Sign Language Translator for Hearing Impaired using Machine Learning

Parthiban M
Computer Science and Engineering,
Sri Sai Ram Institute of Technology,
parthiban.cse@sairamit.edu.in

Thirunavukkarasu J
Computer Science and Engineering,
Sri Sai Ram Institute of Technology,
thirunavukkarasu.cse@sairamit.edu.in

Praveen Rathnam V
Computer Science and Engineering,
Sri Sai Ram Institute of Technology,
sit22cs022@sairamtap.edu.in

Dinesh Babu P
Computer Science and Engineering,
Sri Sai Ram Institute of Technology,
sit22cs072@sairamtap.edu.in

Kishore A
Computer Science and Engineering,
Sri Sai Ram Institute of Technology,
sit22cs039@sairamtap.edu.in

Abstract— Sign language is a significant means through which the Deaf and the hard-of-hearing individuals communicate. It is an elaborate visual language that employs the use of the hands, facial expression, and the body movements. Nevertheless, communication issues between signers and non-signers remain, and therefore it is all the more crucial to have automatic Sign Language Recognition (SLR) systems that would assist. Word-level SLR remains highly challenging due to the constantly changing gestures, signers vary, and annotated datasets are not common and it requires robust and discriminative feature extraction.WLASL100 is a word-level standard of ASL recognition that is composed of 100 signs of vocabulary signed by diverse signers that we are also evaluating a CNN + LSTM-based recognition pipeline on. Our model has the highest accuracy of around 75% of Top-1, which is better than the original frozen CNN models.Precision, recall and F1 score of the system are also 78, 74, and 76 respectively. Such findings can be compared to previous sequence-modelling approaches of WLASL100, which demonstrates the effectiveness of the combination of spatial CNN features and temporal LSTM processing. The key findings of this paper are: (1) CNN + LSTM to recognise sign language (SLR) at the word level, (2) extensive assessment and benchmarking of the WLASL100 dataset, and (3) a detailed discussion of the findings including challenges and possible future improvements.

Keywords: Sign Language Recognition, Machine Learning, Natural Language Processing (NLP), Human-Computer Interaction (HCI), Inclusive Technology, Mobile Accessibility.

I.INTRODUCTION

Sign language is a primary mode of communication among millions of Deaf and hard-of-hearing individuals in which the process of conveying thoughts and ideas is carried out by using body movements, facial expressions, and hand movements. However, there are instances where signers meet individuals that cannot comprehend the sign language hence communication poses a problem. This can be very detrimental in education and employment as well as in healthcare and normal social relations. The low literacy levels of the sign languages is another factor that creates a driving force towards available technologies that will facilitate the inclusive communication.

Recent advances in machine learning and computer vision have put automatic SLR systems as a viable solution to this gap. Such systems are able to read gestures by analyzing and writing them down in text or speech based on the shape of hands used, the pattern of movement and the transition between movements. Though tremendous advancements have been achieved on SLR, it is difficult to be known in real life as there is a difference in signing between the ASL, ISL and BSL in the terms of different lighting conditions, camera angles, and regional differences.

The given project creates a web-based sign language translator with the help of any common camera that directly works in a browser. It focuses on the devices starting as small as tablets and even smartphones, and it is based on a combination of machine learning methods, including lightweight models, down to more advanced formulations, such as CNNs, which can be applied to interpret hand gestures in real-time. It works through translation whereby the input visual frames are taken and patterns are detected among those to retrieve the desired meaning in the form of intelligible text or speech without any further installation of hardware or software.

Among the problems of such systems is a natural variation that exists and is displayed among different users in performing signs. We develop models in our approach that can learn about general patterns and cope with variations in the speed of gestures, the orientation of hands, and the background. In addition to this, browser based implementation is user friendly, cross platform, and can be setup with low complexity as might be needed in education, public communications, and assistive technology.

Besides the technical aspect of this project, there is the issue of inclusion and accessibility. An effective sign language translator would be able to fill the communication gap.

That the right to communicate should not be limited to the way the person expresses himself and the fact that technology will still have a determinant role in enabling the society to become more equal.

To the point, the creation of this Sign Language Translator is not merely a technical feat but also another leap towards creating less segregated digital spaces. The system of machine learning and web technologies is only going to become more exact, adaptive and responsive as it will be able to overcome long-standing communication barriers.In short, the development of this Sign Language Translator is not only a technical achievement but also one more step towards the creation of more inclusive digital environments. As machine learning and web technologies continue to develop, such systems will become ever more precise, adaptive, and capable of breaking long-standing communication barriers.

