JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Dynamic Gestural Sign Recognition Using Deep Neural Network

Purva C. Badhe, Vaishali Kulkarni
Assistant Professor, Professor
Department of Artificial Intelligence and Machine Learning,
DJSCOE, Mumbai, India

Abstract: The ability to communicate through gestural sign language is a fundamental human right for the hearing impaired community. It is the sole medium using which these people communicate with others. It is essential than to be able to converse effectively, the other person is aware of the gestures, their meanings and be able to understand them. However, in India, there has been negligence towards the development and awareness of this sign based language. The national standard sign language of India, known as Indian Sign Language (ISL) was recognized by mass only after 1970s and the standardized dictionary of ISL gestures is still in process of creation with more than 15000 words already released by Indian Sign Language Research and Training Center (ISLRTC) of Govt. of India. To improve the communication between ISL users and others who are not aware of ISL, recent technological developments can be utilized. There have been recent studies in the domain of developing such automatic systems that will translate ISL gestures into spoke languages. Some methods are sensor-based while others use computer vision. Various technologies employed include SVM, HMM, SIFT, DCT, KNN, ANN and so on. With emerging knowledge of artificial intelligence use of CNN and RNN is currently being employed in the same field. In this paper we are presenting one such methodology that makes use of hybrid CNN-RNN based approach to recognize dynamic ISL gestures into English language. A combination of CNN-RNN provides benefits of both spatial as well as hierarchical feature data. The proposed system is trained using self-created database and tested for 10 different classes with testing accuracy of 90% which is at par with other currently available methods.

Index Terms **- Sign Language Recognition, CNN-RNN, Dynamic Gesture Recognition, Indian Sign Language, Transfer Learning.**

Introduction

Indian Sign Language Recognition is an emerging area of research in this era, specifically with fast developing technology and tremendous progress in Human Computer Interface (HCI) domain. Several researchers are working towards developing state of art facility of the deaf and mute community for providing ease of communication. Sign Language is the most essential tool that deaf community uses for social communication This communication medium between the hearing and deaf people comprises of physical movements as well as physical gestures [1]. Sign Language (SL) encompasses both manual gestures and non-manual gestures. Manual gestures, as depicted in Figure 1, solely utilize the hands to convey a gesture. In contrast, non-manual signs incorporate additional elements such as postures, expressions, head movement, eye gaze, and more.

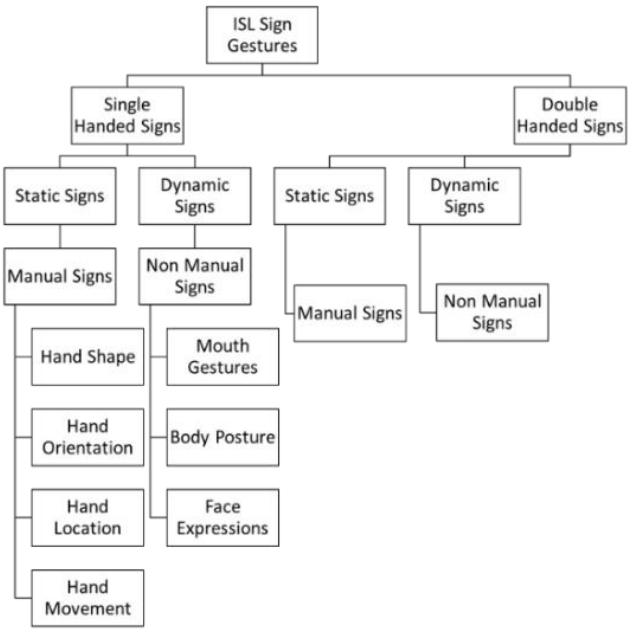

Fig. 1 Classification of ISL Sign Gestures [2]

With recent advancements in computing, graphics and artificial intelligence, recognition of Sign Language has been a popular field of research. Gesture to Text, Audio to Gestures, Gestures to Voice are some of the emerging domains.

