Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 5, June 2021: 1331– 1344

Research Article

An Adaptive, Dynamic and Semantic Approach for Understanding of Sign Language based on Convolution Neural Network

Dr. A. Senthilselvi[1], Kirthika Sivakumar[2], M Reene Stephanie[3], Sweta Pasayat[4]

Abstract

Sign language has been one of the primary sources when it comes to the communication of people with hearing imparity. Just like every country has its own set of languages which people speak over there, similarly, there are different kinds of sign language which differs with regions and nationality. Many research and studies have been going on concerning the recognition of various kinds of sign language. With the help of sign language, people with hearing imparity can communicate with the rest of the society and convey their message, but the communication is not a two-way effective communication that has led to a barrier between us and the people with hearing imparity. Researchers are constantly trying to develop effective Sign Language Recognition (SLR) System. But all these systems that have been devised to date always come with a drawback and i.e., they have been only limited to isolated sign gestures. The main objective that we tend to achieve while devising this Sign Language Recognition System is that the system should assist in converting the input sign into its corresponding text or speech format automatically so that the rest of the society can effectively communicate with the people having hearing imparity and hence the barrier gets removed. Here in this paper, we propose a Sign Language Recognition System which is based on the Convolution Neural Network (CNN). The main purpose of using CNN over TSK is because it not only classifies and recognizes images but also does all the tasks with high accuracy, unlike the other proposed system. This particular technique will extract the important and useful information of the input sign language while ignoring the rest. Then these signed sequences will be modelled by the proposed mechanism and the output will be thrown in form of text or speech. CNN has been recognized as one of the major branches of Neural Networks. It uses multilayer perceptron which gives the system the ability to recognize and classify similarly as the brain does. Moreover, using this technique over all previous techniques will help us to have the least amount of pre-processing.

[1]SRM Institute of Science and Technology, Chennai, India. Email: mailmeselvi@yahoo.co.in
[2]SRM Institute of Science and Technology, Chennai, India. Email: kirthika.sivakumar1@gmail.com
[3]SRM Institute of Science and Technology, Chennai, India. Email: reenestephanie@gmail.com
[4]SRM Institute of Science and Technology, Chennai, India. Email: swetapasayat@gmail.com

A typical Convolution Neural Network has an input layer, multiple internally connected processing layers, and an output layer. All the images need to be trained through all these layers to achieve all the required results.

Index Terms: Convolution Neural Network (CNN), Hidden Markov Model(HMM), Neural Network, Sign Language Recognition (SLR) system, Long short-term memory(LSTM)

1. Introduction

As we all know there has been a great barrier in communication between the people with hearing imparity and the rest of the society and the only way through to this problem has been sign language. Sign language has made it easy for people with hearing imparity to communicate with others but still there has been a lot of problems in effective two-way communication. Although there has been many devised techniques and proposed model to overcome this hurdle still, we are only able to win over half of the problem. The devised techniques till now have been only efficient with isolated sign language whereas they do not produce great results when it comes to sequences of sign language. To overcome this particular hurdle, we have proposed a Long short-term memory model which will be able to recognise connected signed sequences and therefore the two-way communication will not be hindered anymore. In this particular model, we try to split the long sequences of sign language into subunits and then process them with neural networks using Convolution Neural Network. The most effective thing about pre-processing is that we do not need to train the system on the subunits while giving the input of different datasets which makes the pre-processing work easier and cheaper. The system that we propose here has been tested with 942 signed sentences which contain a combination of different words in Indian Sign Language. Moreover, these sentences could yield an output with just 35 signed gestures which were given to the system as trained datasets. In the currently proposed system, we aim at removing the redundant data at the input level itself. Whenever there is an input of signed language which needs to be converted to speech or text the system tends to remove unnecessary and redundant information such as joints of the wrist and hand. In this process only relevant information is produced at the end of the input level as all sorts of unwanted and repeated data is removed. Now, as we all know that hand span, palm size and hand movements differ while delivering sign language therefore, we perform few pre-processing steps so that we can overcome this challenge. For each input we tend to extract certain dynamic features from it, i.e., we extract coordinates of the fingertip. The main goal here is to cancel any such redundant and unnecessary information and minimize the count of dimensions in the feature space. Because of such steps, we will be able to minimize the computational cost and load as a result of which can increase the efficiency of the system. Reducing the count of dimensions in the feature space will help us overcome the curse of dimensionality. Adding the cherry on top it will also decrease the number of training dataset that we usually require to rain the system. Moreover, in real-time scenarios, this can lead to reducing the training time and we can meet the real-time requirements faster. This particular proposed system has many advantages such as low computation cost, high prediction accuracy, works well with optimization and adaptive techniques reduces response time, and improves performance.