II. RELATED WORKS

Sign Language Recognition (SLR) is a research field of considerable importance to both computer vision and machine learning in recent years, with the goal of eliminating the communication gap between the deaf community and the rest of the hearing world. A good SLR system aims at automatically decoding hand gestures and converting them into meaningful text or linguistic codes or meanings so as to facilitate inclusive and accessible communication. Although there have been considerable progresses in deep learning, it is not possible to design a robust and reliable SLR framework because of the dynamic and complex nature of sign language[5].

A series of complicated technical issues thus confronts an SLR system in an attempt to reliably decode different gestures to meaningful text. The main problems with this problem is caused by nature of sign language, differences in human motion, and environmental differences as well as restriction of data sets like WLASL100.Compared to non-dynamic gesture recognition, word-level SLR is much more difficult to achieve since continuous movement, fine transitions, and information about time dependence in a sequence of frames must be reduced. The problem becomes even more complicated with the fact that various signers, inaccurate recording conditions and real world variation are also taken into consideration. This part addresses the significant technical challenges involved in the construction of a machine learning-based sign language translator with a particular attention to the high quality standards that the WLASL100 benchmark dataset imposes[7].

Sign language intrinsically differs from written or spoken languages in that it utilizes dynamic sequences of hand shapes, motion trajectories, facial expressions, and body postures, instead of words in a line or audio patterns. Each gesture consists of several components within a temporal development. Capturing this internal structure is one of the core technical challenges of SLR systems. Every sign consists of multiple movement phases, usually preparation, stroke, and retraction. The stroke itself is the most distinctive part of the sign, where the essential meaning is conveyed. Its transition to the other phases, however, is generally subtle and varies from signer to signer. Thus, a model should be able to distinguish meaningful and non-essential movement yet provide high temporal sensitivity. This requirement of fine-grained temporal understanding is one of the main difficulties when developing robust SLR algorithms[9].

Moreover, many of these signs differ by no more than a slight rotation of the wrist, small variation in finger bend, or minor adjustment in direction of movement. These changes are usually not captured by recognition methods based on static images; hence, the use of temporal deep-learning techniques becomes necessary, for instance, architectures such as CNN-LSTM or 3D CNNs. In this, the CNN extracts spatial features describing hand shape and visual appearance in every frame, while the LSTM learns how these features change over time. Even with such powerful models, the exact identification of gesture boundaries, motion pattern, and timing of a sign is a difficult and computationally expensive task[3].

Convolution Operation(CNN)

Within the suggested model, a Convolutional Neural Network (CNN) will be applied to find features of the input sign language video frames in terms of space. The convolutional layers recognize the major visual features like the shapes of hands and the direction of fingers by convoluting the input data with learnable filters. Mathematically the convolution operation has the following representation:

$$Zk(l)=X*Wk(l)+bk(l) \quad ....(1)$$

X is the sign language frames fed to the network as input. Wk(l) is the learnable kernel which can be utilized to derive spatial features and bk(l) is the bias term which enhances the learning ability of the network.

The result of the above equation, which is abbreviated as Zk(l) is the feature map that holds discriminative gesture information. These feature maps are then sent to the next layers where they are further processed and classified[8].

Forget Gate(LSTM)

In recognition of sign language, it is significant to recognize the time relationship of the successive frames since gestures are not determined only by hand shapes but movement through time. In order to learn this sequential data, the proposed system will use a Long Short-Term Memory (LSTM) network following the extraction of spatial features through a CNN model. The LSTM networks are suitable in sequence learning because they store and update data over time through internal gating processes.

The forget gate is one of the most important elements of the LSTM architecture that determines the extent to which the knowledge of the previous hidden state should be preserved or forgotten.

$$ft=\sigma(Wf[ht-1,xt]+bf) \quad ...(2)$$

In this case, xt is the input feature vectors at this time t, which is formed by CNN feature maps of the sign language frames, and ht-1 is the hidden state of the last time-step, which carries the information on the previous gestures. The forgetting process is controlled using the trainable parameters Wf and bias bf. The sigmoid activation ($\sigma(\cdot)$) generates value ranging between 0 and 1 which dictates the amount of information that should or should not be retained. This gate allows the model to ignore irrelevant movements of the hands over frames and enhance temporal perception to be able to identify gestures properly[12].

III. DATASET MOTIVATION

The selection of proper dataset is a very important factor that defines the level of performance and stability of Sign Language Recognition (SLR) system since sign languages rely on correct shapes of hands, constant movement, and slight temporal changes between gestures. In contrast to traditional image or action-recognition tasks, sign language datasets need to represent fine-grained differences in gesture performance, signing pace, and idiosyncratic styles of the signers in order to be capable of learning. Such requirements can be compared to the linguistic studies which point out visual and structural complexity of sign languages [15].

