Neural Computing and Applications
https://doi.org/10.1007/s00521-020-05448-8

ORIGINAL ARTICLE

Benchmarking deep neural network approaches for Indian Sign Language recognition

Ashish Sharma[1] · Nikita Sharma[1] · Yatharth Saxena[1] · Anuraj Singh[1] · Debanjan Sadhya[1]

Received: 23 April 2020 / Accepted: 14 October 2020

Abstract
Sign language is the language of the deaf and mute. However, this particular population of the world is unfortunately overlooked as the majority of the hearing population does not understand sign language. In this paper, an extensive comparative analysis of various gesture recognition techniques involving convolutional neural networks and machine learning algorithms has been discussed and tested for real-time accuracy. Three models: a pre-trained VGG16 with fine-tuning, VGG16 with transfer learning and a hierarchical neural network were analyzed based on a number of trainable parameters. These models were trained on a self-developed dataset consisting images of Indian Sign Language (ISL) representation of all 26 English alphabets. The performance evaluation was based on the practical application of these models, which was simulated by varying lighting and background environments. Out of the three, the hierarchical model outperformed the other two models to give the best accuracy of 98.52% for one-hand and 97% for two-hand gestures. Thereafter, a conversation interface was built in Django using this model for the real-time gesture to speech conversion and vice versa. This publicly accessible interface can be used by anyone who wishes to learn or converse in ISL.

Keywords Indian Sign Language recognition · Convolutional neural networks · Hierarchical network

1 Introduction

In this fast-growing world, there is an urgent need to uplift the challenged sections of the society. People suffering from speech disabilities communicate in sign language and therefore face trouble in connecting with the able-bodies. Thus, there exists a communication gap that is difficult to mitigate into the mainstream society. According to the estimation of the Indian National Association of the deaf, around a million people suffer from some form of functional hearing loss[1]. This research work targets the hearing impaired population of India. It attempts to develop a machine learning-based conversation interface that uses both one-hand and two-hand gestures for communication. This work can be used for assisting differently-abled people in conversing with others by using Indian Sign Language (ISL).

ISL is a two-hand gesture language. It is a language that is spoken with the facilitation of hand movements, facial expressions and body language. There exist many languages and dialects in India since it is a diverse country. However, the countrywide accepted sign language remains constant. Some of the major aspects of ISL are finger-spelled, which makes the use of finger gestures as alphabets to spell the words. The set of pre-defined gestures involve movements of hands, their forms and positions for conveying information [19]. Face gesture is another technique that involves the movement of eyebrows, mouth, head tilt and countenance to express the message. Lastly, body

✉ Debanjan Sadhya
debanjan@iiitm.ac.in

Ashish Sharma
ipg_2016021@iiitm.ac.in

Nikita Sharma
ipg_2016062@iiitm.ac.in

Yatharth Saxena
ipg_2016121@iiitm.ac.in

Anuraj Singh
anuraj@iiitm.ac.in

[1] ABV-Indian Institute of Information Technology and Management Gwalior, Gwalior, India

[1] http://nadindia.org/.

Published online: 24 October 2020

gesture is another method that requires shifting and positioning of the entire body for explaining the message. Due to the diversity and multilingual culture in India, various regions or states/cities have different gestures in sign language for communication [30]. In addition, some words (including proper nouns and uncommon words) do not have pre-defined gestures, which urges the need for a descriptive ISL interface.

There have been many studies done on the sign languages of countries other than India. Although these works are currently at advanced stages, the ISL recognition problem is still at a primitive stage [9]. A good number of studies have been published on American Sign Language gesture recognition which gives a decent accuracy. However, one can find very few works on gesture recognition of ISL. A significant factor that acts as a barrier in the development of ISL recognition is that it involves one-hand as well as two-hand gestures. Naturally, this becomes a complex task for currently available networks due to feature occlusion. The problem of sign language recognition is complex because it involves 26 categories ('A'–'Z') with very similar images for each class. This specific problem inhibits the development of a robust and scalable system. Also, various factors like human errors and false predictions bring up a need to use an algorithm for correcting falsely predicted alphabets in words. We have discussed one such method, which is based on the Breadth First Search (BFS) algorithm and the English corpora.

1.1 Contributions

The study aims to improve ISL gesture recognition under constrained conditions. The novelty of this work is to classify and recognize all the 26 English alphabets based on Indian Sign Language using three different kinds of approaches: (1) pre-trained VGG-16 model (both transfer learning and fine tuning applied separately), (2) hierarchical model and (3) natural language-based model. Importantly, the third model can generate the most probable word when a sequence of gestures is fed to it. The problem of feature occlusion when segregating one- or two-hand gestures is also addressed in this work by using HOG features. This entire approach is implemented on a Python-based web interface for creating a real-time translator from ISL to audio and vice versa.

2 Related work

Sign Language is the mode of communication for the hearing impaired people of the society. Various researchers have worked on sign languages of their respective countries, including American [10, 16, 29], Chinese [13, 27], Finnish [8], British [20], Italian [15], Ukrainian [11] and Arabic [12, 25]. Beena et al. [4] worked on American Sign Language using the convolutional neural network and support vector machines. They utilized the HOG, LBP and 3D-voxel techniques for extracting the relevant features. Similarly, Quesada et al. [16] worked on the automatic recognition of the American sign language finger-spelling alphabets. The authors presented a system based on hand tracking devices (Leap Motion and Intel Real Sense) and used SVM to classify the gestures. Some of these gestures even gave an accuracy of 100%. Kong and Ranganath [10] applied rule-based segmentation to segment the hand motion trajectories of signed sentences. Subsequently, these segments were clustered using k-means for deriving the phonemes. Luqman et al. [12] proposed Arabic text to Arabic sign language translation using a rule-based translation system. They also included a parallel corpus in the health domain consisting of 600 sentences. They obtained the translation accuracy of more than 80% for the translated sentences.