Two methods can be employed for automatic sign language translation: (A) Sensor-based approach using signal processing techniques, and (B) Computer vision approach utilizing image and video processing. In the first method, a sensor hand glove is worn by the signer to capture the signals originating from their fingers. The later approach uses 2D camera sensor, leap motion sensor or depth image sensor captures image data. The lateral approach proves superior when compared to the first approach, as it encompasses a broader range of details, including the signer's hand, hand movements, hand shape, head position and movement, facial expressions, and torso motion [3]. Automatic recognition involves visual complexity, signer's behaviors, background processing, also lack of awareness and use of standardized Indian Sign Language (ISL) library all this contributing makes this is a complex engineering problem. However, Artificial intelligence has replaced older methods like HMM, CRF, DTW etc. [1] creating more opportunities. SLR uses semantic information from sequence of frames, motion maps, depth images or video sequences thus requiring temporal analysis. Multimodal or multidimensional neural networks prove to be worthwhile in handling spatial-temporal features. In this paper, we present our proposed methodology, which combines Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) units. The aim is to recognize continuous gestures in Indian Sign Language (ISL) and translate them into English text output. Our hybrid model of CNN and RNN is designed to effectively recognize 10 different dynamic gestures commonly used in the daily life of ISL signers. The model's evaluation and validation are conducted using various metrics. The classification of gesture features is performed using the LSTM unit of the RNN. The subsequent sections of the paper are organized as follows: Section 2 provides an overview of relevant studies conducted on ISL as well as sign languages used in other countries. Section 3 elaborates on our proposed methodology, presented through a comprehensive block diagram. Section 4 covers the implementation of our work and includes a comparison with existing methods. Finally, Section 5 concludes the study, summarizing the findings and implications of our research.

I. Related Work

Sign Language recognition systems encounter significant challenges, primarily stemming from the lack of universality across sign languages. Different countries possess their own distinct Sign Languages, with unique grammatical rules [4]. Prominent examples include American Sign Language (ASL), British Sign Language (BSL), Spanish Sign Language, Israeli Sign Language, Indian Sign Language (ISL), Chinese Sign language, Bangla Sign Language, and more. Developing a Sign Language recognition system usually entails a five-stage process, as depicted in Figure 2. This framework guides the systematic progression of the system's development and implementation.

Fig. 2 Block Diagram of Sign Language Recognition System

The initial stage of the system involves capturing Sign Language gestures. The type of capturing device depends on the type of input. Some of the systems are based on Sensor based input data [5] where in the input acquisition device is in term of a hand-glove or wearable sensor template. The other type of input is in term of visual data. These vision-based techniques may use RGB video sequences [2], or Depth Images [6], or Motion sensor-based images [7], Thermal images, or Kinect sensor images [8]. Once these gestures are acquired, they are pre-processed for ease of feature extraction. Enhancing the features obtained in this step is crucial as they are subsequently chosen for further classification. The third stage holds paramount significance within the overall process. In feature extraction stage vital features of the gesture are identified by employing various technologies. Many researchers have reported in their literature successful implementation of various feature extraction methods such as Kinect features, HOG features [9], K-Nearest neighbors and SVM method [10], Hand shape, velocity vectors, motion vectors [11], DCT, DFT [12], EMG and arm sensor data [13], PCA [14] and many different ways. Extracted features were then fed to the classifier for recognizing the gesture. Out of various types of classifiers, some famous ones are Minimum distance classifier [4], ANN [15], CNN [16] etc. the last stage of the system is recognition. The classified gestures can be depicted either in text or in the voice output in spoken language of interest. This section encompasses an array of research focused on Sign Language recognition technology, encompassing the Indian Sign Language (ISL) as well as sign languages from other countries.