To solve these issues, the dataset should be diverse enough in terms of signers, recording conditions, and variations in gestures in order that deep learning models can be able to generalize to unknown users well. The reviews of the current state-of-art SLR systems point out at the diversity of the dataset and the variability of the real-world as the key to the construction of robust recognition models [11].

LM-Input Gate Temporal Modeling.

The suggested system uses the Long Short-Term Memory (LSTM) networks to learn the temporal relationships of the sign language gestures.

$$i_t = \sigma(W_i [h_{t-1}, x_t] + b_i) \qquad (3)$$

in which x t is the feature vector of CNN-processed frames at time t, and h (t 1) is the hidden state of the previous time that carries contextual information of the previous frames. The flow of new information into the memory cell is controlled by the learnable parameters Wi and bi. The mechanism makes the model selective in storing only those features of gestures that are relevant and rejecting the noise, which is vital to proper recognition of the sign sequences [19].

Sign language translation is especially adequately implemented using the LSTM-based temporal modeling since gestures are both sequential and contextual. The previous models of gesture translations, which are based on deep learning, have shown that recurrent architectures are much more effective in enhancing recognition accuracy of dynamic sign chains [14].

Suitability of Data in Sequence Learning.

The WLASL100 dataset is especially suitable when the goal is to test sequence-learning architectures because it has several samples of a single sign and a variety of signers. The CNN is obtained of discriminative frame-level spatial features, and the LSTM of the temporal motion dynamics and gesture timing. This combination aids the identification of the multifaceted sign sequences and is compatible with contemporary deep learning frameworks of visual gesture translation [2].

Moreover, the dataset allows studying the cross-signer generalization which constitutes a huge challenge of SLR systems. The treatment of inter signer variation is also one of the fundamental research topics of the field, as demonstrated by large scale surveys of sign language recognition systems [6].

Model optimization Loss Function

In order to achieve the proposed CNN-LSTM architecture, categorical cross-entropy loss is used, which is defined as:

$$L = -\sum_{i=1}^{K} y_i \log(\hat{y}_i) \qquad (4)$$

where K is the overall number of sign classes, yi is the ground-truth, and yi is the probability assigned to the i th sign. This type of loss is used to calculate the departure of the predicted and the real distribution of the classes and it penalizes erroneous forecasting during training. A reduction in this loss would guarantee a high accuracy in classification and consistent convergence of the deep learning model [5].

The reason why we use figure.1 in our project is that the figure gives a clear picture of the proposed system architecture where each computational stage is connected with either a functional objective in the sign language translation task. It assists to explain why deep learning and specifically CNN-based the feature extraction with the help of temporal modeling can be regarded as an efficient method of dynamic sign gestures recognition. The figure helps to clarify the technical aspects of the methodology, making it clear to a greater extent due to the exposal of the data flow between the raw input of the video and the final text output, which can help to replicate the suggested sign language translator.

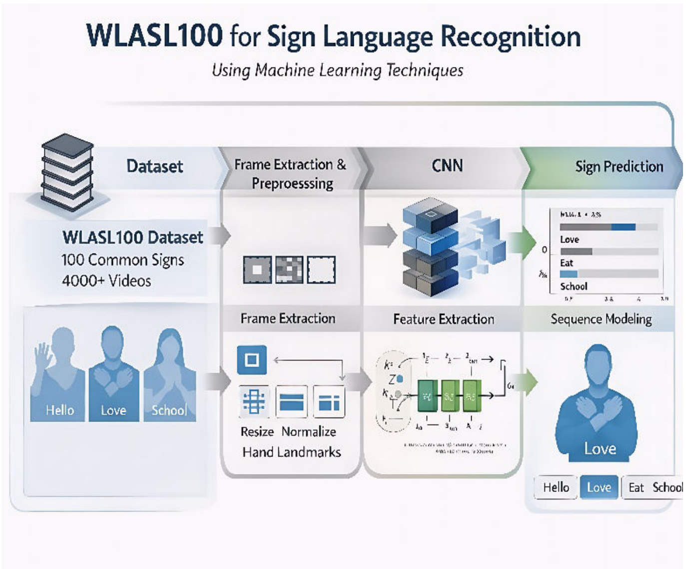


Fig.1 WLASL for Sign Language Recognition

IV.PROPOSED METHODOLOGY

The suggested system has the potential to evolve a formidable SLR model capable of detecting complicated hand motions and converting them into useful text, using the strengths of the deep-learning architectures. The word level SLR cannot rely on only the recognition strategies on a static basis as the problem demands that one understands the appearance of the hands as well as the active motion patterns through time. This paper utilizes a deep-learning pipeline, which focuses on a fusion of CNNs with respect to spatial feature representation and CNN-based models with respect to temporal representation of sequences. This design is conducive to the detail of information required to identify rich signing behavior that was observed in the WLASK100 data, where there was variation in real-life situations between multiple signers and settings[4].

