Prediction of Sign Language Using Cross Pooling Highway Classifier Network

Dr. P. Kavitha, Dr. M. Santhalakshmi,

Assistant Professor, Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur 603203, Chengalpattu District, Tamil Nadu, India.

Corresponding Author

Assistant Professor, Department of Computer Science and IT, School of Computer Science and IT, Jain (Deemed to be University), Bengaluru.

Abstract—

Sign language recognition (SLR) has the potential to significantly diminish the communication barrier between the hearing and vocally impaired group and the rest of society. The first and essential step in comprehending and deciphering sign language is word-level sign language recognition (WSLR). Identifying cues from films, however, is a challenging endeavour since the interpretation of a phrase relies on a combination of nuanced body movements, hand positions, and other behaviours. Current pose-based WSLR approaches either fail to completely use the spatial information while representing the temporal information, or they simultaneously describe the spatial and temporal links among postures in different frames. We use a distinctive methodology rooted in artificial intelligence (AI) that collects data on posture and performs recognition in order to address the problem of WSLR. Initially, the data was obtained and preprocessed using a Fibonacci error filter (FEF). The area of interest is segmented using the Loop string moth optimisation technique. Ultimately, the cross-pooling highway classifier architecture thoroughly examines the connections between features and accurately interprets the significance of sign language, all in order to facilitate the decision-making process. The findings obtained from analysing a well-established dataset for word-level sign language recognition clearly indicate the superiority of the proposed system compared to conventional systems. The recommended system achieves very accurate predictions within a short timeframe, making it highly efficient for decision making.

Index Terms— Sign language recognition, Artificial intelligence, Fibonacci error filter, Loop string moth optimization, cross pooling highway classifier

I. INTRODUCTION

People who are deaf often use video records of sign language (SL) to teach others about the language or to raise knowledge of it. Most people agree that SL has the most well-organized structure of all sign languages. Multimedia, artificial intelligence (AI), and computer vision (CV) researchers are very interested in SL identification because it is so different from other methods. They use it to look for answers to problems like human motion analysis, HCI, and UI design. A lot of the things that pattern recognition and computer vision have in common can be seen in "behavioural recognition," which includes the study of sign language. "Sign language recognition" is the process of looking at people's movements, figuring out what they mean, and turning them into directions that make sense. The three steps that make up the basic methods for recognising sign language and gestures are detection, tracking, and identification. Most hand-to-hand conversations need to go through these steps to end. The motion recognition system's main goal is also to find the user's hands and then separate the visible area that matches to those hands. This division is very important because it only sends information that is needed for the job at hand to the next step, which focuses on tracking and recognition and doesn't need to know anything about the visual background. Because of this, standard methods take a long time and need several rounds of pre-processing to finish. This is why the people who wrote this study created a twin delayed deep reinforcement memory network that uses deep learning to find SL. We suggest a model with four separate parts:

preparation, feature extraction, segmentation, and classification. We discovered that thorough training can't fully utilise the high-complexity deep neural network because there aren't many samples to choose from. As a way to solve this problem, we have been looking into a cross-pooling highway predictor based on deep learning for SL prediction. The following is the outline for the rest of this project's sections. Past studies that have investigated SL recognition are discussed in Section 2. In Section 3, we discuss the deep learning network for SL detection and the optimization strategy we used to get our best results. The experimental findings of the suggested procedure are discussed in Se**ction** 4, and the work is brought to a close in Section 5.

II. RELATED WORKS

"The importance of SLR has prompted computer vision scientists to devote almost 30 years to investigating the topic. Initially, SLR approaches were categorised as either "isolated" or "continuous." Isolated SLR detected a single sign at a time, whereas continuous SLR recognised many phrases over the continually streaming input. The difficulty of identifying sign language gestures has led to many approaches. There have been several attempts to categorise Second Life hand gestures using various methods.