In their work, Kumar et al. [17] have introduced a 3D Sign Language recognition system that utilizes spatio-temporal graph kernels. Within their system, the authors have devised a twin motion algorithm capable of effectively recognizing 3D signs that involve joints with varying degrees of motion. The approach utilized in this method is signer invariant, relying on the range of relative distances between joints. To construct the database, a 9-camera 3D motion capture setup is employed, encompassing 2500 signs that span across 500 distinct gestures. Within the proposed system, motion and non-motion joint frames are clustered using wide motion descriptors, followed by classification using an SVM classifier. The reported average accuracy of the system stands at 98.4%. However, a limitation of this method lies in its failure to consider the separation of motion joints into wide and narrow motions. In a separate study by Rao G.A. et al. [18], a deep convolutional neural network is proposed for Sign Language recognition. The authors have stated the use of mobile camera's selfie capturing feature for this study. Authors claim the real-world usefulness due to a hand-held capturing device. The dataset contains 60000 static signs. The feature extraction as well as classification is done by deep learning technology. Convolutional layers are extracting features by supervised learning and softmax is deployed for classification. The performance of proposed system is reported to give 91.12% accuracy when same dataset is used for testing where as the 82.03% in case of different dataset. The major limitation of this approach is restriction to the static signs only. In another method [1], Modified LSTM model has been described that recognizes continuous sign languages captured using leap motion sensor. Database includes 35 various signs in 942 sentences. The features extracted are 12 per frame using CNN and for recognition LSTM network is used. The accuracy of independent words is reported to be 89.5% and for sentences is 72.3%. As far as international sign languages are concerned, (Tao W. et al, 2018) have proposed use of CNN for American Sign Language (ASL) alphabet recognition. The

distinguishing feature of this study is data augmentation in order to increase the database. Virtual cameras are used from multiple directions to generated augmented images. The drawback of system remains same as stated for other CNN approach that only static signs are considered for study limiting the scope of the usefulness. The researchers have used public ASL database including 500 to 600 samples. The reported accuracy for the proposed system is 84.8%. For Persian Sign Language Recognition system Azar S. and Sayedarabi H, 2020 [19] have proposed Trajectory based recognition using HMM method. Using 1200 videos of 20 dynamic gesture classes system claims to give 97.48% accuracy. Rahaman et Al, 2020 [20] have developed recognition model for hand spelled Bangla Sign Language gestures. This work also uses static gesture images rather than considering the continuous video and provides 95.8% average accuracy. Some of the findings from detailed literature review are noted below:

- In case of ISL, due to unavailability of standardized dataset, the research doesn't show any particular way of handling the dataset. Some authors have created their own dataset by capturing the data using multiple cameras integrating depth and RGB features, where as some have used Kinect sensor and some use normal digital camera or mobile camera.
- There is a limited details available about feature extraction procedure. Not much emphasis has been given on further usefulness of the systems. In case of ISL, being manual signs, there is equal importance to facial expression. This feature is missing in most of the studies.
- Through our communication with a sign language instructor at Ali Yavar Jung National Institute of Hearing Handicapped, we gained valuable insight into the usage of sign space in Indian Sign Language. Unlike the majority of research, which primarily concentrates on hand and finger movement, Indian Sign Language encompasses the entire upper half of the body, including the head, face, neck, upper body posture, and hands.
- There is still scope of betterment possible in case of applying deep learning technology and its advancements to the domain of study and to consider dynamic video gestures instead of static image signs.

After carrying out extensive literature review, we propose a methodology for recognition dynamic ISL gestures using hybrid CNN-RNN technology. In section 3 the detailed proposed methodology has been explained.

II. Proposed Methodology

The objective behind this study is to come up with a method to identify gestures made in ISL and recognize them into spoken language English. As stated in section 2 with reasons, we have proposed a combined CNN_RNN technology for the same. CNN is a powerful tool to handle hierarchical data and to extract distinctive features automatically, while RNN has proved its success in generating sequential labels, thus the combination of two can be deployed to take benefit from both. Many advantages of CNN-RNN combination have been listed down by [21] that include:

- Learning features in a hierarchical manner aligns with human perception and the organization of concepts, thereby promoting consistency.
- Combination of CNN-RNN classification performance improves.
- The combination of CNN-RNN is versatile and transferable, meaning that the framework can be applied on any CNN architecture designed for single-level classification. By doing so, it enhances the performance for each hierarchical level within the system.
- The utilization of end-to-end training facilitates the joint learning of features and their relationships to classes within the RNN framework.
- Flexibility in RNN gives us freedom to have variable hierarchical label length.