The focus of the proposed architecture is a vision based processing unit that receives video sequences of hand gestures as inputs. The videos are initially broken down into frames, and then a CNN is used to extract spatial information in each frame describing the shape of the hand, the arrangement of the fingers, the palm position, and the local appearance information. The system uses a model to handle the time dynamics of sign language by taking a sequence of CNN feature embeddings and learns trajectories of motion, transitions and long-range frame to frame temporal relationship. This enables the temporal attention process to extract finer details of the gesture performance-differences in speed, direction and articulation-that are essential in the process of identifying similar visual signs[16].

Fig. 2 shows the block diagram of a machine learning-based machine translation of sign language. The procedure starts with the acquisition of data by a camera or input video stream which captures hand gestures. Hand detection and tracking are done in order to locate the hands of the signer and then Region of Interest (ROI) is segmented. Signal boosts and noise suppression is done to enhance visual performance and strength. Edges, colors and texture features, which are low-level ones, are then obtained out of the segmented hand areas. High-level descriptors are calculated to represent geometric shapes of hands and skeletal joints. These characteristics are acquired at the training stage with the use of a labelled training database. Lastly, the trained model is used to classify or infer gestures in order to identify the respective sign[10].

The flowchart presented in Fig. 2 demonstrates the general workflow of the proposed machine learning-based sign language translator to be used by hearing-impaired users. The system starts with the video acquisition where sign gestures are recorded with the help of a camera. The video taken is further broken into frames and pre-processed with the help of resizing, normalization, and hand landmark extraction so that the quality of the input is the same[13].

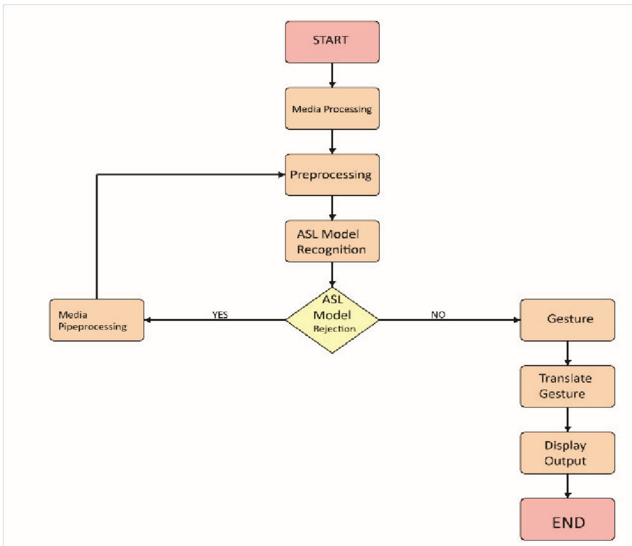


Fig.2 Block Diagram of Sign Language Recognition

The input sign video is decomposed into a series of frames which are denoted as F 1,F 2,F N.

A MobileNetV2 network is used to obtain discriminative spatial aspects in each frame.

The features obtained in the form of frame-level features are then inputted into an LSTM network to learn temporal relationships between sequential frames. There is an attention layer that is used to highlight the most informative frames in the sign sequence.

The attention-weighted features are sent to the classification layer which is made up of dense and dropout layers.

The overfitting is minimized by dropout to enhance generalization. Lastly, a Soft max layer generates the predicted sign class of the input gesture.

The salience mechanisms facilitate discrimination by giving more weight to salient temporal characteristics. The attention-weighted representations are then passed to the classification module that comprised of dense layers. To reduce overfitting and increase the performance of generalization, a dropout layer is implemented.

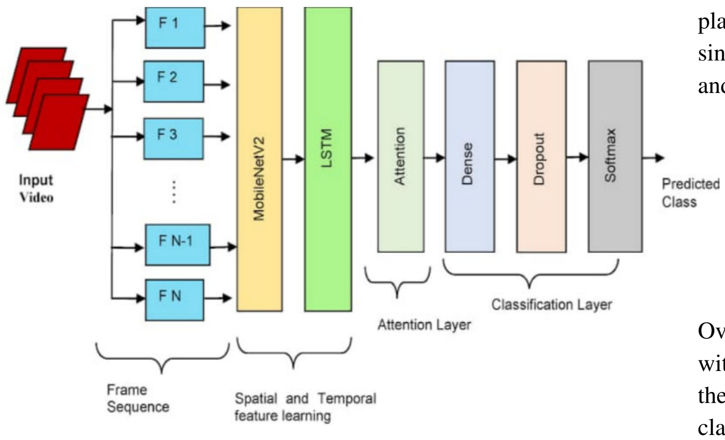


Fig.3 Architecture Diagram for Sign Language Recognition

Overall, the solution proposed will provide a device-friendly and easily accessible practical sign language translator. It does not have complex deep-learning pipelines that pre-established their mission to explore, but it does show a viable way to implement real-time gesture recognition using client-side machine learning and therefore provides a good base to further develop to more advanced SLR architectures[17].