Some generic studies on sign languages were performed by Stein et al. [24] and Boulares et al. [5]. The authors analyzed existing sign language data and emphasized its quality and usability for the statistical machine translations. For data pre-processing of the sign language corpus, the authors introduced sentence end markers, split compound words and handled parallel communication channels. Moreover, they focused on optimization procedures such as scaling factor optimization and alignment optimization. Hardware device, namely kinetic sensors (by Microsoft), creates a 3D model of the hand and observes the hand orientation and movements. This model was compared against pre-stored models in a database. Another utilized method was the Glove-based approach wherein the user was required to wear a special glove that recognized the position and orientation of the hand. Although this technique was highly accurate, the initial setup cost was relatively high. A compilation of various sign language recognition techniques using hand gestures is presented by Cheok et al. [6].

Predicting ISL gestures still remains a big question in the domain of computer vision; the primary reason being it is a set of both one-hand and two-hand gestures. This property causes the problem of feature occlusion and hence acts as a barrier in the development of ISL [23]. Many researchers have tried various approaches for gesture recognition of ISL. Some of the more important works are explained as follows. Raheja et al. [18] worked on the recognition of four gestures namely 'A,' 'B,' 'C' and 'HELLO.' Extraction of the Hu-Moments and motion trajectory was done in this work, which resulted in an accuracy of 97.5%. In another work, Ansari and Harit [3] implemented a functional unobstructive ISL recognition system on the vocabulary of 140 symbols collected from 18 subjects. However, they

worked only on two-hand gestures and excluded the study of one-hand gestures. Their results were 90% accurate for 13 alphabets and 100% accurate for three alphabets, with an overall accuracy of 90.68% on 16, one-hand alphabets. Hore et al. [9] proposed three frameworks for ISL recognition based on Genetic Algorithm (GA), Evolutionary algorithm (EA) and Particle Swarm Optimization (PSO). Although the best accuracy of 99.96% was reported for the PSO-based approach, the simulations were performed on a relatively small dataset of 22 characters.

Some important contributions to the field of ISL biometrics were made in [14, 23]. Patil et al. [14] worked upon a robust biometric method for communication for the hearing and speech impaired people. The database was tested using the SIFT (Scale-Invariant Feature Transform) features, which gave comparatively good results. In the context of this research, it can be stated that an appropriate optimization tool that incorporates more number of languages could benefit speech and hearing people of the word as a whole. A robust modeling of static signs in the context of ISL was performed by Wadhawan and Kumar [26]. The authors utilized deep learning-based convolutional neural networks over 35,000 sign images of 100 static signs. The highest training accuracy was reported to be 99.72 and 99.90% on colored and grayscale images, respectively. Agrawal et al. [1] fused the HOG and SIFT features for recognizing two-handed ISL. An overall accuracy of 93% was reported on the Multi-class Support Vector Machine (MSVM) classifier. Most recently, Raghuveera et al. [17] utilized the depth and color information of hand gestures (captured through Kinect Xbox 360) for ISL recognition. The authors used an ensemble of Speeded Up Robust Features (SURF), HOG and LBP features to train an SVM. The average recognition accuracy was reported up to 71.85%.

3 Methodology

In this section, we discuss the architectures of various pre-trained deep neural networks and machine learning algorithms. We also state their performances for the task of hand gesture to audio conversion and vice versa. The complete implementation was done on *Keras* using *Tensorflow* as the backend. A pictorial overview of our entire framework is presented in Fig. 1. The three individual models are briefly discussed as follows.

- Pre-trained VGG16 Model: Under this approach, the gestures were classified using a pre-trained VGG16 model based on the Imagenet dataset. We truncated its last layer and then added custom designed layers to provide a baseline comparison with the state of the art networks.
- Natural Language-Based Output Networks: For this model, a Deep Convolutional Neural Network (DCNN) with 26 categories was developed. Later, the output was fed to an English Corpora-based model for eradicating any errors during classification. This process was based on the probability of the occurrence of the particular word in the English vocabulary. Only the top three accuracy scores provided by the neural network were considered in this model.
- Hierarchical Network: Our final approach comprises of a novel hierarchical model that resembles a tree-like structure. It involves initially classifying gestures into two categories (one-hand or two-hand) and subsequently feeding them further into the deep neural networks. The corresponding outputs were utilized for categorizing them into the 26 English alphabets.

3.1 Dataset

According to the best of the knowledge of the authors, there does not exist an authentic and complete dataset for all the 26 alphabets of the English language for ISL. As such, we manually prepared our dataset by clicking various images of each finger-spelled alphabet and applying different forms of data augmentation techniques. In the end, the dataset contained over 150,000 images of all 26 categories. There were approximately 5500 images of each alphabet. The same background was used for most of the images for keeping the data consistent. Furthermore, the images were clicked in different lighting conditions to train a robust model that was resistant to any such changes in the surroundings. The images in this dataset were clicked by a Redmi Note 5 Pro, 20-megapixel camera. All the RGB images were resized to 144×144 pixels per image to remove the possibility of varying sizes. Figure 2 shows a few sample images from this dataset.

3.1.1 Data augmentation

Data Augmentation is a necessary step to avoid over-fitting and to ensure that the model performs well in different lighting conditions. Seven batches of the original dataset were created in order to perform different augmentations. Rotation by an angle $\theta \in \{10°–20°\}$ was performed on the first batch, whereas horizontal flipping was performed on the second. Similarly, feature-wise central alignment and standard normalization, shearing by 0.01–0.02, height–width shift by 0.01°–0.02°, zoom to the range 0.9–1.25 and brightness range of 0.5–1.5 were performed on the consecutive seven batches. Figure 3 depicts the images after various augmentations were performed in batches.