2. Literature Survey

Sign language understanding helps to understand meaningful hand motions and is a critical component of intelligent contact between the deaf and hearing communities. However, we have some drawbacks in the current dynamic sign language recognition system, such as not being able to read difficult hand movements, low recognition accuracy and possible issues in training a bigger video sequence data. To address these drawbacks a BLSTM-3D Residual Network, a multimodal dynamic SLR system that is focused on a deep 3D Residual ConvNet and Bidirectional LSTM networks (B3D ResNet) is introduced in this paper. There are three main parts to this system. In the video frames, the hand object is localized to reduce the complexity of space and time made by network calculations. After feature analysis, the spatiotemporal features from the video are extracted automatically by the B3D ResNet and for every action in the video, an intermediate score is calculated by the B3D ResNet. In the end, by classifying the video sequence the complex sign language is correctly defined. The experiment was carried out on research datasets such as SLR Dataset and DEVISIGN D dataset. The results of the proposed approach showed industry-leading recognition accuracy. 86.9% accuracy on the SLR Dataset and 89.8% on the DEVISIGN D dataset. Furthermore, difficult hand movements in large videos can be recognized by the B3D ResNet and it can achieve high recognition accuracy. Since hand gestures are a long-term action, most existing sign language recognition systems that are dynamic cannot identify hand gestures that are similar and are focused on spatiotemporal features that are short-term. An innovative approach to interactive SLR based on video sequence is proposed as a solution to this issue. This approach is broken down into three parts. Based on a Faster R-CNN, the object localization module is the first component, and it captures information about the location of the hand. For feature extraction, the video frames are trained by Conv layers. Then on the final layer of the feature map, the proposals are mapped after they are submitted through the region proposal network. The ROI pooling layer then generates function maps with a fixed scale. The hand location could be correctly identified using the classifier. The feature vectors were obtained by using a B3D ResNet model to train full-time segmented images. The third part will be given each video function vector for analyzing complex information in sign language. The third element is interactive sign language, which is used to construct a shared representation for these separate sources. A recognition module capable of analyzing long-term temporal patterns predicts the hand gesture mark based on dynamics. Via research, the frames mark could be predicted for each video function vector. As a result, it was possible to predict the video sequence mark. This mark is then treated as the label of the video sequence and based on the scores from the top label prediction it is outputted as the recognition result. As a result, the difficult sign language can be understood effectively. Using the 3D convolution principle, the basic residual unit is updated to evolve 2D Residual Units into 3D architectures for encoding spatiotemporal video information. On each of these three channels, 3D convolutions with a kernel size of 333 are implemented separately. By adding residual connections to the 3D CNNs, the spatiotemporal features from the video sequence for SLLR in images are automatically extracted by the B3D ResNet model. [1]

In sign language, understanding human expressions is a complicated and time-consuming activity. Human sign language movements are made up of a variety of individual hand and finger articulations that are often coordinated with the body, face and head of a person. Recording 3D sign videos that are frequently influenced by inter-object are all part of 3D motion capture of sign language. This paper proposed that sign language movements expressed by different body parts such as the head, face, etc., are defined as 3D motionless, which are a subset of joint motions that represent the different signs. A two-phase quick algorithm is used to recognize 3D question signs. All joints in the human body are divided into motion joints (MJ) and nonmotion joints (NMJ) in the Phase-I process. The database is divided into four motionlets based on the relationship between the MJs and NMJs. The Phase-II method looks at the relationships between the movement joints to reflect a sign's shape details as 3D motionlets. Three adaptive motionlet kernels are included in the 4-class sign database. The database is ranked using a simple kernel matching algorithm based on the highest-ranking query symbol. The proposed method can characterize sign language using a 3D spatiotemporal structure and it is sign invariant to misalignment caused by the temporal. To test this proposed system five 500-word Indian sign language datasets were used. The results showed that the proposed system improved the recognition. Body movements that are either autonomous or body movements that depend on another body part are used in sign language. As a result, sign representations include both moving and non-moving human pieces, with a strong degree of similarity between them. The use of 20–25 marker models or joints for human motion representation and analysis has been recorded in the literature. These versions do not have fingers or a face. Finger and face skeletal representation is an essential part of SLR. A 57-marker prototype that has a precision placed marker locations built-in is used to capture the Indian sign language at its full potential. The entire dataset is then categorized, and appropriate labels are assigned. To find out where the hand is exactly pointing on the forehead, we use GJRD values. Every dataset label has a collection of signs associated with GJRD values. Without understanding the finger shapes, however, recognizing a symbol is difficult. To represent the shape of the hand as a motionlets, this necessitates a local distribution of finger joints concerning adjacent fingers. Small motions of the fingers the orientations per frame are evaluated to recognize the shape and to classify it. In this experiment five 500 signs from five different signers are. One of the signers is a native sign language translator who can provide flawless gestures; the four others are seasoned sign language interpreters. These five signers have different body sizes and different hand gestures, as well as the articulation of signs that match. This allows the proposed algorithm's robustness to be tested using 3D sign language motion capture data. Recognition, precession, and recall are used as regular success metrics. Analyzing the advantages and disadvantages of the approach to other models used exclusively for SLR can be achieved by analyzing the individual categories of symptoms. [2]