A technique for learning to understand Chinese hand signals is proposed by the author of [1]. Anxieties using electromyography (EMG) armbands and multimodal data from the ACC, GYR, and SMR channels, the researchers probe the problem of loneliness. They used a convolutional neural network (CNN-RNN) design for their network so that it could extract features from data as well as the temporal domain. We can make significant improvements to our approach by integrating an attention mechanism like SE-Block, which helps to give diverse sources of information the weight they deserve. This study aims to clarify the impact of spatial and temporal occlusion on the comprehension of sign language [2]. Manual cropping may include sufficient information for accurate prediction, as shown by video transformer model (VTN) results on the WLASL dataset being equivalent to an I3D baseline. The promise of convolutional neural networks (CNNs) to address issues in picture identification and classification has prompted much research into these networks. Over the past few years, there has been a lot of research and development focused on sign language recognition. A convolutional neural network (CNN) based method that removes the background and uses a Gaussian skin colour model was proposed as a potential solution for motion detection in camera images. We achieved a 93.80% success rate on a subset of the sample by eliminating background colours and using the Gaussian skin colour model to control how much light affects skin tone [3]. A two-stage CNN design called HGR-Net was proposed [4]. In the first step, semantic segmentation at the pixel level would be used to determine the region of interest. By integrating a fully convolutional residual network with spatial pyramid pooling, the proposed architecture improves the first stage's recognition accuracy by 1.6% on the OUHands dataset. A deep convolutional network (MultiD-CNN) was trained using a multidimensional feature learning strategy to detect motion in RGB-D movies [5]. The strategy surpassed previous approaches on a wide range of datasets because it used 3D ResNet to train a model with spatiotemporal properties and LSTM to grasp temporal correlations. Chen et al. introduced a fresh approach to hand gesture recognition using the dynamic graph method to spatiotemporal attention (DG-STA). A newly developed spatiotemporal mask is used to reduce the computational cost, while a fully connected network and a self-attention method are used to learn the hand skeleton's node properties and edges. When compared to other methods for hand gesture recognition, experimental findings demonstrate that DG-STA is the clear winner [6]. For the purpose of object detection in motion, a deep learning-based approach was proposed that makes use of two ResNet CNNs with w attention and a fully connected layer. A different approach was proposed, which included compressing digital movies into a single RGB image before feeding it into the algorithm for final classification. The proposed method achieves better accuracy than the state-of-the-art, according to experimental findings obtained from publicly available datasets [7]. We suggest Convolutional Neural Networks as a paradigm for gesture recognition. We may get useful spatial and temporal data by combining dynamic depth images (DDI), dynamic depth normal images (DDNI), and dynamic depth motion normal images (DDMNI) with bidirectional rank pooling. The proposed model demonstrated a 16.34% improvement in accuracy [8] when tested on the IsoGD dataset during the 2016 ChaLearn LAP competition assessment. The gesture detection issue was tackled using two separate deep learning algorithms. By explicitly defining the short-term and long-term structures using a convolutional two-stream consensus voting network (2SCVN), the motion properties of RGB sequences were uncovered. When applied to the ChaLearn IsoGD dataset, the proposed techniques improved accuracy 4.47 percentage points over the