their algorithm was trained on seven hand gestures and achieved an accuracy of approximately 95% when applied to these movements. The authors utilized an American Sign Language (ASL) dataset to develop a real-time system that translates gestures into text. They captured images via a webcam, preprocessing the data by performing a hand gesture scan to amplify the gesture. The data was then fed into a pre-trained Keras CNN model, which generated a predicted label. Each gesture label had an associated probability, and the label with the highest probability was selected as the final prediction. This method achieved an overall accuracy of 95.8%. Garcia, B et al. created a real-time sign language translator aimed at improving communication between the deaf community and the general public. They used a pre-trained GoogLeNet architecture for classification, which successfully identified letters a-e with first-time users. Another real-time sign language translation technique was proposed by S. S Kumar [4], which utilized time series neural networks for sign language conversion. Several other techniques, such as Hidden Markov Models (HMM) and LSTM-RNN, have also been explored. V. N. T. Truong et al. combined the principles of AdaBoost and Haar-like classifiers. They utilized the American Sign Language (ASL) dataset, leveraging a large dataset to improve model accuracy. Specifically, they employed 28,000 positive images along with 11,100 negative samples to train and implement the translator. A camera captured the data, which was then processed by the model. Additionally, the use of HMM, or Hidden Markov Models, addressed the dynamic features of gestures. In this approach, skin color blobs corresponding to the hands were tracked within a body-facial space centered on the user's face, allowing gestures to be extracted from a sequence of video images. The main objective was to differentiate between deictic and symbolic movements.

The image is filtered using an indexing table for quick lookups. After filtering, pixels with similar skin tones are grouped into clusters called blobs. These blobs, identified based on the (x, y) coordinates and colorimetry (Y, U, V) of the skin-colored pixels, represent statistical objects that help detect homogeneous areas. Another approach used for ASL classification is LSTM-RNN. Additionally, the k-Nearest Neighbors (kNN) method has been proposed for recognizing 26 alphabets. Features extracted for the classification model include finger angles, sphere radius, and the distance between finger positions.

III. METHODOLOGIES

CAMERA STREAMING

Capture video: The camera is fixed at a specified distance to capture the video of the frontal image of a person.

Separated into frames from the video: The captured video needs to be converted into frames per second for easier detection.

POSE, FACE, HAND DETECTION USING MEDIAPIPE HOLISTICS

Face mesh: The ML pipeline consists of two real-time deep neural network models that work together. A detector that operates on the full image and computes face locations and a 3D face landmark model that operates on those locations and predicts the approximate 3D surface via regression.

Hand: A palm detection model that operates on the full image and returns an oriented hand bounding box. A hand landmark model that operates on the cropped image region defined by the palm detector and returns high-fidelity 3D hand key points.

Pose: It utilizes a two-step detector-tracker ML pipeline, proven to be effective in the Hand detection and Face Mesh detection. Using a detector, the pipeline first locates the person/pose region-of-interest (ROI) within the frame. The tracker subsequently predicts the pose landmarks and segmentation mask within the ROI using the ROI-cropped frame asinput.

EXTRACTING KEY POINTS

Hand Key point detection is the process of finding the joints on the fingers as well as the finger-tips in a given image. It is similar to finding key points on Face (a.k.a Facial Landmark Detection) or Body (a.k.a Human Body Pose Estimation), but, different from Hand Detection since in that case, the entire hand is treated as one object. They use key point detectors and multi-view images to come up with an improved detector. The detection architecture used is similar to the one used for body pose. The main source of improvement is the multi-view images for the labelled set of images.

SETUP THE DATABASE FOR COLLECTION

The tf.data API enables you to build complex input pipelines from simple, reusable pieces. For example, the pipeline for an image model might aggregate data from files in a distributed file system, apply random perturbations to each image, and merge randomly selected images into a batch for training.

COLLECT DATA FOR TRAINING AND TESTING

The training data builds up the machine learning algorithm. The data scientist feeds the algorithm input data, which corresponds to an expected output. The model evaluates the data repeatedly to learn more about the data's behavior and then adjusts itself to serve its intended purpose.