3.1 Block Diagram

The proposed methodology's block diagram, depicted in Figure 3, illustrates the following steps: Input gestures are captured and stored locally, subsequently divided into training and testing folders.

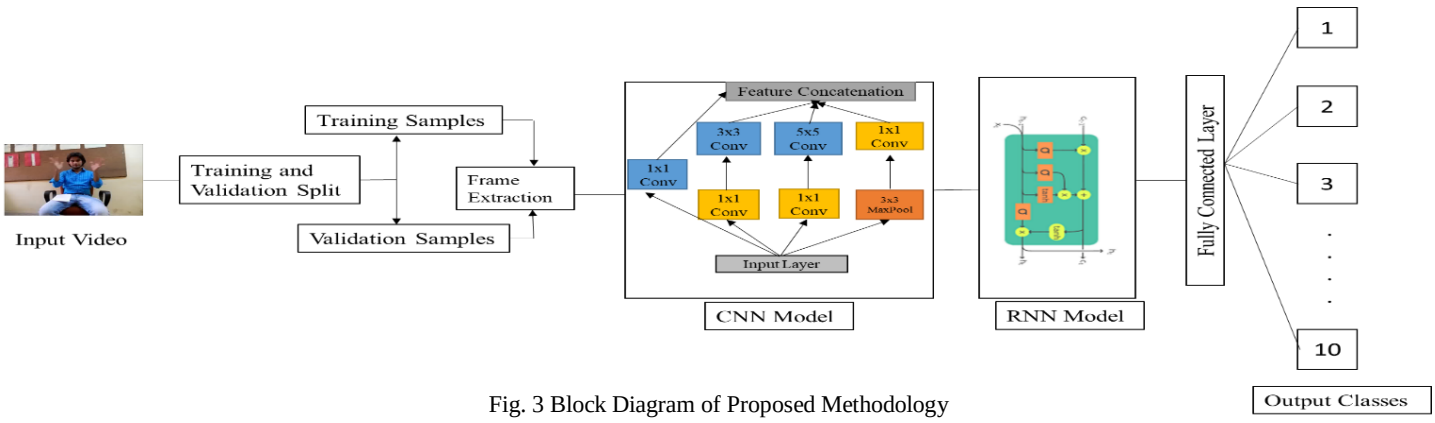

Fig. 3 Block Diagram of Proposed Methodology

Convolutional neural networks (CNNs) are employed for automatic feature extraction from the captured videos. Framewise feature extraction is performed on each frame of the stored videos. The extracted features are then sequentially passed through a Long Short-Term Memory (LSTM) model to classify the sequence. Unlike other Neural Networks, the CNNs do not have weights for each layer, instead the weights are shared and then convoluted across the input as a moving window. The important parameters while defining a CNN are: Convolution Layer, which uses a filter that acts as a weight matrix. The output from the previous layer is slide across to calculate the dot product of inputs and weights. Backpropagation is performed to update the weights in order to minimize the error. The next parameter, an activation function, it a non-liner function (ReLU), it is applied to the convolution elements. The proposed