models from 2016 [9]. The authors of [10] developed a system that can detect and identify gestures in real-time using a model based on recurrent 3D convolutional neural networks (CNNs). To enhance the accuracy of identification, data from RGB, depth, optical flow, and stereo IR were integrated. The proposed model outperforms the mean of all existing models on the ChaLearn dataset with a 1% improvement in positive accuracy. It was proposed that two-stream convolutional neural networks (CNNs) be used to recognise and identify hand motions using the depth map and optical flow as inputs. An accuracy gain of 18.91% is achieved by the proposed model when compared to earlier models that used the same MSR Action3D dataset [11]. In order to decipher sign language, Rastgoo et al. created the model. As for the picture analysis, they relied on a restricted Boltzmann machine (RBM). We used information from three colour and depth-enhanced images—the original, a cropped version, and a noisy version—to construct the model. First, the hands in each picture are recognised by a convolutional neural network (CNN). Then, the RBM receives one of three renditions of the recognised hands via RGB and depth channels. To find out what the sign is called, we'll mix the RBM's output with other signals. Results on four publicly accessible datasets demonstrate that the suggested model outperforms the state-of-the-art methods [12]. A deep cascaded approach for movie sign language recognition was introduced in 2020 by the author of [13] using the RBM paradigm. Combining hand traits with ESHR features and HP information allowed us to extract temporal components from the data using the long short-term memory (LSTM) model. This is why the problem of hand detection was also tackled using the SSD model. They found that, when tested on the IsoGD dataset, their proposed model outperformed rival algorithms by 4.25%. In [14], Bag of Visual Words (BOVW) is introduced as a paradigm for identifying the A-Z and 0-9 numbers and letters of the Indian sign language alphabet in a real-time video feed, with the expected labels either spoken aloud or shown on the screen. In our segmentation approach, we use both skin tone and background removal techniques. After features were extracted from the images using SURF (Speeded Up Robust Features) and histograms were created, labels could be changed into symbols. Convolutional Neural Networks (CNNs) and the Support Vector Machine (SVM) are used to achieve classification. The development of a user-friendly GUI for the client is also undertaken. Using convolutional neural networks, the ASL alphabets may be detected, as stated in [15]. deep neural networks. The author of [16] classified the 26 ASL letters that correspond to the English alphabet using a mix of k-nearest neighbour and a support vector network that had been trained on sensory input-generated features. In [17], the author offered his services in the hopes of securing the company's future success. We provide a recommendation system for LCSs (RS-LCS) based on a hierarchical data architecture that considers user queries in relation to current developments and wants. Operating under tight deadlines is part of the company's strategy. This paper presents a novel LCS architecture that relies on group ratings and a set of services [18]. To make the LCS network more adaptable to fresh data, they want to include a group recommendation mechanism. The year 19 To improve the accuracy of your suggestions, train a Neural Network to use user ratings and reviews. As shown in [20], a customer's repayment capacity may be ascertained with remarkable accuracy by maximising a collection of metrics. Python and machine learning techniques allowed us to accomplish our main objective. Using Min-Max standardisation, Logistic Regression, a Random Forest classifier, and an image classification system, a deep learning model developed using tensor flow can detect whether a loan application is creditworthy or not.