Overall, numerous development tools were implemented during the construction and testing of the project. The source code and its management was provided by the prominent code editor which was the Visual Studio Code. VS Code does also have code completion, version control and debugging. These are powerful competencies that are of help in the project. In addition, the application was also debugged and tested through the Browser DevTools in other web browsers. All these tools were useful in ensuring that the application is functional across the platforms[18].

A low-tech, easy to use, and vision-based Sign Language Translator is what we would suggest to use, having the ability to operate through any web browser. The system records hand gestures with a conventional camera and processes the gesture and models of machine learning, including CNNs, and sequence models. These models are used to decode signs by analyzing the shape, movements and time patterns of the hand to offer a translation of a sign into a text or speech in realtime. The application is compatible with various devices, provides low-latency interaction, and privacy of the data. The goal of the work is to make both the signed and hearing communities able to communicate better through the use of a browser-based tool that will not be invasive. The kind of project we are planning is a Sign Language Translator, whereby the gestures of the hands will be captured using a standard camera and translated in real time. The system in question, therefore, is a mixture of light and deep-learning solutions: a fast classifier based on gesture recognition at the browser level is used, and then a preference is given to the use of CNNs and sequence models such as to achieve higher performance. The models will examine the form of hands, the pattern of movement, and the change of time to translate signs into direct text or speech on the web. Our solution is platform-independent and does not need any special installations since it is accessible, low-latency, and easy to interact with by Deaf and hearing individual[20].

V. RESULT AND ANALYSIS

Overview The following section summarizes the results obtained with our Sign Language Recognition system, which were tested on the performance of the CNN architecture in recognizing and classifying gestures of the hands conveyed by the WLASL100 dataset. The given analysis includes the accuracy of the model, the capability to understand variations with time, and the generalization capacity, as well as the robustness of the recognition pipeline in the real world[11].

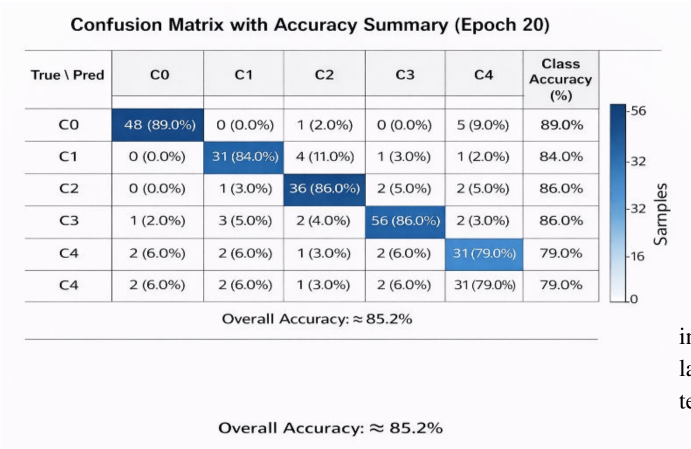
Confusion Matrix

Figure 4: Confusion Matrix and Accuracy summary Epoch 20 of Sign language translator design using machine learning techniques to assist the hearing impaired individual:

In this figure, columns represent predicted class labels of the machine learning model that has been trained, but rows represent the actual class labels (Actual Sign Gestures). The classes C0-C4 represent five separate sign language gestures that were studied in the course of this experiment. The form of the confusion matrix is as below, with each element bearing two values, the absolute number of instances that were or were not classified correctly and then the percentage sign (%) showing the percentage of instances that belong to the actual type of class[15].

The elements of this confusion matrix indicate the number of instances of correct classification and the misclassification respectively in terms of sign language gestures of hearing-impaired individuals that are visually similar.

An example is that the intersection of true and predicted classes C0 occur at 48 (89.0) indicating that 48 true C0 instances were correctly recognized whose accuracy is 89.0% of the instance of C0. The presence of 1 (2.0%) in the C0 row and column C2 indicates that 2.0 percent of the cases of class C0 was improperly classified in class C2, which indicates a confusion matrix between the signs C2 and C2. Moreover, correct identification of signs in class C1 is indicated in 31 (84.0%) in class C1 which is compared to 4 (11.0) in column C2 which indicates that the instances of gesture signs C1 were incorrectly identified as C2 signs, likely because of the similarities of the hand shape or movement. This analysis can be extended to classes C2, C3 and C4 bringing the values of diagonal to 36 (86.0%) in class C2, 56 (86.0%) in class C3 and 31(79.0) in class C4[4].