methodology incorporates the usage of a feature map, where each component represents the output of a single neuron. To reduce computational complexity, a pooling layer is employed to decrease the size of the feature map. The output layer, with 10 nodes in this particular case, corresponds to the number of classes. The CNN implementation in the methodology utilizes the well-known Inception V3 model [22], depicted in Figure 3, known for its multi-level feature extraction capabilities. It performs convolutions of sizes 1x1, 3x3, and 5x5 within the same modules, stacking the outputs along the channel dimension. After extracting features using the CNN, they are subsequently passed through a sequential model, namely the Long-Short Term Memory (LSTM) network, for classification. RNN models, particularly LSTM, are well-suited for temporal data. Hochreiter et al. [23] introduced LSTM as an improved RNN architecture that utilizes a gradient-based learning algorithm. LSTM overcomes several limitations encountered by conventional RNNs and finds widespread application in computer vision and machine learning tasks. RNNs, in general, make use of previous learning for classification tasks and contain loops that enable information to be fed back. However, issues can arise with long-term dependencies in short-term memory. In the proposed model, four layers of LSTM, including one dropout layer, are employed, followed by an output layer with the SoftMax function for classification.

3.2 DATASET

The dataset for conducting the experiment is captured from natural deaf signers. 10 signers have made dynamic gestures in 5 iterations. Table 1 below has a list of signs included for the study. In table 2 details of dataset specifications have been mentioned.

Table 1. List of gestures included in database

Hello	India	Thank You	Sorry	Please
Danger	Fire	Namaste	Help Me	I am Hungry

Fig. 4 has few examples of frames extracted from the dynamic gesture video sequence. It is evident from the figure that our database is not restricted to any background or clothing set up. Gestural space includes the complete frame of signer including his head, hands and trunk. The average length of the video is 2 sec. Video recording is done in natural light illumination.

Table 2: Dataset Specifications

Signers	Age Group	Acquisition Tool	Resolution of Video	Background
Male: 04 Female: 06 Left and Right-Handed both included	18-25	Nikon Coolpix S9100 & Nikon Coolpix S9500	640 x 480 Frame Rate: 30FPS	White board Brown Notice board Blue cloth Yellow wall

Fig. 4 Sample frames of gesture sequences

III. IMPLEMENTATION

The system implementation unfolds in three major steps, each serving a specific purpose. The initial stage involves collecting and splitting the input gestures. To ensure a comprehensive dataset, we have gathered data from naturally deaf signers, including both male and female signers, as well as users with right and left hand dominance. Sample signs are exemplified in Figure 4. Once the gestures are inputted into the system, a training-testing split is performed using an 80:20 ratio. Table 3 provides insights into the number of video sequences and frames that are extracted and utilized for modeling. Frame extraction is accomplished using the open-source media tool FFMPEG [24], which offers robust support for multiple transmission protocols and various media containers. FFMPEG employs a unified data structure to store the extracted information. The working description of FFMPEG is illustrated in Figure 5, wherein the process is referred to as transcoding. FFMPEG leverages the libavformat library to read input files and obtain packets containing encoding data. In the case of multiple input files, FFMPEG ensures synchronization by tracking the lowest

timestamp on the active input video. These encoded packets are then passed to the decoder to generate uncompressed frames. If necessary, these frames can undergo further processing using filters. Post-preprocessing, they are fed into an encoder, which forms encoded packets. Finally, the output files can be accessed through the multiplexer. Within the FFMPEG framework, the invocation of a library called libavformat facilitates the utilization of a demultiplexer to read input files and extract packets containing encoded data. In scenarios involving multiple input files, FFMPEG further ensures synchronization by monitoring the lowest timestamp among active input videos. Subsequently, these encoded packets are forwarded to the decoder, responsible for generating uncompressed frames. If necessary, these frames can undergo further processing through the application of filters. Following the preprocessing stage, the frames are directed to an encoder, where encoded packets are once again formed. Finally, the output files can be accessed via the multiplexer.