Numerous deep-based models, ranging in complexity, have been created by researchers during the last few years. By combining different kinds of data with different kinds of inputs, a broad variety of models might be given. When dealing with video input, it is best to employ dynamic and continuous dynamic signs rather than the static image paradigm often used by most signs. Due to the input video's frequent transitions between sign kinds, continual dynamic signs impede movement epenthesis. Researchers have so far concentrated on distinguishing between static and dynamic sign language, rather than continuously dynamic sign language. If we want to localise signals, gestures, and postures properly, we may require strategies that replicate the spatio-temporal modelling of structures and patterns in present models. Adding discriminative data makes the model more difficult, but it performs better. Models used for sign language recognition and related fields can be made simpler by utilising deep learning's parallel computation capabilities, implementing more accurate fusion methods to integrate multiple input modalities (particularly in continuous dynamic sign language), and combining traditional methods with deep-based models".

III. PROPOSED WORK

"Communicating with hearing persons who do not use sign language is a visually attractive and practically substantial challenge for deaf people. Examining the feasibility of using deep learning to the problem of sign identification is the primary objective of this research. Figure 1 shows the whole process of sign language recognition.

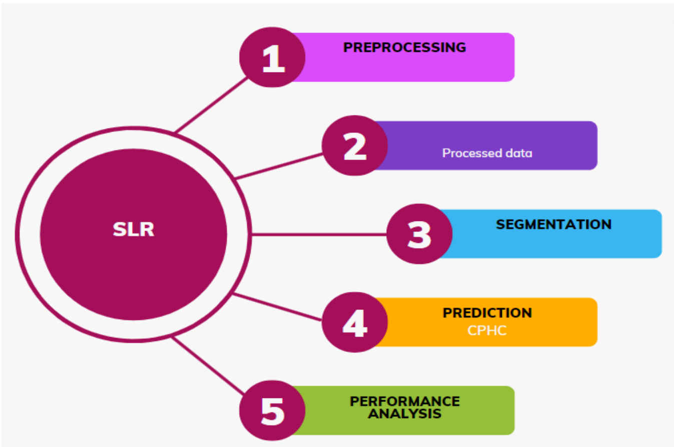

Figure 1 Schematic representation of the suggested methodology

3.1 Data source

For Word-Level American Sign Language (ASL) identification, there is no larger video collection than WLASL, which has over 2,000 unique words. Our long-term goal is to improve communication between the hearing and deaf communities via more research into sign language understanding, which we hope WLASL will promote.

3.2 Preprocessing

Noisy and deteriorated images are filtered before restoration utilising several FEF techniques. The corresponding mathematical expression would be:

$h(O,z) = f(O,z) * u(0,z) + n(O,z)$
(1)

$h(O,z) = R[h(O,z)]$
(2)

f(O,z) represents the original input image, u(O,z) is the degradation function, "*" denotes the error function, n(O,z) stands for the noise (usually Gaussian noise), g(O,z) is the deteriorated output image, and h(O,z) is the final degraded output image after applying the technique R. By using noise reduction filters with nonlinear spatial domains, like the one seen above, it is feasible to reconstruct denoised images from noisy source images. The proposed filter use the average and standard deviation of the pixel values in the mask matrix.

$$\mu = \frac{1}{OM}\sum_{n,m\in\eta} a(0,m)$$

(3)

$$\sigma^2 = \frac{1}{OM}\sum_{o,m\in o} a^2(0,m) - \mu^2$$

(4)

The mask's neighbourhood area has a size of nm., σ^2 is the variance of the Gaussian noise in the picture is denoted by σ^2, and a(n,m) represents each pixel in the mask.

$$c_w(0,m) = \mu + \frac{\sigma^2 - v^2}{\sigma^2}\cdot(a(0,m) - \mu)$$

(5)

where v^2 is the mask matrix's noise variance setting when using the filter.

The imputed pixel values are now provided as follows:

$$c_i^{imp} = \sum_{j=1}^{k} w_j \boldsymbol{z}_j,\ i = 1,\dots,m.$$

(6)

After error removal and imputing the pixel values the error free images are obtained

3.3 Segmentation

The Loop string moth optimisation segmentation algorithm may be provided with the produced image. We will denote the regions in Si containing the initial moths as $B_1, B_2, \dots, B_i$. $(\bar{O}, \bar{D}_{\mathrm{b}}, \bar{D}_{\mathrm{r}})$ will illustrate how the total amount of pixels in all Si moths is divided into $(\bar{O}, \bar{D}_{\mathrm{b}}, \bar{D}_{\mathrm{r}})$Here, we present our proposed segmentation approach.

(1) Choose your moths automatically.

(2) Give each moth area a label.

The moth pixel must have several similarities with the neighbouring pixels. To produce the anticipated region, at least one seed must be generated. Thirdly, it is crucial to store seeds for different sites separately.

The algorithm provided is used to determine the level of similarity between a certain pixel and its surrounding neighbours. The dispersion measurements of the luminance, blue chrominance, and red chrominance Utilising the components of a 3x3 grid

$$\sigma_Y = \sqrt{\frac{1}{9}\sum_{i=1}^{9}(O_i - \bar{O})^2},$$

(7)

where O can be Y, D_{b}, or D_{r}, then the mean value $\bar{Y} = \frac{1}{9}\sum_{i=1}^{9} x_i$. Standard deviation, on the whole, is

The second step is to determine the $YD_{\mathrm{b}}D_{\mathrm{r}}$ distances (relative Euclidean distances) between a pixel and its immediate neighbours.