Deaf people use Sign Language as one of their primary modes of communication. To communicate with them, one should learn sign language. Peer learning is the most common form of training. There are very few research materials available for learning signs. As a consequence, it is difficult to learn a sign language. Spelling the letters one by one using fingers is the first part of learning sign language and it is often used when there is no accompanying sign or when the signer is unaware of it. The majority of current sign language learning tools rely on expensive external sensors. In this paper, we explore a program called SignQuiz, a low-cost web-based fingerspelled sign learning program that uses an automatic sign language recognition technology to teach Indian sign language (ISL). SignQuiz aids in the learning of signs without the use of external resources. This is the first attempt in ISL to use a deep neural network to learn finger-spelled signs. The findings show that SignQuiz is superior to the written medium for learning fingerspelled signs. The center of SignQuiz is Automatic Sign Language Recognition (ASLR). Our model is tuned to detect ISL signals using transfer learning. Transfer learning allows you to practice on new courses even if the training set is not large. The weights of a previously trained model are used to initialize the mechanism instead of using random weights to begin with and it is a safer starting for preparation. Transfer learning allows us to do ASLR quickly by using an already existing model and changing the weights of it according to what we need. The user is supposed to make a sign and capture a photo. When the user takes the photo, the machine records the sign displayed by the user and offers input about whether it is right or incorrect after a few seconds. If the user is asked to show the sign “Hi” and he does so correctly, the user will receive the feedback ‘‘Hi detected.”. If the user shows a different sign instead, the user will receive a message "(name of sign) detected.”. The group length in SignQuiz is five. After the completion of each set, the user receives a completion message, and a new set will be created. The learning process is stored in the signs list, which contains a list of alphabets to be learned. The first element from that list is transferred to the study list, where learning takes place. The current alphabet is stored in variable current, and the sign recognized by the classifier is stored in the variable user selected. The accuracy threshold used for detection is stored in the parameter threshold. [3]

In this paper, a glove-based method is proposed to recognize the Arabic sign language, which employs a novel sequential data classification technique. A dataset of 40 sentences is built based on a sensor using an 80-word lexicon. Using two DG5-VHand data gloves the hand movements are registered in the dataset. A camera is used to match the hand gestures to its respective word during data marking. A low-complexity pre-processing and feature extraction techniques are used to emphasize and capture the data’s temporal dependency. Then a Modified k-Nearest Neighbour (MKNN) approach is used to do the classification. The proposed solution obtained a 98.9% sentence comprehension score. The findings are compared to those of a previous vision-based system that had the same set of input sentences. The proposed system proved to be better than the previously existing systems in terms of classification rates thus removing their limitation. Continuous marks that are similar are then grouped into sign language terms after a collection of feature vectors has been categorized. Post-processing can be used at this stage to enforce rules on the expected terms and sentences. This research employs two laws. The first is concerned with the smallest number of feature vectors that make up a word, while the second is concerned with the identification of repeated terms. If the length of a word is 3, for example, recognizing a word requires a series of at least three identical labels. The second law, on the other hand, forbids consecutive repetition of a word in a sentence. The goal of this is to preserve the long-term trends while minimizing the short-term fluctuations. As a result, we get a smoother version of the input result and it’s referred to as a low pass filter. Each function vector may provide contextual information that aids incorrectly classifying it. Resampled versions and raw sensor readings were tested without feature extraction. The best results for classification rates with various ModeW, MedW, and WordTH settings are presented. The best result for sentence recognition was 82% using the original set of feature vectors. With a standard deviation of 4.88, this precision is based on an average of 3 testing rounds. The rate at which a collection of feature vectors is correctly categorized into a single word is referred to as the word recognition rate. The term "feature vector recognition rate" refers to the accuracy with which a feature vector is classified, regardless of the sentence and word recognition rates [4]. Renith and Senthilselvi [18] used deep learning network for diabetic retinopathy disease detection. Surya and Senthilselvi [19] explained food adulteration detection using machine learning. Mohammed taha et al. [20] explained brain tumour detection and classification using Convolutional Neural Network (CNN). Senthilselvi et al. [21] explained metal purification enhancement using deep learning network. Nivetha and Sethilselvi [22] explained feature extraction and feature matching process in forgery detection.