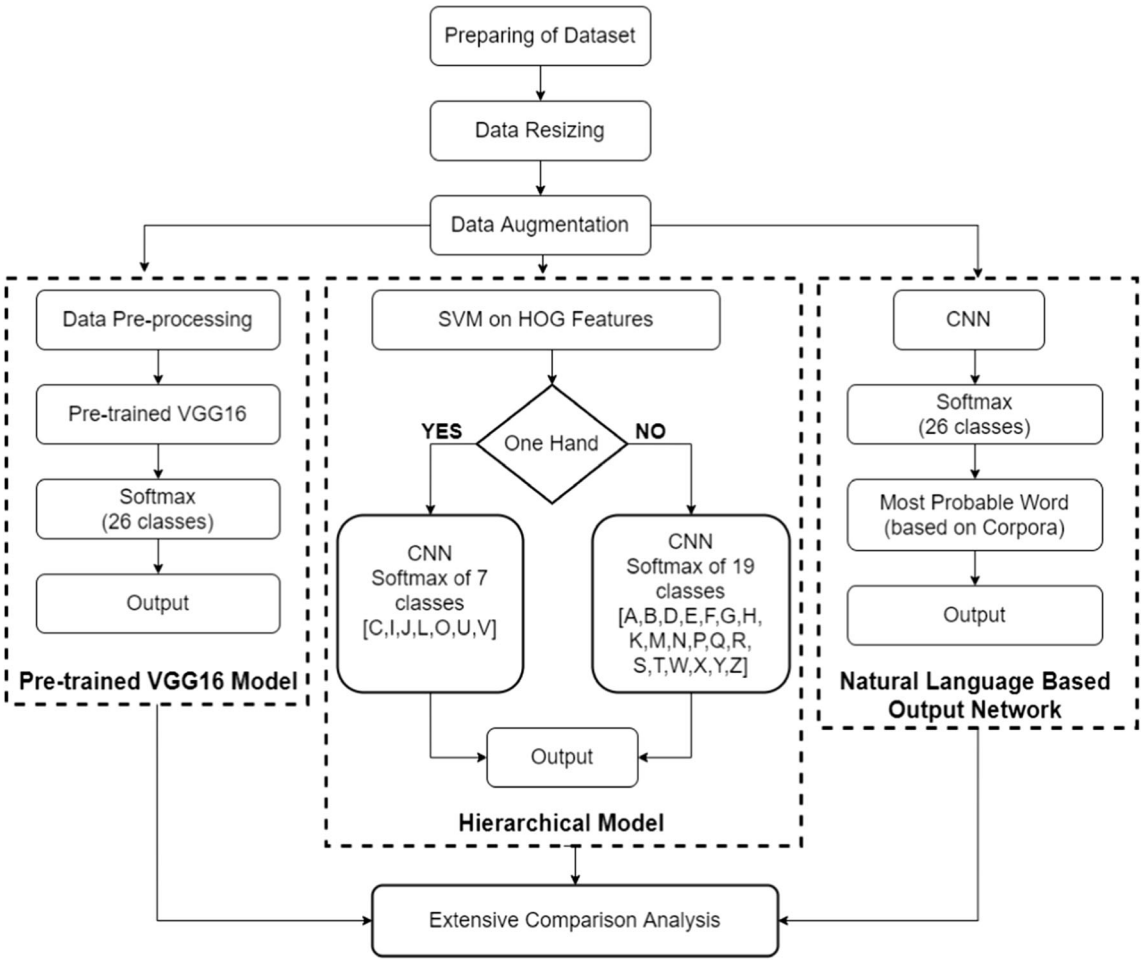


Fig. 1 An overview of the proposed framework involving Indian Sign Language recognition

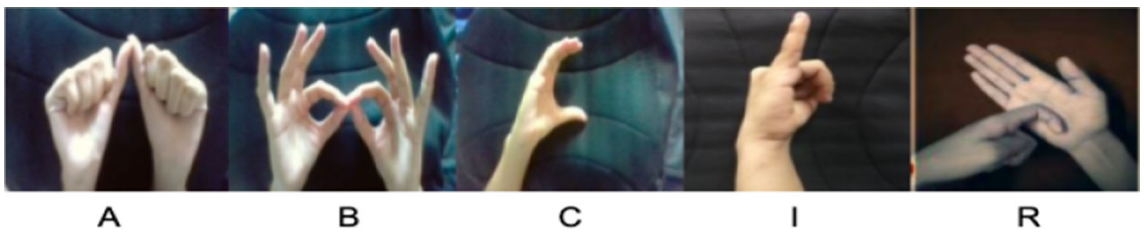


Fig. 2 Sample images from our collected dataset. The gestures represent the alphabets 'A,' 'B,' 'C,' 'I' and 'R' from left to right, respectively

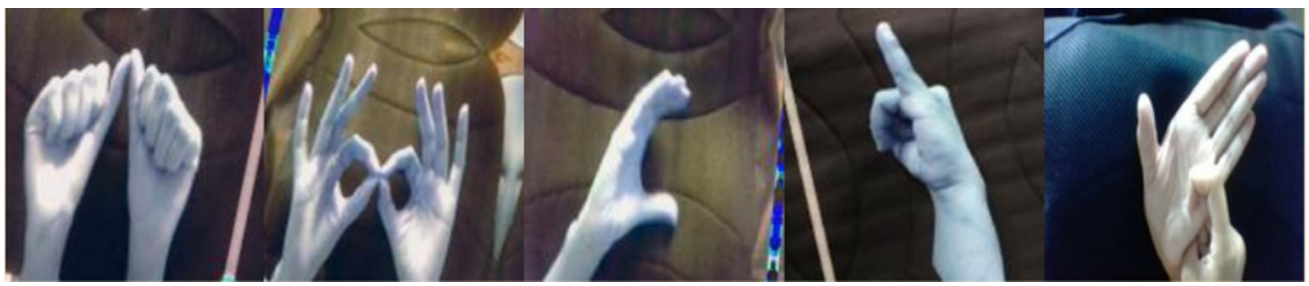


Fig. 3 Images corresponding to the samples from Fig. 2 after augmentation

3.2 System architecture

This study is essentially a comparative analysis between three different architectures, as well as an attempt to judge their strengths and weaknesses for the problem of ISL recognition. These three architectures are explained in detail in this section.