$$d_i = \frac{\sqrt{(O-O_i)^2 + \left(D_{\mathrm{b}} - D_{\mathrm{b}_i}\right)^2 + \left(D_{\mathrm{r}} - D_{\mathrm{r}_i}\right)^2}}{\sqrt{O^2 + D_{\mathrm{b}}^2 + D_{\mathrm{r}}^2}}$$

(8)

We use the labels to discuss the differences in colour between these places.

$$d(K_i, K_j) = \frac{\sqrt{\left(\bar{O}_i - \bar{O}_j\right)^2 + (\bar{C}_D - \bar{C}_{\mathrm{D}})^2 + \left(\bar{D}_{\mathrm{r}_i} - \bar{D}_{\mathrm{r}_j}\right)^2}}{\min\left(\sqrt{\bar{Y}_i^2 + \bar{D}_{\mathrm{b}_i}^2 + \bar{D}_{\mathrm{r}_i}^2}, \sqrt{\bar{O}_j^2 + \bar{D}_{\mathrm{b}_j}^2 + \bar{D}_{\mathrm{r}_j}^2}\right)}.$$

(9)

After repeating the process the ROI can be separated.

3.4 Prediction

The sign may be readily manipulated during analysis. The frequency of each pixel value may be calculated by combining all the pixels in a single picture file. The recommended approach was used to discover the most common pixel pairings. The labels' probabilities are calculated using a softmax layer. Backpropagation is used in the training phase to automatically modify the majority of parameters inside the neural network. Each command group is represented in the first layer of the representation by a single hot vector, xn. We next create one-hot vectors for each of the next n occurrences.

$\{y1, y2, \dots, yn\}$.

The entire technique is shown by the use of Formula 10.

$$\boldsymbol{Y} = \boldsymbol{y} * \boldsymbol{v_E}$$

(10)

Convolutional layers are used to extract several characteristics from the input data. A network with several convolutional layers may progressively extract complex features from smaller input data. The suggested architectural architecture enables the incorporation of several layers of convolutional processing. The convolutional layers are numbered consecutively from 1 to L. The matrix Y, with dimensions n*E, is fed into the first convolutional layer. The number of convolution filters in the layer is denoted by the notation m1 for the lth convolutional layer. The filter size in the first convolutional layer is represented as s1*E, with s1 being the maximum length of executable instructions that can be precisely recognised. Deeper convolutional layers accept input from the previous layer and use their own output as input. The filter size in the next convolutional layer is calculated by multiplying the spatial dimensions of the input feature map in the previous layer (sl) by the number of channels in that feature map (m(l-1)). After applying convolution filters on variable Y, the activation function is used to retain important features and eliminate irrelevant information. The ReLU (Rectified Linear Unit) activation function is used in this specific circumstance.

$$\mathrm{Re}\ LU = \max(0, x)$$
(11)

Each convolutional filter in convolution layer l (of size m) generates an activation map $\boldsymbol{a}_{l,m}$ of size n*1. $\boldsymbol{W}_{l,m}$ and $\boldsymbol{b}_{l,m}$ is the mth convolutional filter in layer l has weight and bias parameters. In the first Convolutional layer, Conv represents the convolutional process of the filter on the input matrix Y.

$$B_{l,m} = \mathrm{Re}\ \boldsymbol{LU}\big(\boldsymbol{Conv}(\boldsymbol{Y})_{W_{l,m},b_{l,m}}\big)$$
(12)

Max-pooling generates a vector by selecting the highest value from each activation map in the final layer. After the inclusion of the max-pooling layer, we successfully generated a β of magnitude $\boldsymbol{m}_{\boldsymbol{L}}$..

$$\beta = \Big(\max(B_{L,1})\big|\max(B_{L,2})\big| \ldots \mid \max(B_{L,m})\Big)$$
(13)

Using the max-pooling approach, a vector $\boldsymbol{\beta}$ of length $\boldsymbol{m}_{\boldsymbol{L}}$ the data was produced/generated.

IV. Performance Analysis

The suggested sign language identification model is evaluated via a number of tests. The primary goal of the experimentation series is to assess the proposed model by implementing it on a dataset. Most of the testing was conducted in a Python environment.