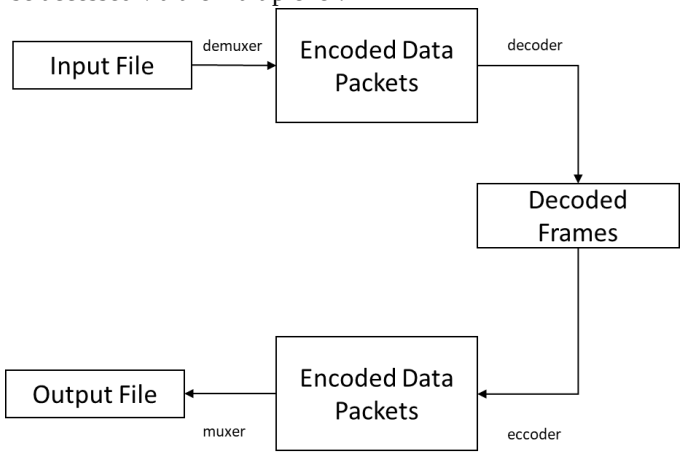

Fig. 5 Description of FFMPEG (https://ffmpeg.org/ffmpeg.html)

In the second stage, the extracted frames are utilized for feature extraction through the utilization of the Inception V3 convolutional neural network model. This stage incorporates the concept of transfer learning, where the pre-trained model weights from "imagenet" are employed to train the base model, considering the relatively smaller size of the dataset used. Table 4 summarizes the base model and total trainable parameters.

Table 3: Number of video gestures, gesture frames and classes

Training Videos	Validation Videos	Training Video Frame	Validation Video Frames	Output Classes
400	100	7365	1825	10

Table 4: Base Model Summary

Layer	Patch/Size/Stride
Input Layer	240, 320, 3,0
Conv	3 x 3/2
Conv	3 x 3/1
Padded Conv	3 x 3/1
Max_pooling	3 x 3/2
Conv	3 x 3/1
Conv	3 x 3/2
Conv	3 x 3/1
3 x Inception	--
5 x Inception	--
2 x Inception	--
Max_pooling	8 x 8
Dense/ Linear	1 x 1 x 2048
Total Parameters	21,802,784
Total Trainable Parameters	21,768,352

On top of the inception V3 model LSTM model has been applied for classification purpose in the last stage. The feature extracted in above models have not been classified in the same as we can see from above model summary. The classifier model details are as mentioned in Table 5.

Table 5: Classifier Model Summary

Layer	Output Shape
Output layer of CNN	6,8,2048
Flatten	98304
Dense Layer	1024
Dropout Layer	1024
Fully Connected Dense_1	1024

Fully Connected Dense_2	10
Total Parameters	123,526,954
Trainable Parameters	123,492,522

Experimental results are as seen in Table 6. The results depict various model parameters experimented and highlights the best results obtained.

Table 6: Comparison of Accuracy across various parameters

Training Samples	Validation Samples	Epochs	Batch Size	Training Accuracy	Validation Accuracy
7365	1825	20	32	89.03%	70%
7365	1825	20	64	93.4%	74.8%
7365	1825	40	64	69%	52%
7365	1825	50	32	78%	51%
7365	1825	50	64	97.1%	88.53%

Based on the above comparison, it is evident that the final model, trained with a batch size of 64 and over 50 epochs, achieved the highest validation accuracy of 88.53%. This model was evaluated using 30 signs from 10 distinct classes. The testing process encompassed gestures performed by signers from both the same database and other signers producing the same sign gestures. Performance parameters of the testing are as seen in Table 7.

Table 7: Performance parameters of test samples

Gesture	precision	recall	f1-score	support
Danger	0.667	0.667	0.667	3
Fire	0.667	0.667	0.667	3
Help Me	1.00	0.667	0.800	3
Hello	1.00	1.00	1.00	3
I am Hungry	1.00	1.00	1.00	3
India	1.00	1.00	1.00	3
Namaste	1.00	1.00	1.00	3
Please	0.750	1.00	0.857	3
Sorry	1.00	1.00	1.00	3
Thank You	1.00	1.00	1.00	3
Accuracy			0.900	30

Table 8 depicts the confusion matrix for the test samples with 27 True positives out of 30 total test samples, the accuracy of the system is as high as 90%. Thus, from the experimental results we can report that the system works well for dynamic sign gestures in case of variable background setups. Same dataset had been earlier checked in our previous work. In Table 9 we have compared the outcome of the comparison.