3.2.1 Pre-trained VGG16 model

A pre-trained VGG16 model was employed for training on the dataset of ISL alphabets. The training was done using transfer learning and fine-tuning techniques, thereby establishing a baseline for comparing the models. The VGG network architecture was introduced by Simonyan and Zisserman [22] in their seminal work. This network only uses 3×3 convolutional layers stacked on top of each other in increasing depth. The reduction in the volume size was handled by max-pooling. Two fully connected layers, each with 4096 nodes, are then followed by a softmax classifier. Due to the computational capability and simplicity of the structure, it becomes an excellent choice for any baseline network. Images of size 100×100 were fed to the network due to memory and time restrictions (although the original VGG16 [2] work proposes 224×224 images). The pixels were scaled sample-wise to be between -1 and $+1$ before feeding them to the network. For the transfer learning model, three fully connected layers were stacked over each other. This was followed by a dropout (0.3–0.5) and batch normalization layer for the purpose of regularization.

The entire model took around an hour to train with transfer learning. This process was done by freezing the rest of the VGG16 layers and appending a few trainable layers along with a different softmax layer corresponding to the 26 classes. Around 2.5×10^5 parameters were trainable, whereas the rest were frozen while considering the ImageNet weights. For fine-tuning the model, the last four layers of the VGG16 model were switched to trainable. This was achieved by removing the last softmax layer and appending two fully connected layers. This was followed by a dropout (0.3–0.5) and batch normalization layer with a customized softmax layer having 26 nodes for each alphabet. Out of the total 17.3×10^5 parameters, around 9.7×10^5 parameters were trainable and around 7.6×10^5 were non-trainable. The model took around 2 h to be trained. In both of these models, the Adam optimizer [31] worked well with a learning rate of 0.0005. These specific model parameters were selected because VGG16 was trained on the ImageNet dataset having 1000 classes, whereas the dataset used in this work has 26 hand gestures that are not fairly distinguishable. As such, using a higher learning rate caused the model to skip a few details. Furthermore, Adam optimizer worked well with a low learning rate since it employs momentum and a running exponential decay rate for recent gradients. The associated loss function which we used was categorical cross-entropy [32]. Figure 4 presents the basic VGG-16 architecture which we have used in our work.

3.2.2 Hierarchical network

The hierarchical network comprises of a segment based on SVM [28], which categorizes images as either one-handed or two-handed. SVM is a supervised machine learning algorithm which is built on the idea of a multi-dimensional

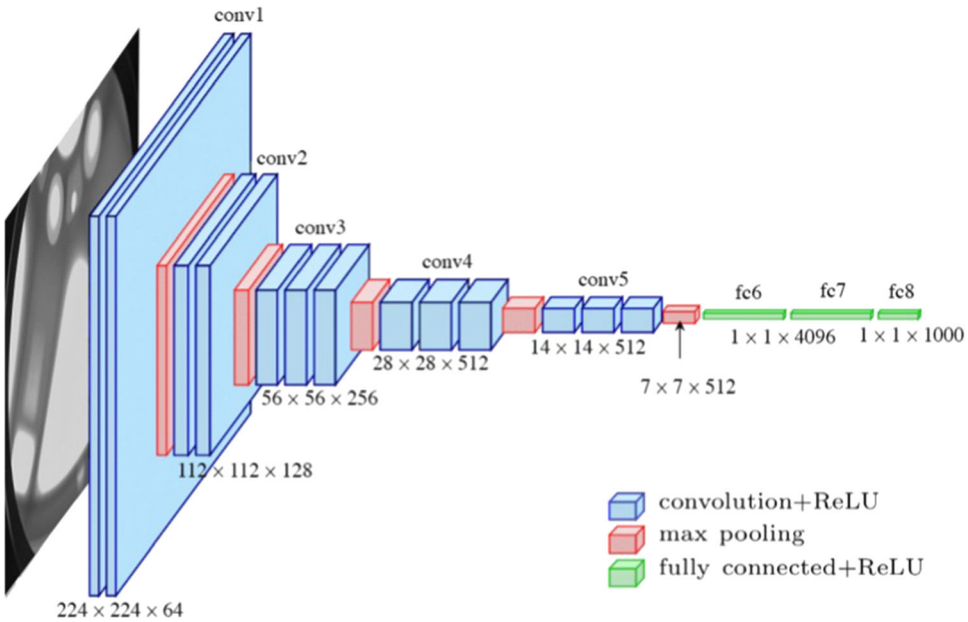


Fig. 4 Standard VGG-16 network architecture as proposed in [22]