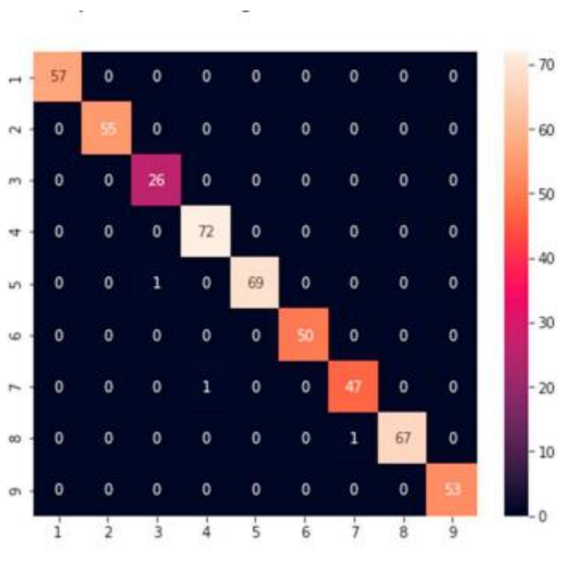

Figure 2 confusion matrix

The confusion_matrix library is used to analyse the real-time performance of the model. The confusion matrix quantifies the number of correctly predicted labels. It also allows for visualising the discrepancy in forecasts when the model makes incorrect predictions. Figure 2 displays the confusion matrix of the model.

Figure 3 Simulated output

The overall simulated output was illustrated in figure 3

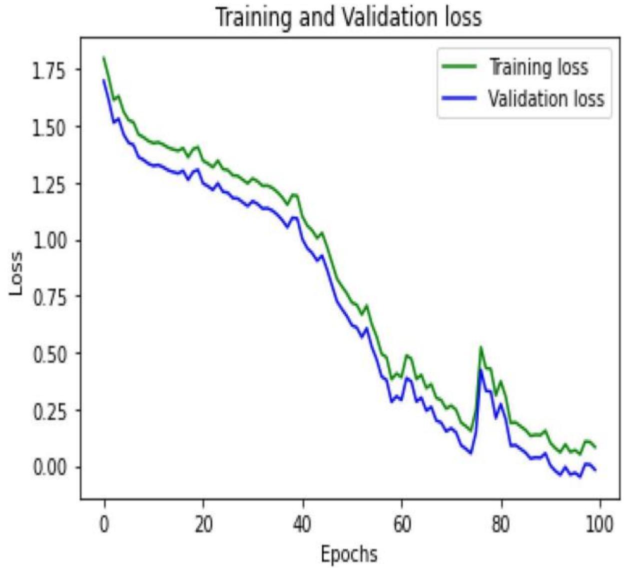

Figure 4 Epoch Vs. Loss

Figure 4 shows that the epoch experienced a loss of 0.1 and an MSE of 0.02.

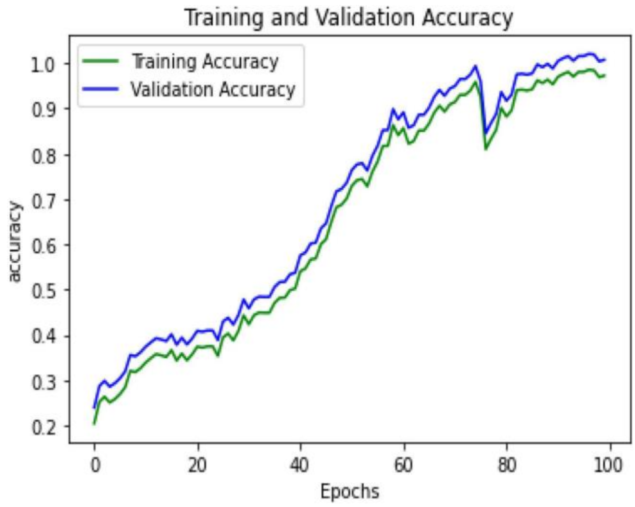

Figure 5 Epoch Vs. Accuracy

The proposed approach improves the average accuracy of EEG-based emotion recognition to 98% by the combination of feature extraction and a classification model. Figure 4 displays the accuracy rates for testing and learning. An accuracy of 0.996 and a Mean Squared Error (MSE) of 0.008 were attained in the training model after 100 epochs.