Table 8: Confusion Matrix of Test Samples

	Danger	Fire	Help Me	Hello	I am Hungry	India	Namaste	Please	Sorry	Thank You
Danger	2	1	0	0	0	0	0	0	0	0
Fire	1	2	0	0	0	0	0	0	0	0
Help Me	0	0	2	0	0	0	0	1	0	0
Hello	0	0	0	3	0	0	0	0	0	0
I am Hungry	0	0	0	0	3	0	0	0	0	0
India	0	0	0	0	0	3	0	0	0	0
Namaste	0	0	0	0	0	0	3	0	0	0
Please	0	0	0	0	0	0	0	3	0	0

Sorry	0	0	0	0	0	0	0	0	3	0
Thank You	0	0	0	0	0	0	0	0	0	3

Table 9: Comparison with Previous Work

	Methodology Used	Features Extracted	Accuracy	Limitation
(Badhe and Kulkarni, 2015) [4]	Combinational Algorithm Minimum Distance Classifier	2D- Fourier Descriptors quantized features	92.91 %	This system contains static as well as dynamic sign gestures.
(Badhe and Kulkarni, 2020) [2]	ANN with hand crafted features	2D- Fourier Descriptors as input neurons	63% Validation Accuracy	Due to smaller database, the system performance is poor, in spite of having good training accuracy, model overfits.
Proposed Methodology	Combined CNN and RNN	Spatio temporal features	90%	There is further scope of increasing the database and observe its effect on accuracy

IV. Conclusion

This paper presents a novel approach that combines Convolutional Neural Network (CNN) with Recurrent Neural Network (RNN) for sign language recognition. The experiment is conducted using a self-created database consisting of natural deaf signers. The feature extraction process utilizes the widely adopted Inception V3 model. To address the limitation of a relatively smaller database, transfer learning is employed, where pre-trained model weights are imported to effectively train the current system. Following successful feature selection, gesture classification is performed using the Long Short-Term Memory (LSTM) approach. By leveraging the combination of CNN and RNN, the proposed methodology takes advantage of both hierarchical and sequential features inherent in dynamic gesture datasets. The reported training accuracy of 97% and validation accuracy of 88.53% demonstrate that the system is comparable to existing methodologies in the same domain. Furthermore, the same database proves to be effective in other scenarios, such as with the minimum distance classifier and neural network classifier with handcrafted features, which yield promising results with a testing accuracy of 90% using the proposed methodology. As part of future work, the authors aim to enhance the database, increase the number of output classes, and observe the impact on accuracy.

References

[1] A. Mittal, P. Kumar, P. P. Roy, R. Balasubramanian and B. B. Chaudhuri. A Modified LSTM Model for Continuous Sign Language Recognition Using Leap Motion, in IEEE Sensors Journal, vol. 19, no. 16, 15th August 2019, pp. 7056-7063, doi: 10.1109/JSEN.2019.2909837

[2] P. C. Badhe and V. Kulkarni, "Artificial Neural Network based Indian Sign Language Recognition using hand crafted features," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225294.

[3] Wadhawan, A., Kumar, P. Sign Language Recognition Systems: A Decade Systematic Literature Review. Arch Computat Methods Eng (2019). https://doi.org/10.1007/s11831-019-09384-2

[4] P. C. Badhe, and V. Kulkarni, “Indian Sign Language translator using gesture recognition algorithm”, Proceedings of IEEE international conference on computer graphics on vision and information security (CGVIS), Bhubaneshwar, India, pp 195–200, 2015

[5] Cemil Oz, Ming C. Leu, American Sign Language word recognition with a sensory glove using artificial neural networks, Engineering Applications of Artificial Intelligence, Volume 24, Issue 7, 2011, pp 1204-1213

[6] A. Hamed, N. A. Belal and K. M. Mahar, "Arabic Sign Language Alphabet Recognition Based on HOG-PCA Using Microsoft Kinect in Complex Backgrounds," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, India, 2016, pp. 451-458, doi: 10.1109/IACC.2016.90.