After the model is built, testing data once again validates that it can make accurate predictions. If training and validation data include labels to monitor performance metrics of the model, the testing data should be unlabeled. Test data provides a final, real-world check of an unseen dataset to confirm that the ML algorithm was trained effectively.

PREPROCESSING DATA

Data preprocessing involves transforming raw data to well-formed data sets so that data mining analytics can be applied. Raw data is often incomplete and has inconsistent formatting. The adequacy or inadequacy of data preparation has a direct correlation with the success of any project that involve data analytics. Preprocessing involves both data validation and data imputation. The goal of data validation is to assess whether the data in question is both complete and accurate. The goal of data imputation is to correct errors and input missing values - either manually or automatically through business process automation (BPA) programming.

BUILD AND TRAIN LONG SHORT TERM MEMORY (LSTM) NEURAL NETWORK

In order to train this LSTM, we be use TensorFlow's Keras API for Python. In these models, the input is a vector of features, and each subsequent layer is a set of "neurons". Each neuron performs an affine (linear) transformation to the previous layer's output, and then applies some non-linear function to that result. The output of a layer's neurons, a new vector, is fed to the next layer, and so on. A LSTM (Long Short-term Memory) Neural Network is just another kind of Artificial Neural Network, containing LSTM cells as neurons in some of its layers. Much like Convolutional Layers help a Neural Network learn about image features, LSTM cells help the Network learn about temporal data, something which other Machine Learning models traditionally struggled with. In order to train an LSTM Neural Network to generate text, first preprocess the text data so that it can be consumed by the network.

EVALUATION USING CONFUSION MATRIX AND ACCURACY

The popular Scikit-learn library in Python has a module called metrics that can be used to calculate the metrics in the confusion matrix.

For binary-class problems the confusion_matrix() function is used. Among its accepted parameters, we use these two:

1. y_true: The ground-truth labels.
2. y_pred: The predicted labels.

The function calculates the confusion matrix for each class and returns all the matrices. The order of the matrices match the order of the labels in the label's parameter.

TEST IN REALTIME

In the field of software testing, real time testing refers to the process of testing the real time software product or system i.e., the system is constrained by some specific time limits to perform and complete the job.

Generally, it involves, testing of the software product in its running/operational mode to evaluate its capability, to execute and finish a particular in a given stipulated time. The chief purpose behind the real time testing is to drastically minimize the probability of software's failure in real time environment at the user site. It is done to ensure that all the bugs and defects have been removed or fixed before its delivery to the client or user.

TECHNIQUE OR ALGORITHM USED

In this system, a combination of mediapipe holistics, convolutional neural network (CNN) and long short-term memory (LSTM) is used. The detection of facial hand and pose is made simpler with mediapipe holistics as it can skip the process of data augmentation and dedicate most of its capacity towards coordinate prediction accuracy. The convolutional neural network with LSTM helps with CNNs to learn visual features from video frame and LSTM to transform a sequence of image embedding into class label, sentences, probabilities, etc.

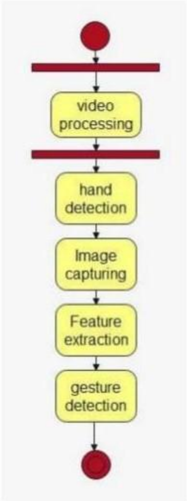

Fig. 1 State Diagram

IV. PROPOSED ALGORITHM

The proposed system for sign language detection represents a cutting-edge fusion of three influential technologies: Mediapipe Holistics, Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM). This strategic combination aims to create an integrated and efficient approach to simplify the intricate process of sign language detection. Each technology brings its unique strengths to the system, collectively enhancing its capabilities and streamlining the overall workflow.

Mediapipe Holistics:
Mediapipe Holistics serves as the foundational technology for the proposed system, providing a holistic approach to the detection of various components essential for sign language interpretation. This includes comprehensive tracking of facialexpressions, hand movements, and body poses, ensuring a nuanced understanding of the signer's gestures. The inclusion of Mediapipe Holistics contributes to the system's robustness and accuracy.Convolutional Neural Network (CNN):
The integration of Convolutional Neural Network (CNN) introduces advanced image processing capabilities to the proposed system. CNN is adept at recognizing patterns and features within images, making it instrumental in the extraction of key points from visual data. By incorporating CNN, the system gains the ability to efficiently analyze and interpret the intricate details of sign language gestures captured by the input data.