Class label	Precision	Recall	F1 score	Support
1	1.00	1.00	1.00	57
2	1.00	1.00	1.00	55
3	0.96	1.00	0.98	26
4	0.99	1.00	0.99	72
5	1.00	0.99	0.99	70
6	1.00	1.00	1.00	50
7	0.98	0.98	0.98	48
8	1.00	0.99	0.99	68
9	1.00	1.00	1.00	53
accuracy			0.99.6	499
macro avg	0.99	0.99	0.99	499

Figure 6 Classification report

The classification report for the suggested methodology was illustrated in figure 6

Table 1 Comparative performance analysis

"Sign Language	Authors	Approach	Accuracy	Year of Development
American Sign Language	Das et al. [21]	CNN based Inception V3 model	90%	2018
Indian Sign language	Rekha et al. [22]	Skin color segmentation with SVM	86.3%	2011
Indian Sign language	Sahoo [23]	k-NN and Naïve Bayes classifier	98%	2021
American, Indian, Italian and Turkey Sign Language	Halder and Tayade [24]	MediaPipe with SVM	99%	2021
(this work)	This approach	CPHC	99.6%	2024"

As of from the analysis the suggested methodology outperforms other methodology by obtaining high range of accuracy.

V. CONCLUSION

This effort aims to provide a visual solution for the issue of sign language recognition. The proposed methodology's classification results indicate that it is more efficient (99.6%) and quicker in recognizing complex hand signs and gestures compared to other models in the literature that demand significant computational power and longer training time. Additionally, because of its lightweight nature, the model becomes more resilient and can be deployed across a range of computing devices with varying processing capabilities without compromising on speed and accuracy. This study may be expanded to include the identification of more signals from the Assamese Sign Language, including dynamic motions used in everyday conversation. Furthermore, other deep learning methods may be evaluated after the integration of MediaPipe's hand tracking technology to enhance the precision and effectiveness of the model".