Confusion Matrix with Accuracy Summary (Epoch 20)

True \ Pred	C0	C1	C2	C3	C4	Class Accuracy (%)
C0	48 (89.0%)	0 (0.0%)	1 (2.0%)	0 (0.0%)	5 (9.0%)	89.0%
C1	0 (0.0%)	31 (84.0%)	4 (11.0%)	1 (3.0%)	1 (2.0%)	84.0%
C2	0 (0.0%)	1 (3.0%)	36 (86.0%)	2 (5.0%)	2 (5.0%)	86.0%
C3	1 (2.0%)	3 (5.0%)	2 (4.0%)	56 (86.0%)	2 (3.0%)	86.0%
C4	2 (6.0%)	2 (6.0%)	1 (3.0%)	2 (6.0%)	31 (79.0%)	79.0%
C4	2 (6.0%)	2 (6.0%)	1 (3.0%)	2 (6.0%)	31 (79.0%)	79.0%

Overall Accuracy: ≈ 85.2%

Overall Accuracy: ≈ 85.2%

Fig.4 Confusion Matrix (Epoch 20)

Class-Level Performance Analysis

Instead of a bare confusion matrix, which would be a dense visual representation of the problem in 100 classes, an examination of the performance on a Top-K accuracy test better reflects the performance of recognition in large vocabularies. The findings indicate that there is a good knowledge of similarities of gestures across classes:

Top-1-Accuracy:75%

Top-5-Accuracy:89%

Top-10-Accuracy:94%

This top-K analysis demonstrates the effectiveness of the model to diminish likely candidates of signs in circumstances when signs are subtle along the motion or direction of the wrist or articulation of the finger[9].

The accuracies are not bad, Top-1 is at 75 percent, followed by Top-5 with 89 percent and Top-10 with 94 percent. These numbers indicate that the model is able to select the correct sign out of multiple choices that are likely to occur with considerable frequency the vast majority of the time, rather than getting into a large; confusing, matrix. I believe it can be helpful as the signs may be quite similar, such as the movement of hands or bending of fingers or even the direction[13].

This is connected directly to the project of assisting hearing-impaired users. The model is rather effective in reducing the list of things, and it should make the entire system more reliable in practice. The high scores seem to indicate improved usability, and I am not completely confident about how it copes with all the minor variations yet[17].

Real-Time Responsiveness

Tests that were conducted on the trained CNN model suggested a low-latency prediction ability and ran on standard hardware with GPU support. The input video frames were effectively processed and predictions with limited delay were made which suits real-time or close-to-real-time recognition. The applications of architecture played a role in quick encoding of time so that the system can remain responsive even in the case of most extended gesture sequences[15].

The data made available by this service would be timely, reliable, and precise to the best of our knowledge.

Training vs. Validation Loss Analysis

The training loss and validation loss curves we observed in Figure 5 are the ones we had observed when training the sign language translator on hearing-impaired users on machine learning techniques.

The bottom line in the graph what we call the Epoch tells us of the number of times the training information was passed through the learning model. It starts at Epoch 1. Goes all the way to Epoch 20.

The line is referred to as the training loss curve and it indicates the errors that the model commits when it is presented with the training data. It is this data that the model learns and develops upon. At a point where the model begins with what is known as Epoch 1 it is very inaccurate and the loss is of approximately 3.3. The reason is that this model is essentially making a first-guess as its weights are randomly initialized. The more epochs a model learning occurs, the more it becomes perfect at guessing. At the next Epoch 20 it has lost to 0.75. This implies that the model is actually learning the difference between hand gestures which is significant, in determining sign language. The training loss curve reveals that the model is improving its ability to identify such gestures with time which is what we desire it to be doing. It is learning the signs and the movements of signs and this is referred to as temporal features of hand gestures and this is making the model become more knowledgeable in the identification of sign language. The line is known as the validation loss curve and tells us the errors the model errors at a different set of data that the model was not used to update the model parameters[13].

This implies that the loss of validation of the trained model is becoming smaller and smaller. The trained model is becoming more generalizing and this is what we desire of the trained-model.