Long Short-Term Memory (LSTM):
The proposed system leverages the power of Long Short-Term Memory (LSTM), a type of recurrent neural network known for its proficiency in handling sequential data. In the context of sign language detection, LSTM plays a pivotal role in capturing the temporal aspects of gestures, allowing the system to understand the dynamic nature of sign language expressions. This ensures a more accurate interpretation of the signer's intent.

Unified Approach for Key Point Extraction:
A distinctive feature of the proposed system is its unified approach to key point extraction. Unlike traditional methods that necessitate separate coding and labeling for each detection module, this integrated system streamlines the process. By seamlessly combining Mediapipe Holistics, CNN, and LSTM, the system can extract key points from the input data with greater efficiency, reducing redundancy and simplifying the overall architecture.
Enhanced Efficiency and Reduced Complexity:
The integration of these three powerful technologies not only enhances the efficiency of the sign language detection process but also reduces the complexity of the system architecture. The unified approach ensures seamless communication between the different components, fostering a synergistic relationship that contributes to the overall effectiveness of the proposed system.
In summary, the proposed system stands at the forefront of innovation in sign language detection, leveraging the strengths of Mediapipe Holistics, CNN, and LSTM. This integrated approach not only simplifies the intricate process of key point extraction but also ensures a more efficient and accurate interpretation of sign language gestures, marking a significant advancement in technology for inclusive communication.

V. CONCLUSION AND FUTURE ENHANCEMENT

In conclusion, this project has taken a significant step towards bridging the communication gap for the deaf and hard-of-hearing community. The developed Sign Language Recognition (SLR) system, built upon a Deep Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) network, achieved a commendable validation accuracy of 95%. This accomplishment highlights the effectiveness of deep learning architectures in creating robust SLR systems. However, our work also paves the way for exciting future advancements. The current accuracy, while impressive, can be further enhanced by incorporating larger and more diverse datasets. These datasets should encompass regional variations in sign language, individual signing styles, and even facial expressions and body language, which play a crucial role in conveying meaning. Additionally, exploring even more sophisticated deep learning models, such as those incorporating 3D hand pose estimation and multimodal learning, could lead to a deeper understanding of sign language communication.

The ultimate objective lies not just in recognizing individual signs, but in achieving true semantic understanding – grasping the meaning behind the gestures. By delving into syntactic analysis of sign language grammar, future systems could provide even more accurate and natural-soundingtranslations.
The impact of this technology extends far beyond basic communication. Imagine a world where SLR is seamlessly integrated into daily life. Augmented reality glasses displaying real-time captions or translations during face-to-face interactions, orsmart homes responding to sign language gestures, would

revolutionize how deaf and hard-of-hearing individuals interact with the world around them.

In essence, this project serves as a stepping stone towards a future of inclusivity and accessibility. The potential for SLR technology to empower the deaf and hard-of-hearing community is vast, and continued research and development hold the key to unlocking its full potential.

The drawback is that not everyone possesses the knowledge of sign languages which limits communication. This limitation can be overcome using automated Sign Language Recognition systems which will be able to easily translate the sign language gestures into commonly spoken language.

In the future, the dataset can be enlarged so that the system can recognize more gestures. The TensorFlow model that has been used can be interchanged with another model as well. The system can be implemented for different sign languages bychanging the dataset

1. Larger and More Diverse Datasets:

Dialect and Sign Variations: Today's systems might struggle with regional variations or sign language dialects. Imagine datasets that capture these variations, allowing the system to understand the nuances of Kenyan Sign Language compared to American Sign Language.

Signer Specific Recognition: Incorporating data from a specific signer could allow the system to recognize their unique signing style, including subtlevariations in hand position or movement. This personalization would enhance accuracy for frequent users.