3. Proposed System

Figure 1 shows the architecture diagram for the proposed system.

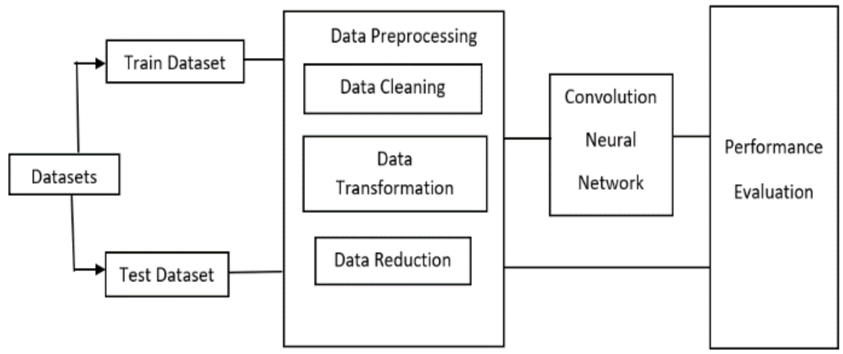


Fig. 1 Architecture diagram

A. Image Preprocessing

Low-pass filtering is generally considered as the method of removing all the components of the signal that contain high spatial frequencies and thus, it is very much common to implement this process in the domain of spatial frequency. In spite of that, it is definitely possible to carry out this process quite directly in the spatial domain itself. In this spatial domain, if the convolving function is significantly narrow, then the huge amount of computation is significantly reduced: hence the efficient working of the low-pass filter is possible. Thus, it is just the convolving function that is more suitable is to be found.

The array of the histogram is first initialized by clearing the array and one of the images is scanned. This will now generate a new image in the existing space, then the intensity values produced are created in the form of a histogram for each and every neighborhood; then the median needs to be calculated. Then finally, the particular incremented points are cleared or removed from the histogram. Here, the overall computation is saved as the last feature eradicates the clearing of the entire histogram.

B. Feature Extraction

Inverse transformation of any image is obtained by performing linear transformation. The main goal of this Image inverse transformation is to transform input images containing dark intensities to output images which are inversely transformed to bright intensities and the converse is also true.

To extract the smaller portions of the image, some of the operators are used, for instance, down-sampling or windowing. Few operators can induce any sort of change in the shape of the content by the process of bending, swapping, rotating, scaling, shifting, or affine transformations. One of the operators called the Concatenation operator involves in combining smaller images to create a new larger image. Though the rule of alteration of the pixel values of the existing content is broken by this operator, instead of using a user-defined operation, this is achieved by merging the given set of input images.

C. Model Prediction

An image can be differentiated by various blocks and each block of an image is a collection of pixel cells. The Kernel works by striding over the given image, block by block. In this process, images having lower resolutions are obtained as a result of performing matrix multiplication. By finding the average pixel value also known as average pooling or the maximum pixel value known as max pooling, an image with lower resolution is obtained in the subsampling layer or the down-sampling layer. In the end, the output is finally connected to the already fully connected layer. In this fully connected layer, every output of max-pooling is connected to each of the nodes. This image processing is performed over multiple layers. Figure 2 shows the working of CNN.