[7] M. Alfonse, A. Ali, A. S. Elons, N. L. Badr and M. Aboul-Ela, "Arabic sign language benchmark database for different heterogeneous sensors," 2015 5th International Conference on Information & Communication Technology and Accessibility (ICTA), Marrakech, Morocco, 2015, pp. 1-9, doi: 10.1109/ICTA.2015.7426902.

[8] N. A. Sarhan, Y. El-Sonbaty and S. M. Youssef, "HMM-based Arabic Sign Language Recognition using Kinect," 2015 Tenth International Conference on Digital Information Management (ICDIM), Jeju, Korea (South), 2015, pp. 169-174, doi: 10.1109/ICDIM.2015.7381873.

[9] Zhong Y, Sun L, Ge C, Fan H. HOG-ESRs Face Emotion Recognition Algorithm Based on HOG Feature and ESRs Method. Symmetry. 2021; 13(2):228. https://doi.org/10.3390/sym13020228

[10] C. -H. Chuan, E. Regina and C. Guardino, "American Sign Language Recognition Using Leap Motion Sensor," 2014 13th International Conference on Machine Learning and Applications, Detroit, MI, USA, 2014, pp. 541-544, doi: 10.1109/ICMLA.2014.110.

[11] Miklas Riechmann, Ross Gardiner, Kai Waddington, Ryan Rueger, Frederic Fol Leymarie, Stefan Rueger, Motion vectors and deep neural networks for video camera traps, Ecological Informatics, Volume 69, 2022

[12] P. A. Nanivadekar and V. Kulkarni, "Indian Sign Language Recognition: Database creation, Hand tracking and Segmentation," 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), Mumbai, India, 2014, pp. 358-363, doi: 10.1109/CSCITA.2014.6839287.

[13] Wu, J., Sun, L., & Jafari, R. (2016). A wearable system for recognizing American sign language in real-time using IMU and surface EMG sensors. IEEE journal of biomedical and health informatics, 20(5), 1281-1290.

[14] Aryanie, D., & Heryadi, Y. (2015, May). American sign language-based finger-spelling recognition using k-Nearest Neighbors classifier. In 2015 3rd International Conference on Information and Communication Technology (ICoICT) (pp. 533-536). IEEE.

[15] Joudaki, S., & Rehman, A. (2022). Dynamic hand gesture recognition of sign language using geometric features learning. International Journal of Computational Vision and Robotics, 12(1), 1-16

[16] Chevtchenko, S. F., Vale, R. F., Macario, V., & Cordeiro, F. R. (2018). A convolutional neural network with feature fusion for real-time hand posture recognition. Applied Soft Computing, 73, 748-766.

[17] A. Kumar, A. S. C. S. Sastry, and P. V. V. Kishore. 3D sign language recognition using spatio temporal graph kernels, Journal of King Saud University – Computer and Information Sciences, 2019

[18] Rao, G. A., Syamala, K., Kishore, P. V. V., & Sastry, A. S. C. S. (2018, January). Deep convolutional neural networks for sign language recognition. In 2018 conference on signal processing and communication engineering systems (SPACES) (pp. 194-197). IEEE.

[19] Azar, S. G., & Seyedarabi, H. (2020). Trajectory-based recognition of dynamic Persian sign language using hidden Markov model. Computer Speech & Language, 61, 101053.

[20] Rahaman, M. A., Jasim, M., Ali, M. H., & Hasanuzzaman, M. (2020). Bangla language modeling algorithm for automatic recognition of hand-sign-spelled Bangla sign language. Frontiers of Computer Science, 14, 1-20

[21] Yao G, Lei T, Zhong J. A review of convolutional-neural-network-based action recognition. Pattern Recogn Lett. 2019;118:14–22.

[22] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). pmlr.

[23] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[24] Xu, Y., & Cao, S. (2015, September). Design and implementation of a multi video transcoding queue based on MySQL and FFMPEG. In 2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS) (pp. 629-632). IEEE.