Facial Expressions and Body Language: Sign language goes beyond hand gestures. Datasets that include facial expressions and body language will allow the system to capture the full context of communication.

2. Advanced Deep Learning Models:

3D Hand Pose Estimation: Current systems often rely on 2D images or videos. Imagine 3D models that can capture the depth and spatial orientation of hands, leading to more precise recognition, especially for complex signs.

Multimodal Learning: Combining deep learning models for hand gestures with those for facial expressions and body language would create a holistic understanding of sign language communication.

3. Real-time Translation and Captioning:

Speaker Identification: Imagine a system that recognizes not just the signs but also the speaker. This would allow for personalized captioning or translation styles, catering to individual preferences.

Emotional Nuance: Sign language conveys emotions too! Advanced systems might analyze facial expressions and integrate them into the translation, providing a more complete picture of the message.

4. Integration with Devices and Platforms:

Augmented Reality (AR) Glasses: Imagine AR glasses that display real-time captions or translations overlaid on the signer. This would be a game-changer for face-to-face conversations.

Smart Home Integration: Sign language recognition integrated into smart home devices could allow deaf users to control lights, thermostats, or appliances through gestures.

5. Sign Language Understanding:

Semantic Understanding: The goal is for systems to grasp the meaning behind the signs, not just the physical gestures. This would allow for more accurate and natural-sounding translations.

Syntactic Analysis: Sign languages have their own grammar and syntax. Advanced systems might analyze the order and structure of signs to provide a deeper understanding of the message.

These are just a few examples, and the possibilities are constantly evolving. As technology advances, sign language recognition systems have the potential to revolutionize communication for deaf and hard-of-hearing individuals, fostering a more inclusive and connected world.

REFERENCES

[1]. Hsien-I Lin, Ming-Hsiang Hsu, Wei-Kai Chen, “Human Hand gesture recognition using a convolution neural network”, 10.1109/CoASE.2014.6899454, August 2014

[2]. J. Zhang, W. Zhou, C. Xie, J. Pu and H. Li, \"Chinese sign language recognition with adaptive HMM,\" 2016 IEEE International Conference on Multimedia and Expo (ICME), Seattle, WA, USA, 2016, pp. 1-6, doi: 10.1109/ICME.2016.7552950.

[3]. V. N. T. Truong, C. Yang and Q. Tran, ”A translator for American sign language to text and speech,” 2016 IEEE 5th Global Conference on Consumer Electronics, 2016, pp. 1-2.

[4]. Kohsheen Tiku, Jayshree Maloo, Aishwarya Ramesh, Indra R, “Real-time Conversion of Sign Language to Text and Speech,” 2020 Second International Conference on Inventive Research in Computing Applications, Coimbatore, India, 2020, pp. 346-351.

[5]. Machine Learning Techniques for Indian Sign Language Recognition, International Conference on Current Trends in Computer, Electrical, Electronics and Communication (ICCTCEEC-2017) - Kusumika Krori Dutta,

Sunny Arokia Swamy Bellary

[6]. J. R. Pansare, S. H. Gawande and M. Ingle, “Real-Time Static Hand Gesture Recognition for American Sign Language in Complex Background,” Journal of Signal and Information Processing, No. 3.pp. 364-367

[7]. Indian Sign Language Recognition System for Deaf People " Int. J.Adv. - A. Thorat, V. Satpute, A.
Nehe, T.Atre Y.Ngargoje

[8]. Dhivyasri S, Krishnaa Hari K B, Akash M, SonaM, Divyapriya S, Dr. Krishnaveni V An Efficient Approach for Interpretation of Indian Sign Language using Machine Learning , 2021 3rd International Conference on Signal Processing and Communication | 13 – 14 May 2021 | Coimbatore

[9]. R. San Segundo, B. Gallo, J. M. Lucas, R. Barra-Chicote, L. D’Haro and F. Fernandez, ”Speech into Sign Language Statistical Translation System for Deaf People,” in IEEE Latin AmericaTransactions, vol. 7, no. 3, pp.400-404, July 2009, doi:10.1109/TLA.2009.5336641.