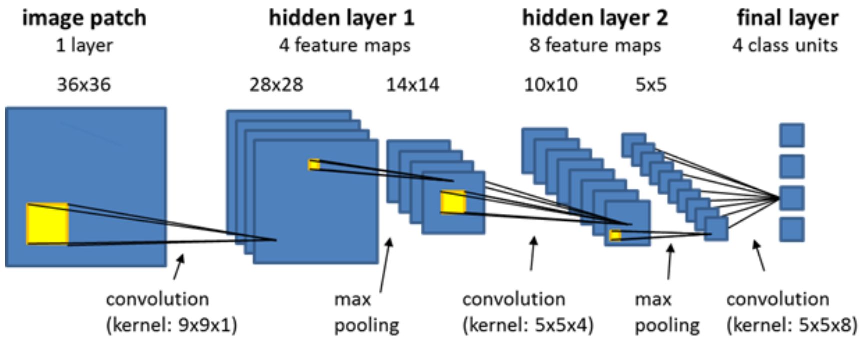


Fig. 2. Working of CNN

In the first layer, edge detection is applied, wherein the edges are detected and the templates are built. The layers that come next, utilize the templates formed by the first layer for the base and takes the simpler shapes from the image. This layer also creates more such templates that are inclusive of different positions, illuminations or scales. In the last layer, the input images are matched with all the above templates and the weighted sum of all the outputs becomes the final output. This is quite helpful when it comes to handling images with higher accuracy having variations that are complex. Figure 3 shows the output of CNN Layer 1 and figure 4 shows the output of CNN Layer 2.

D. Technology Used

Backend Technologies

- Python
- Numpy
- Sci-learn
- Eclipse IDE
- Keras
- Tensorflow

Frontend Technologies

- Web Technologies
- Jupyter Notebook

E. Algorithm of the Proposed Method

1. Start
2. Firstly, the required libraries are imported from various sources in python like cv2, pickle, glob, numpy, sklearn, tensorflow, keras and matplotlib.

3. Then the functions are declared for displaying various images and labels to be displayed during output in the console.
4. Now, the images and labels in the datasets are trained, that is, 80% of datasets are trained.
5. Validation of datasets is done during the training of datasets, that is, 10% of the dataset available for training.
6. Then the datasets are tested, that is, 20% of the datasets are tested.
7. A function is declared for displaying the green rectangular box inside which we need to show the sign language symbol during the implementation of the program.
8. The histogram array is initially empty and once we show any sign language symbol in front of the camera, it starts reading, and then image resizing, and processing is done.
9. Now, in the CNN model, there is one input layer, few hidden layers and one output layer. All image transformations are done in this model.
10. Finally, when the project is implemented, in the console output, all the datasets are loaded, the CNN layer operations are shown, the loss and accuracy are tested for each dataset.
11. After all the loading process, once the camera is turned on by the program, we can show various sign language symbols inside the green box and whatever we are showing can be seen on the histogram with inversed color, and finally, the output text is shown on the console.
12. End

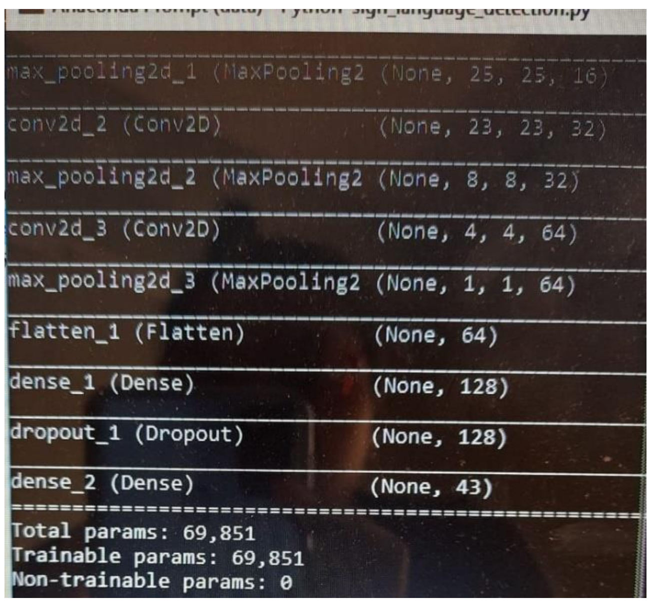
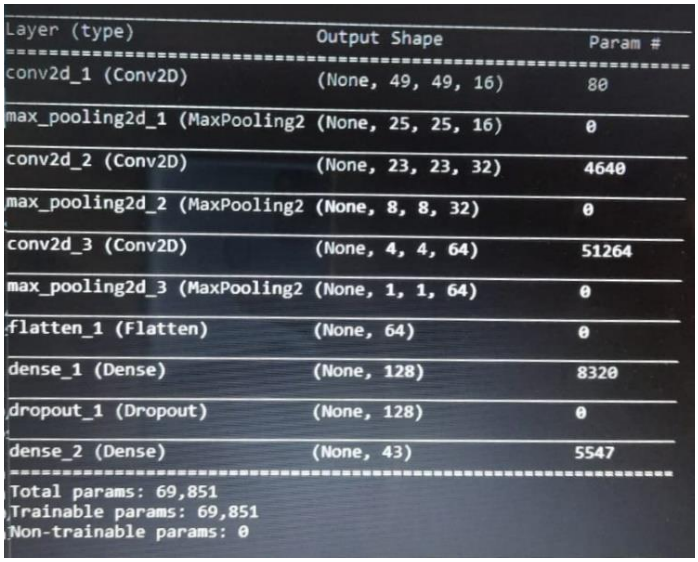