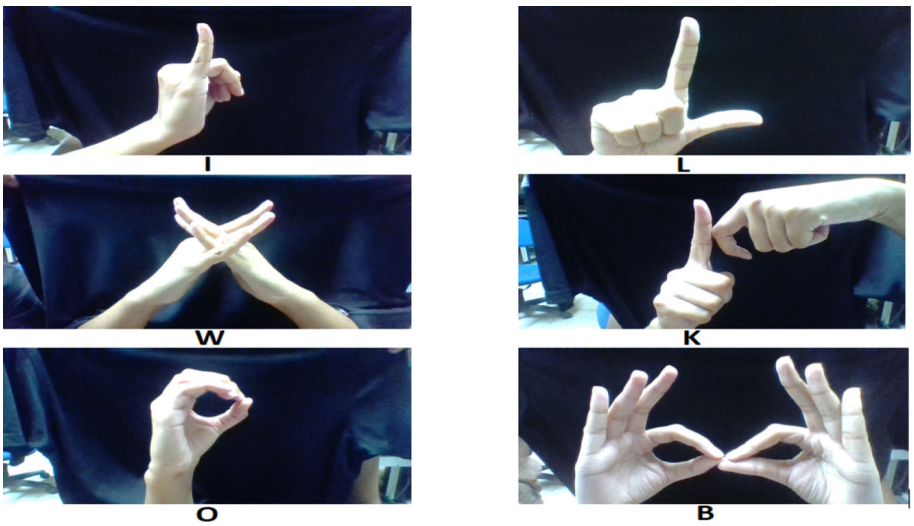


Fig. 5 Sample images in our dataset. 'I,' 'L,' 'O' are one-hand gestures, whereas 'W,' 'K,' 'B' are two-hand gestures. The HOG features of these images are presented in Fig. 6

hyperplane dividing the data into their respective training categories. This step was performed to simplify the task of the network by reducing the number of classes. Around 10,000 images were separated and categorized into two classes—seven one-hand gestures and the remaining 19 two-hand gestures. Each of these gestures comprised of a few hundred images of each alphabet. These images were combined with the Histogram of Oriented Gradients (HOG) [7] to get their best feature representations. The gradients along x and y directions of an image can be vital around edges and corners as it enables us to rectify the regions of abrupt intensity changes. Since the edges and corners pack-in a lot of information about the object (hands in our case), extracting the HOG features before feeding it to the actual model can be quite helpful. A total of 784 pixels per image were taken since SVM requires fewer than the neural networks for making the predictions. Few sample images from our dataset are presented in Fig. 5, and their corresponding HOG features are illustrated in Fig. 6. It is noticeable that the histogram captures the gradients which are present in the hand gestures. In Fig. 6, the bright intensity of color corresponds to a higher gradient value, whereas darker pixels refer to lower gradient and sharp edges (i.e., there exist sharp outlines between two hands or between the hand and background). In this manner, the gradient images removed a lot of non-essential information by highlighting the regions where color intensity changes.

After extracting the HOG features (which returned a 24339×2028 matrix), standardization was performed by scaling the values. The high number of features seemed too overwhelming for the SVM, and hence, Principal Component Analysis (PCA) [21] was performed to extract the most essential 1500 features. The dataset was then split by an 80–20 ratio as the training and testing sets. Finally, an SVM model was fit on the training data by applying a linear kernel. The output of the SVM depicted two classes, '0' for one-hand and '1' for two-hand gestures.

Following the hierarchical approach, two deep Convolutional Neural Networks (CNN) were built. The first model was used to categorize the output of the SVM further into alphabets belonging to each class. The neural networks were based on the VGG16 [22] architecture since it has been proven to be effective in the field of object recognition. The neural networks comprised of convolutional filters followed by max-pooling layers, the fully connected layers and an output softmax layer. The hidden layers used the ReLU activation function. The Adam (Adaptive moment estimation) optimizer with a learning rate of 0.0005 and the categorical cross-entropy loss function were used in this model. One of the neural nets was fed seven classes in the output softmax layer for one-hand gestures, and the other one has 19 classes for two-hand gestures. The segregation of these gestures into two different classes consequently improves the model's ability to learn their differences. The SVM took 13 minutes to train on an Acer-Predator Helios 300 with NVIDIA GeForce-GTX 1050Ti with 8 GB RAM. The neural networks took around 2 h to train on our dataset.

3.2.3 Natural language-based network

The natural language-based output network was developed for rectifying errors made by the CNN model. The main motive of this model is to correct the falsely predicted outcomes during ISL-conversation. Thus, a misspelled word can be corrected by using an algorithm that considers some specific words in the English language. These words are formed by the predicted alphabets via intelligently changing a letter or two. Such algorithms are useful in practical terms to overcome the flaws of CNN. A 13-layer