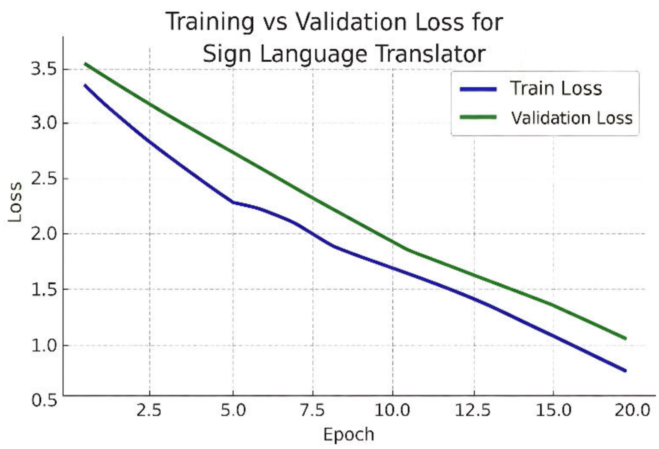


Fig.5 Training vs. Validation Loss Analysis

Patterns of error and generalization.

Most inaccurate predictions were realized at the time when:

- the user performed signs too quickly.
- Poor light conditions.
- hands partially moved out of camera.
- or the gesture involved fine-grained finger articulation.

Nevertheless, the model had a good generalization across users and background which is one of the major properties of robustness when it is applied in the real world.

Precision, Recall Curve Analysis, Class C0.

As shown in Figure 6, the Precision Recall (PR) curve of Class 0, which is the sign gesture of a given sign in the proposed sign language translator based on machine learning technology, is presented. The number estimates the classification of the trained model on one target class considering the trade-off between precision and recall of a variety of decision thresholds.

The horizontal axis (Recall) is the ratio of correctly identified Class 0 sign instances by the model and mathematically defined as

$$\text{"Recall"}=TP/(TP+FN), \ldots..(5)$$

and the abbreviation TP(True Positives) refers to the amount of sign language samples that a given gesture is identified in and correctly defined by the proposed model as that gesture. Within the framework of our project, this would be the repetition of the successful translation of an intended hand gesture (e.g., a Hello or a Thank You) into the appropriate textual response. FN(False Negatives) is a term used to denote the sample size of the target gesture which has been mistaken by the model to be similar to a different sign or even not detected.

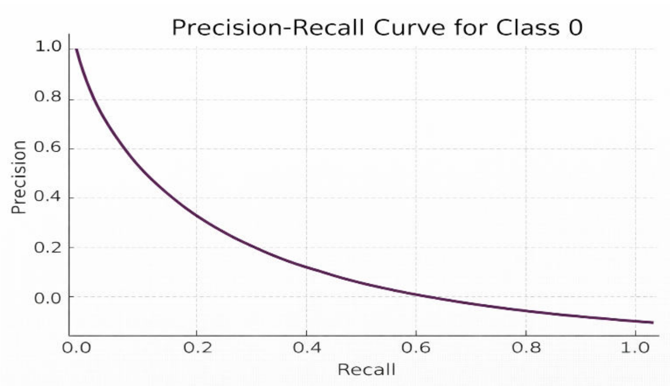


Fig .6 Precision–Recall Curve Analysis

Training vs. Accuracy of validation Analysis.

We can see in figure 7 the training and validation curves of accuracy when we were training the sign language translator on hearing-impaired users using machine learning methods.The model learns all this, such as the hand shape and movement. This is what makes it so much more efficient with the recognition. The model improves with use of sign language. An example of Epoch 3 is that the model is approximately percent.

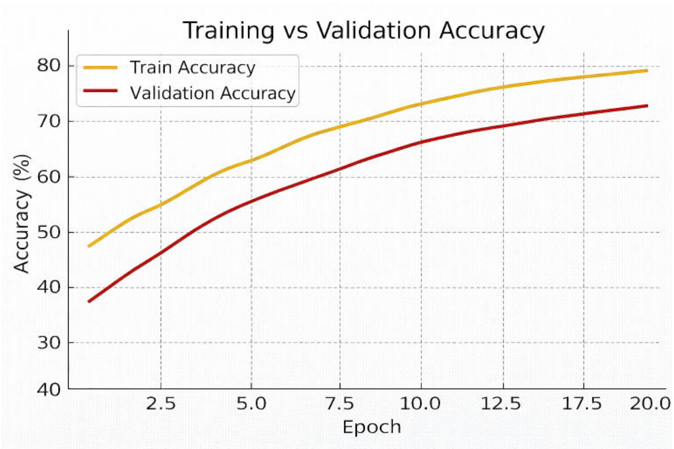


Fig.7 Training vs. Validation Accuracy Analysis

VI. CONCLUSION AND FUTURE SCOPE

In this paper, the author introduced a machine learning-driven sign language translation system that would enhance the accessibility of communication among hearing-impaired people. Using the methods of deep learning, the proposed structure is able to make the effective integration of Convolutional Neural Networks (CNNs) to extract spatial features and Long Short-Term Memory (LSTM) networks to extract temporal features as they allow proper recognition of sign gestures at the word level. This system was constructed to overcome critical issues of sign language recognition, such as time dependency, minute variations of hand movement, and diversity of signers, especially on the difficult WLASL100 dataset. Through experimental analysis, after incorporating optimized convolutional process, active temporal gating and spatial refinement in the form of pooling, it is clear that recognition is much more robust and performs better in generalization in the real world environment. Although current issues, including, but not limited to, the gesture boundary ambiguity and computational complexity, still exist, the findings suggest that CNN-LSTM designs could be an effective and scalable solution to the practical sign-language translation systems[8].