Fig. 3 CNN Layer 1



Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 49, 49, 16)	80
max_pooling2d_1 (MaxPooling2	(None, 25, 25, 16)	0
conv2d_2 (Conv2D)	(None, 23, 23, 32)	4640
max_pooling2d_2 (MaxPooling2	(None, 8, 8, 32)	0
conv2d_3 (Conv2D)	(None, 4, 4, 64)	51264
max_pooling2d_3 (MaxPooling2	(None, 1, 1, 64)	0
flatten_1 (Flatten)	(None, 64)	0
dense_1 (Dense)	(None, 128)	8320
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 43)	5547

Total params: 69,851
Trainable params: 69,851
Non-trainable params: 0

Fig. 4 CNN Layer 2

E. Performance Measure

Accuracy calculation:

Accuracy =$(True_{positive} + True_{negative}) / (True_{positive} + True_{negative} + False_{positive} + False_{negative})$

In proposed model,
Total test cases = 45
Positive class = 40
Negative class = 5

The proposed model makes a total of 39/40 correct predictions for the positive class compared to just 4/5 predictions for the negative class.

Accuracy = (39+4) / (39+4+1+1)
= 43/45
= 95.55%

G. Result and Discussion

Table 1: Comparative analysis between the existing LSTM model and the Proposed CNN model

Model	**Accuracy**
Existing LSTM	89.50%
Proposed CNN	95.55%

From the above table, it is clearly shown that there is a huge difference between the accuracy of the existing LSTM model and the proposed CNN model. The reason for this difference in accuracy is because the LSTM method had a huge computational load which took a lot of time for recognition and feature extraction. Also, since LSTM used fuzzy logic, it requires more fuzzy grades for attaining better accuracy which led to exponentially the rule. But in the case of CNN, there are lots of layers, but all of these layers including the input and output layer, are hidden. Hence it has less computation, takes less time to extract features and produces more accurate results than the LSTM model. Figure 5 shows the output for the number 4, figure 6 shows the output for the alphabet B, figure 7 shows the output for the alphabet C and figure 8 shows the output for the number 2.