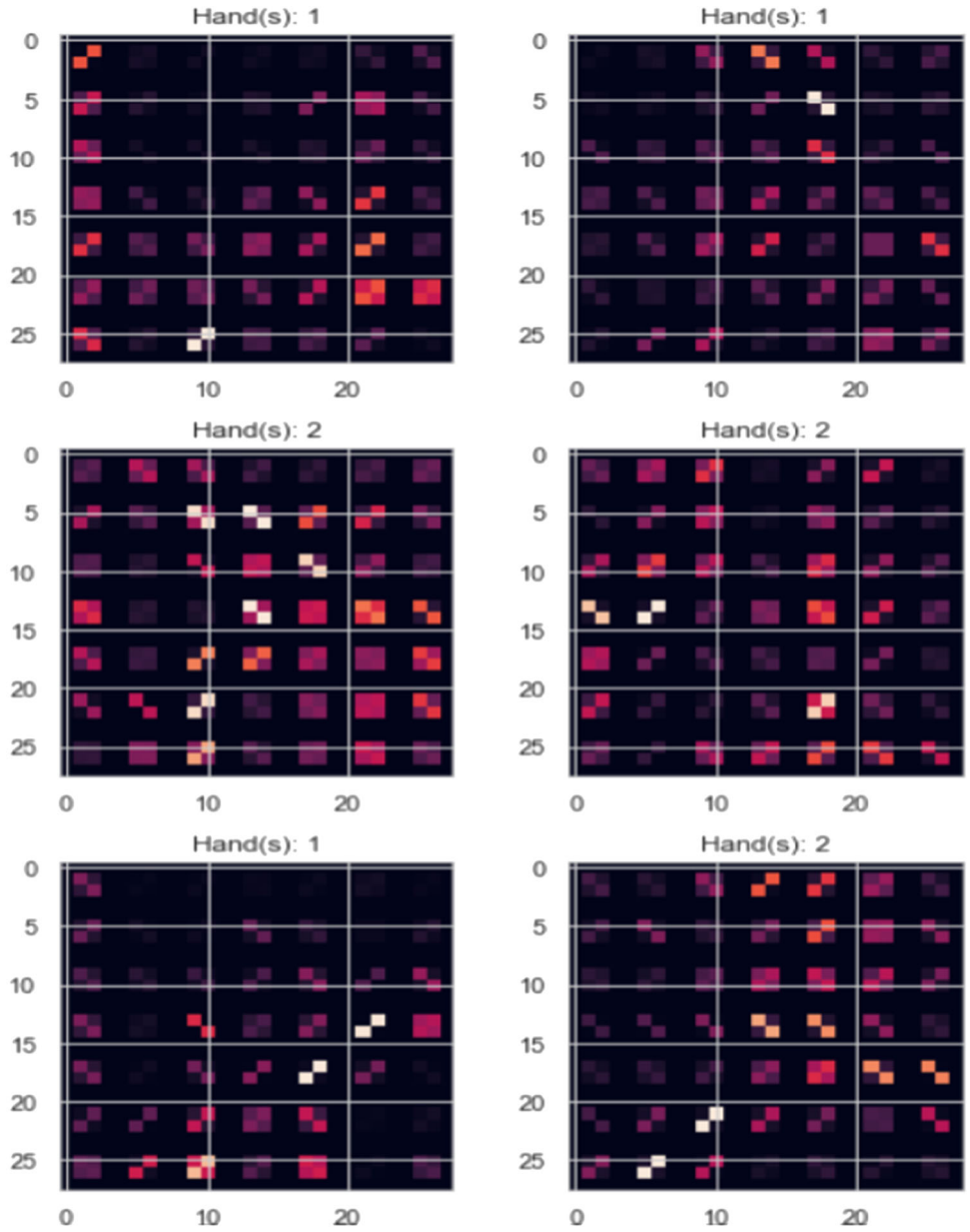


Fig. 6 HOG features corresponding to the sample images presented in Fig. 5

CNN was developed which received these images, with their pixels scaled between -1 and $+1$. The neural net was a simple network comprising of 3×3 convolutional filters followed by max-pooling. This network architecture was inspired by VGG16 since it works well for image classification problems. The latter layers consisted of dropout (0.3–0.4) and batch normalization for avoiding any chances of overfitting. The Adam optimizer with a learning rate of 0.0002 was used to minimize the categorical cross-entropy loss function. The learning rate of 0.0002 was chosen since it was observed that a few gestures were relatively similar and required more focus for differentiating. Thus, a lower learning rate with more number of iterations helped the network to converge in a better manner. The softmax layer provided output as 26 probabilities, each corresponding to the particular alphabet. Exploiting this characteristic of the softmax layer, we calculated the total probability for a given word. This probability is estimated as the sum of all the highest predicted output probabilities for that alphabet.

Let us suppose the word that a user inputs is 'cat.' For each letter 'c,' 'a' and 't,' the probabilities for the top three predicted letters would get saved. This quantity will be the overall probability of the word being 'cat.' Now, if the output probabilities that the CNN provided with respect to each letter corresponded to 'cet,' then this word will be searched through a corpus of length $= 3$ in the English dictionary. If no such word exists, it will change the letters

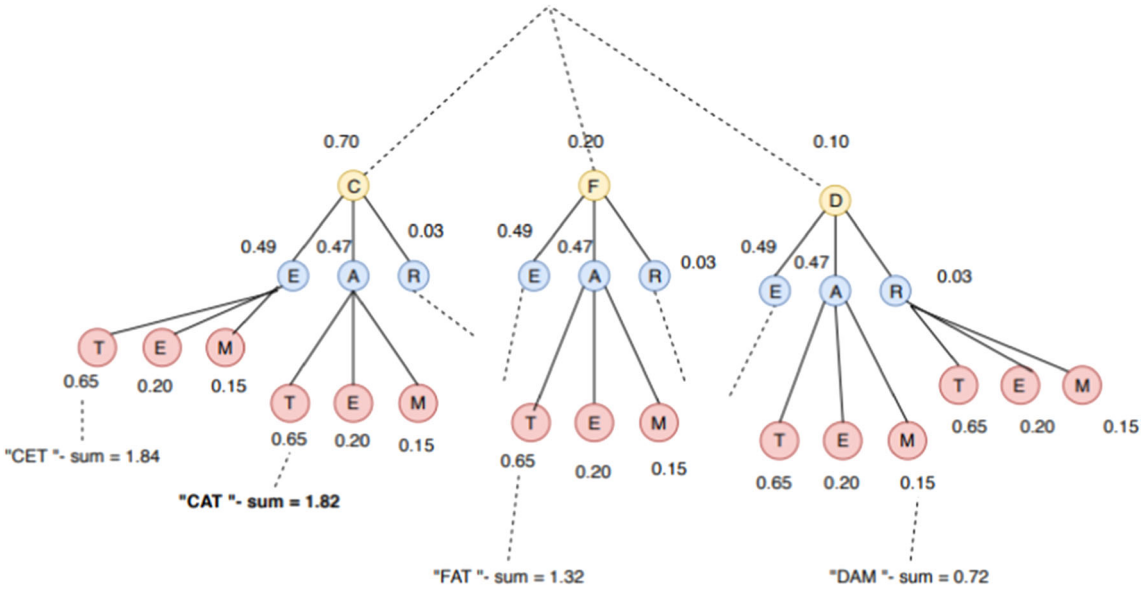


Fig. 7 Formation of all possible words and the respective sum of probabilities

(one at a time) by the next highest probable letter and recheck it in the dictionary. If such a word exists, it gets stored along with its total probability. The model outputs the word with the highest probability belonging in the English dictionary as the final prediction. This framework works on the idea that if a user wants to converse in finger-spelled ISL, he/she is likely to depict a word that exists in the English dictionary (apart from unusual proper nouns). This model is practically demonstrated in Sect. 5 and diagrammatically presented in Fig. 7.