References

1. George, F.P.; Shaikat, I.M.; Hossain, P.S.F.; Parvez, M.Z.; Uddin, J. Recognition of emotional states using EEG signals based on time-frequency analysis and SVM classifier. Int. J. Electr. Comput. Eng. 2019, 9, 1012–1020.
2. Alhagry, S.; Aly, A.; Reda, A. Emotion Recognition based on EEG using LSTM Recurrent Neural Network. Int. J. Adv. Comput. Sci. Appl. 2017, 8, 8–11.
3. Yin, Y.; Zheng, X.; Hu, B.; Zhang, Y.; Cui, X. EEG emotion recognition using fusion model of graph convolutional neural networks and LSTM. Appl. Soft Comput. 2020, 100, 106954.
4. Cimtay, Y.; Ekmekcioglu, E. Investigating the Use of Pretrained Convolutional Neural Network on Cross-Subject and Cross-Dataset EEG Emotion Recognition. Sensors 2020, 20, 2034.
5. Wei, C.; Chen, L.-L.; Song, Z.-Z.; Lou, X.-G.; Li, D.-D. EEG-based emotion recognition using simple recurrent units network and ensemble learning. Biomed. Signal Process. Control 2020, 58, 101756.
6. Chao, H.; Liu, Y. Emotion Recognition From Multi-Channel EEG Signals by Exploiting the Deep Belief-Conditional Random Field Framework. IEEE Access 2020, 8, 33002–33012.
7. Pandey, P.; Seeja, K. Subject independent emotion recognition from EEG using VMD and deep learning. J. King Saud Univ. Comput. Inf. Sci. 2019, 11, 1–9.
8. Misulis, K. E., Abou-Khalil, B., Sonmezturk, H., and Ess, K. C. (2022). Atlas of EEG, seizure semiology, and management. Oxford: Oxford University Press.
9. Michielli, N., Acharya, U. R., and Molinari, F. (2019). Cascaded LSTM recurrent neural network for automated sleep stage classification using single-channel EEG signals. Comput. Biol. Med. 106, 71–81. doi: 10.1016/j.compbiomed.2019.01.013
10. Qazi, E.-U.-H., Hussain, M., Aboalsamh, H., and Ullah, I. (2019). Automatic Emotion Recognition (AER) system based on two-level ensemble of lightweight deep CNN models. arXiv [Preprint]. doi: 10.48550/arXiv.1904.13234
11. Korotkova, T., Ponomarenko, A., Monaghan, C. K., Poulter, S. L., Cacucci, F., Wills, T., et al. (2018). Reconciling the different faces of hippocampal theta: The role of theta oscillations in cognitive, emotional and innate behaviors. Neurosci. Biobehav. Rev. 85, 65–80. doi: 10.1016/j.neubiorev.2017.09.004
12. Newson, J. J., and Thiagarajan, T. C. (2019). EEG Frequency bands in psychiatric disorders: A review of resting state studies. Front. Hum. Neurosci. 12:521. doi: 10.3389/fnhum.2018.00521
13. Ramzan, M., and Dawn, S. (2019). Learning-based classification of valence emotion from electroencephalography. Int. J. Neurosci. 129, 1085–1093.
14. Fitzgerald, J. M., Gorka, S. M., Kujawa, A., DiGangi, J. A., Proescher, E., Greenstein, J. E., et al. (2018). Neural indices of emotional reactivity and regulation predict course of PTSD symptoms in combat-exposed veterans. Prog. Neuropsychopharmacol. Biol. Psychiatry 82, 255–262.
15. Suhaimi, N. S., Mountstephens, J., and Teo, J. (2020). EEG-based emotion recognition: A state-of-the-art review of current trends and opportunities. Comput. Intell. Neurosci. 2020:8875426.
16. Topic, A., and Russo, M. (2021). Emotion recognition based on EEG feature maps through deep learning network. Eng. Sci. Technol. Int. J. 24, 1442–1454. doi: 10.1016/j.jestch.2021.03.012
17. Wang, J., and Wang, M. (2021). Review of the emotional feature extraction and classification using EEG signals. Cogn. Robot. 1, 29–40. doi: 10.1016/j.cogr.2021.04.001
18. Pereira, E. T., Gomes, H. M., Veloso, L. R., and Mota, M. R. A. (2018). Empirical evidence relating EEG signal duration to emotion classification performance. IEEE Trans. Affect. Comput 12, 154–164. doi: 10.1109/TAFFC.2018.2854168
19. Liu, J., Wu, G., Luo, Y., Qiu, S., Yang, S., Li, W., et al. (2020). EEG-Based emotion classification using a deep neural network and sparse autoencoder. Front. Syst. Neurosci. 14:43. doi: 10.3389/fnsys.2020.00043
20. Iyer, A., Das, S. S., Teotia, R., Maheshwari, S., & Sharma, R. R. (2023). CNN and LSTM based ensemble learning for human emotion recognition using EEG recordings. Multimedia Tools and Applications, 82(4), 4883-4896.

21. A. Das, S. Gawde, K. Suratwala and D. Kalbande. (2018) "Sign language recognition using deep learning on custom processed static gesture images," in International Conference on Smart City and Emerging Technology (ICSCET).
22. C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee and others. (2019) "Mediapipe: A framework for building perception pipelines," arXiv preprint arXiv:1906.08172.
23. A. K. Sahoo. (2021) "Indian sign language recognition using machine learning techniques," in Macromolecular Symposia.
24. A. Halder and A. Tayad. (2021) "Real-time vernacular sign language recognition using mediapipe and machine learning," Journal homepage: www. ijrpr. com ISSN, vol. 2582, p. 7421.