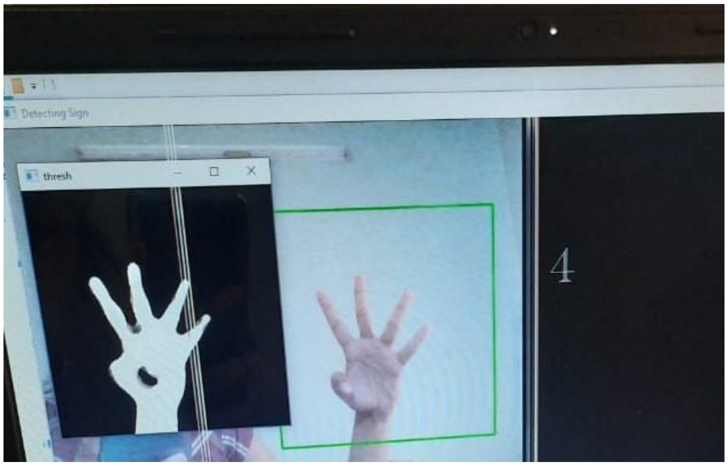


Fig. 5 Output for 4

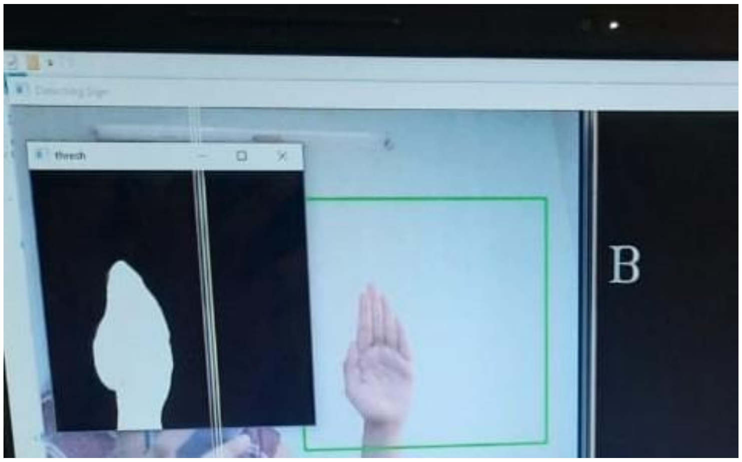


Fig. 6 Output for B

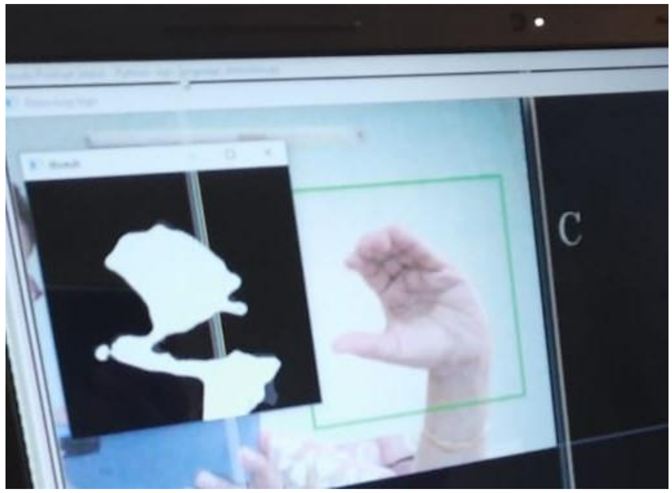


Fig. 7 Output for C

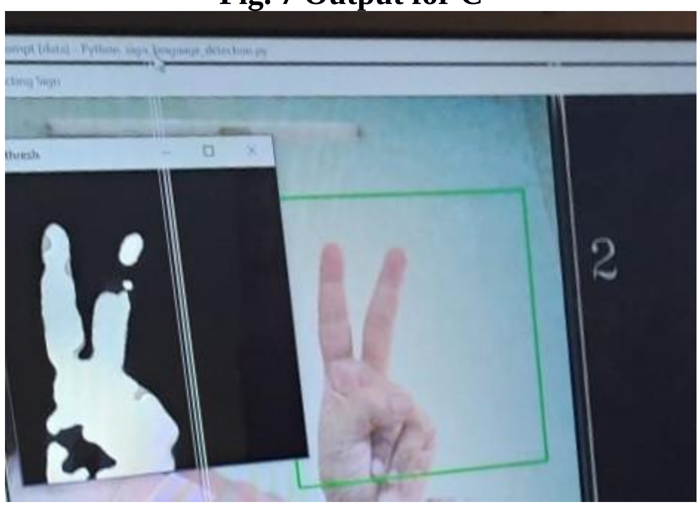


Fig. 8 Output for 2

4. Conclusion and Future Enhancements

In this paper, an architecture has been proposed in order to construct a system for recognition. The recognition model that is proposed is established by CNN and it is quite convenient to model the time-series data. Generally, any sign language portrays a particular meaning by analyzing various continuous actions that come under time-sequence data. There are two major layers in which the lower layer processes the component channels and gives output to the upper layer which performs sign recognition. In this approach, before the recognition process, the segmentation of various signed sentences takes place. In this system, for the continuous-SLR, a novel framework has been presented. For recognizing sign letters, a CNN architecture has been proposed. Also, while training the model, isolated sign words are utilized in the datasets.

Currently, this project has succeeded in providing a system or a prototype in which the following results have been achieved:

- Automatic detection of important features

- Computational efficiency
- Weight sharing
- Great image handling processing

Though these results have been achieved by this project, the proposed system lacks in achieving high accuracy in detection. Accuracy is one of the most important features to consider as having less or even little inaccuracy could lead to miscommunication which could result in delaying the other person to comprehend what the person had said. In case a person is calling someone out for help or in case of emergency, having a system that is not accurate enough may lead to critical situations. Hence, as a future enhancement, the overall accuracy of the proposed work could be improved for better results.

References

[1] Dynamic Sign Language Recognition Based on Video Sequence with BLSTM-3D Residual YANQIU LIAO, PENGWEN XIONG, WEIDONG MIN, (Member, IEEE), WEIQIONG MI , and JIAHAO LU

[2] Motionlets Matching with Adaptive Kernels for 3D Indian Sign Language P.V.V. Kishore, Senior Member, IEEE, D.Anil Kumar, Student, IEEE, A.S.C.S.Sastry, Member, IEEE, and E.Kiran Kumar,

[3] SignQuiz: A Quiz Based Tool for Learning Fingerspelled Signs in Indian Sign Language Using ASLR JESTIN JOY, KANNAN BALAKRISHNAN, AND SREERAJ M

[4] Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode Noor Tubaiz, Tamer Shanableh, Member, IEEE, and Khaled Assaleh, Senior Member, IEEE

[5] G. F. Simons and C. D. Fennig, Ethnologue: Languages of the world. Dallas, TX, USA: SIL, 2017.