4 Experimental results

All the three networks were tested on our dataset of around 149,568 images. The testing dataset consists of hand images clicked on a black background. The images were augmented, resized and pre-processed as per the network requirements. The results for all three models are presented and analyzed as follows.

4.1 VGG16 transfer learning and fine tuning

The pre-trained VGG16 model with transfer learning produced an accuracy of 53% and the fine-tuned model resulted in an accuracy of 94.52%. The reason for this low accuracy in VGG16 can be attributed to the fact that it was initially trained for 1000 categories, which were fairly distinguishable from one another. However, the present model works on 26 hand gestures, and out of them, quite a few are very similar to one another. Furthermore, the presence of one- and two-hand gestures causes feature occlusion. The final results were accumulated after running the model for 100 epochs with early stopping based on the validation error.

4.2 Hierarchical model

The hierarchical model resulted in a training loss of 0.0016, thus translating to a training accuracy of 99% and a validation accuracy of 98.52% for categorizing one-hand features. For two-hand features, the training and validation accuracies were 99 and 97%, respectively. The SVM used in the hierarchical model for categorizing gestures into one-hand and two-hand gestures, combined with the HOG features, produced an accuracy of 96.79%. Importantly, this recognition performance is significantly better than any other machine learning algorithm for 26 classes. The confusion matrix in Table 1 segregates the results obtained on the 6085 test samples into true positives, true negatives, false positives and false negatives. The standard performance parameters were calculated as precision = 0.97, recall = 0.96 and F1-score = 0.96.

4.3 Natural language-based model

The natural language-based model produced an accuracy of 92% when applied as a follow-up to a convolutional neural network. This model helps in cases with a few wrongly predicted alphabets due to ambiguities caused by human error or background noises. To summarize, the results produced by the hierarchical model are significantly better than any other machine learning algorithm for the 26 classes. The individual recognition accuracy for some

Table 1 Confusion matrix of the SVM used in the hierarchical model

$n = 6085$	Positive	Negative
True	2930	2960
False	90	105

Table 2 Comparison of the accuracy (%) of various models for a few ISL gestures

Alphabet	VGG 16		Hierarchical
	Transfer learning	Fine-tuning	
A	0.53	0.73	0.71
C	0.69	0.87	0.78
F	0.47	0.67	0.83
H	0.46	0.59	0.71
I	0.81	0.92	0.93
L	0.79	0.94	0.91
M	0.46	0.71	0.88
O	0.74	0.85	0.87
V	0.72	0.81	0.93
X	0.48	0.71	0.77

sample ISL gestures is presented in Table 2. All of the results are summarized in Table 3.

4.4 Comparative analysis

In this final empirical section, we compare our results with some state-of-the-art-related works. As shown in Table 4, many of the pre-existing works refer to gesture recognition with an incomplete set of alphabets. The average accuracy of our proposed model is at par with the presented works. The model introduced in [18] provides an accuracy of 97.5% for just four letters, while our model provides an average accuracy of approximately 97.76% for all 26 characters irrespective of the background noise. The proposed framework uses a hierarchical model of neural networks, which is fairly novel compared to the pre-existing works. The accuracy for ISL recognition was noted to be comparatively lower than other languages due to the involvement of both one- and two-hand gestures. The model proposed in our work encompasses the complete English alphabets, which can consequently lead to practical implementation for ISL recognition. The proposed method works well in noisy backgrounds and different lighting conditions. Furthermore, it is free from feature occlusion due to the presence of both one-hand and two-hand features. Since it employs a hierarchical approach to address this problem, the treatment of one-hand and two-hand features by separate neural networks enhances the model accuracy. As an additional advantage, our model does not require any hardware support from the user's end for gesture detection and recognition.

5 Interface development

The trained model was tested against all the alphabets by taking snaps of gestures through a laptop web camera in different lighting conditions. A web interface has been developed in Django (which is a macro-web-framework of Python) through which the hand gestures of the users are captured. The user is required to form hand gestures keeping the hands within the boundaries of a square being displayed on the camera window. This process is pictorially presented in summarized Fig. 8. The web camera snaps the images automatically at an interval of 5 s. Once done, the gesture snaps are fed to our deep learning model which produces the word for which the gestures were given. If no such valid word exists in the English corpora, the model attempts to generate the most probable word that can be formed. This generated word appears on the screen as text in an HTML Modal. Figure 9a shows the sequence of gestures involving one-hand which is fed to the hierarchical model, and Fig. 9b illustrates the translated text from the input images. Figure 10a shows the sequence of one- and two-hand gestures which are fed to the hierarchical model, and Figure 10b presents the corresponding translated text. The entire code of this project is made available at https://github.com/yatharth77/Indian-Sign-Language-Gesture-Recognition.