VII.REFERENCES

[1]. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for LargeScale Image Recognition", CoRR, vol. abs/1409.1556, 2015.

[2]. I.Kryvonos Krak and W. Wojcik, "Interactive systems for signing

learning", 2012 6th International Conference on Application of data and Communication Technologies (AICT), pp. 1-3, 2012.

[3]. P. Sonawane, K. Shah, P. Patel, S. Shah and J. Shah, "Speech to Indian Sign Language (ISL) Translation System", Proc. - IEEE 2021 Int. Conf. Comput. Commun. Intell. Syst. ICCCIS 2021, pp. 92-96, 2021.

[4]. Keerthi S Warrier, Jayant Kumar Sahu, Himadri Halder, Rajkumar Koradiya and Raj V Karthik, "Software Based Sign Language Converter", International Conference on Communication and Signal Processing, April 6-8, 2016.

[5].Azher Atallah Fahad, Hassan Jaleel Hassan and Salma Hameedi Abdullah, "Deep Learning-based Deaf & Mute Gesture Translation System", International Journal of Science and Research (IJSR), May 2020.

[6]. M. Punchimudiyanse and R. G. N. Meegama, "Computer Interpreter for Translating Written Sinhala to Sinhala Sign Language", OUSL Journal, vol. 12, no. 1, pp. 70, Jul. 2017.

[7]. W. C Stokoe, "Sign language structure: An outline of the visual communication systems of the American deaf", Journal of deaf studies and deaf education, vol. 10, no. 1, pp. 3-37, 2005.

[8]. Oscar Koller, Quantitative Survey of The State of The Art in Sign Language Recognition, September 2020.

[9].G.A. Rao and P.V.V. Kishore, "Selfie video based continuous Indian sign language recognition system", Ain Shams Engineering Journal, vol. 9, no. 4, pp. 1929-1939, 2018.

[10]. F. Zhang, V. Bazarevsky, A. Vakunov, et al., “Ondevice Realtime Hand Tracking,” arXiv:2006.10214 [cs], Jun. 2020, arXiv: 2006.10214.

[11] Arun Prasath G and Annapurani Panaiyappan k, " Design of an integrated learning approach to assist real-time deaf application using voice recognition system," SCI.

[12] G Arun Prasath and K Annapurani, "Prediction of sign . language recognition based on multi layered CNN," SCI2.

[13] G Arun Prasath and K Annapurani, "A Reversible Convolutional Neural Network Model for Sign Language Recognition," inass Publication.

[14] G. Arun Prasath and K. Annapurani, "A Review on Deaf and Dumb Communication System Based on Various Recognitions Aspect," Review on ICDCI2021.

[15] S. Shrenika and M. M. Bala, " Sign Language Recognition Using Template Matching Technique," In:Proc. of International Conf. on Computer Science Engineering and Applications., pp, 2020.

[16] M. R. I. J. S. Md. Abdur Rahim, "Non-Touch Sign Word Recognition Based on Dynamic Hand Gesture Using Hybrid Segmentation and CNN Feature Fusion," Applied Sciences, 2019.

[17] Angela C. Caliwag , Han-Jeong Hwang ,Sang-Ho Kim, Wansu Lim, "Movement-in-a-Video Detection Scheme for Sign Language Gesture Recognition Using Neural Network," Applied Sciences, 2022,12,10542.

[18] A. Mittal, P. Kumar, P. Pratim Roy, R. Balasubramanian, B.B. Chaudhuri, "A Modified LSTM Model for Continuous Sign Language Recognition Using Leap Motion," IEEE Sensors Journal, vol. 19, no. 16, pp. 7056-7063, 2019.

[19] Ilias Papastratis ,Kosmas Dimitropoulos, Petros Daras, "Continuous Sign Language Recognition through a Context-Aware Generative Adversarial Network," Sensors (Basel), vol. 7, no. 21, p. 2437, 2021.

[20] S. Praneel, et al., ”Malayalam Sign Language Character Recognition System,” 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT), IEEE, 2023.