[6] L. de Marcos, E. Garcia-Lopez, and A. Garcia-Cabot, ‘‘On the effectiveness of gamelike and social approaches in learning: Comparing educational gaming, gamification social Networking,’’ Comput. Edu., vol. 95, pp. 99–113, Apr. 2016

[7] S. Ebling et al., ‘‘Smile swiss german sign language dataset,’’ in Proc. LREC, Aug. 2018, pp. 19–25.

[8] Y. Bouzid, M. A. Khenissi, F. Essalmi, and M. Jemni, ‘‘Using educational games for sign language learning—A signwriting learning game: Case study,’’ Educ. Technol. Soc., vol. 19, no. 1, pp. 129–141, Jan. 2016.

[9] O. Koller, J. Forster, and H. Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Computer Vision and Image Understanding, 141:108–125, 2015

[10] W. Kong and S. Ranganath. Towards subject independent continuous sign language recognition: A segment and merge approach. Pattern Recognition, 47(3):1294–1308, 2014. 7

[11] P. Kumar, H. Gauba, P. P. Roy, and D. P. Dogra. Coupled hmmbased multi-sensor data fusion for sign language recognition. Pattern Recognition Letters, 86:1–8, 2017. 1, 2

[12] P. Kumar, H. Gauba, P. P. Roy, and D. P. Dogra. A multimodal framework for sensor based sign language recognition. Neurocomputing, 2017.

[13] P. Kumar, R. Saini, P. Roy, and D. Dogra. Study of text segmentation and recognition using leap motion sensor. IEEE Sensors Journal, 2016.

[14] P. Kumar, R. Saini, P. P. Roy, and D. P. Dogra. 3d text segmentation and recognition using leap motion. Multimedia Tools and Applications, pages 1–20, 2016

[15] C.-H. Chuan and C. A. Guardino, ‘‘Designing smartsignplay: An interactive and intelligent american sign language app for children who are deaf or hard of hearing and their families,’’ in Proc. 21st Int. Conf. Intell. User Interfaces. New York, NY, USA: ACM, 2016, pp. 45–48. doi: 10.1145/2876456.2879483.
[16] S. K. Behera, D. P. Dogra, and P. P. Roy. Analysis of 3d signatures recorded using leap motion sensor. Multimedia Tools and Applications, pages 1–26, 2017
[17] J. Gameiro, T. Cardoso, and Y. Rybarczyk, ‘‘Kinect-sign: Teaching sign language to ‘listeners’ through a game,’’ in Proc. Int. Summer Workshop Multimodal Inter. New York, NY, USA, Springer, 2013, pp. 141–159.
[18] G. Renith, A Senthilselvi , "Accuracy Improvement in Diabetic Retinopathy Detection using DLIA", Journal of Advanced Research in Dynamical and Control Systems, titled Volume 12, Issue 7, July 2020
[19] V. Surya, A Senthilselvi , "A Qualitative Analysis of the Machine Learning Methods in Food Adultery: A Focus on Milk Adulteration Detection", Journal of Advanced Research in Dynamical and Control Systems, titled Volume 12, Issue 4, May 2020
[20] Mohammed Thaha .M, Pradeep mohankumar .K Murugan .B.S, Dhanasekar .S, Vijay Karthick , P,Senthilselvi.A "Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images",Journal of Medical Systems, Volume 43 No 9 July 2019,
[21] A. Senthilselvi , V. Sellam , Saad Ali Alahmari , Sivaram Rajeyyagari ," Accuracy enhancement in mobile phone recycling process using machine learning technique and MEPH process", Environmental Technology & Innovation,volume20,November2020.
[22] R Nivetha, A Senthilselvi, "Hybrid Feature Matching for Image Forgery Detection",International Journal of Engineering Science and Computing, volume 7, issue 3, p. 5075 – 5080, 2017

Authors' Profiles

Dr. A. Senthilselvi
Associate Professor in the Department of CSE at SRM Institute of Science and Technology, Chennai, India. She received her B.E degree in Electronics and Communication Engineering from Anna University, Tamilnadu, India, in 2005. She has received her M.Tech degree in Information Technology from Kalasalingam University, Tamilnadu, India, in 2010. She has completed her PhD degree in 2019 at Anna University, India.

Kirthika Sivakumar
UG Scholar, B Tech, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India.

M Reene Stephanie
UG Scholar, B Tech, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India.

Sweta Pasayat
UG Scholar, B Tech, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India.