Table 3 Summary of the results obtained through the three proposed models

Metric	VGG 16		Hierarchical	Natural language based
	Transfer learning	Fine-tuning		
# of trainable parameters	1024	9,704,962	One-hand: 9,707,527; Two-hand: 9,713,683	9,704,962
# of non-trainable parameters	17,088,858	7,637,312	One-hand: 7,637,312; Two-hand: 7,637,312	7,637,312
Training accuracy	84%	92%	One-hand: 99%; Two-hand: 99%	92%
Validation accuracy	53%	90%	One-hand: 98.52% Two-hand: 97%	–

Table 4 Comparative analysis of the proposed ISL recognition techniques with other related studies

References	Language	# of characters	Primary components	Average accuracy
[4]	American	35	LBP[a], 3D Voxel, CNN, SVM[b]	92%
[16]	American	26	Kinect, SVM	100% for 2 char 90% for 'S' 80% for 'I' 77% for 'F,' 'T'
[18]	Indian	4	SVM, PCA[c]	97.5%
[3]	Indian	140	Kinect, VFH [d], NN[e]	90.68%
[14]	Indian	26	SIFT[f]	98%
[9]	Indian	22	PSO[g]	99.96%
[1]	Indian	36	HOG[h], SURF[i], MSVM[j]	93%
[17]	Indian	140	Kinect, SIFT, HOG, LBP, SVM	71.85%
Proposed	Indian	26	SVM, VGG16	98.52% (One-Hand) 97% (Two-Hand)

[a]Local Binary Pattern
[b]Support Vector Machine
[c]Principal Component Analysis
[d]Viewpoint Feature Histogram
[e]Nearest Neighbor
[f]Scale-Invariant Feature Transform
[g]Particle Swarm Optimization
[h]Histogram of Oriented Gradients
[i]Speeded Up Robust Features
[j]Multi-class SVM

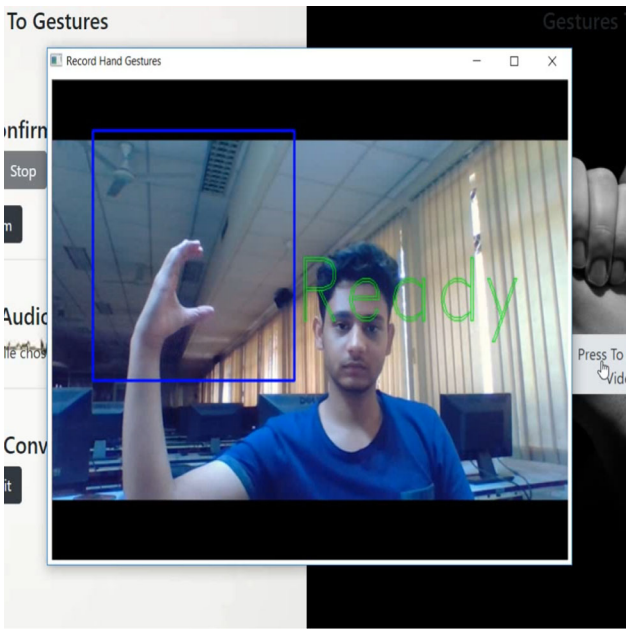
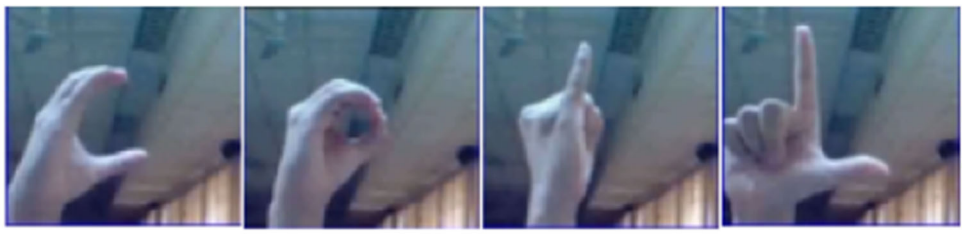


Fig. 8 Capture of the image frames in our developed interface for ISL conversion

6 Conclusion and future works

In this study, we have attempted to perform an analytic comparison among three promising deep learning-based approaches (the pre-trained VGG16 model, natural language-based output network and the hierarchical network) for identifying ISL gestures. The hierarchical model comprehensively surpasses the other two models with an accuracy of 98.52% for one-hand and 97% for two-hand gestures. The proposed model eliminates the need for any hardware support, thereby making it practical and cost-effective. Another feature of our model is its ability in handling feature occlusion. Occlusions are caused due to the use of a single hand in some ISL gestures and both hands in other ISL gestures. Our model deals with this problem in the first stage by recognizing the number of hands in the input gesture. In contrast to the previous works involving gesture recognition, our work also gives a prediction of the word formed when a sequence of gestures are given as input.

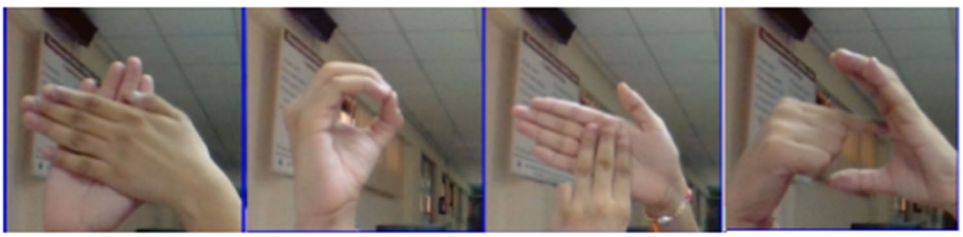
The proposed model faces a few problems, the first one being the issue in categorizing the alphabet 'J.' Unlike others, 'J' is a dynamic gesture that involves the movement of hands as well. The models mentioned in Sect. 3 only deal with images and need to be modified to capture



(a) ISL gestures using one hand of the word 'COIL'

(b) ISL gestures translation of the word 'COIL'

Fig. 9 ISL gesture and corresponding text of the word 'COIL'



(a) ISL gestures combination of both hands of the word 'HOME'

(b) ISL gesture translation of the word 'HOME'

Fig. 10 ISL gesture and corresponding text of the word 'HOME'

frames. The VGG16 model with transfer learning faces the issue of feature similarity since the classes are quite similar to each other (unlike the data on which it was originally trained). The VGG16 model with fine-tuning mixes up some similar looking characters like 'm' and 'n,' 'x' and 'f.' Although the hierarchical model works better in comparison with the other two, the classification phase could be further improved to learn the essential features. Some lighting changes also cause mild effects on the results, which can be overcome by adding more images in the dataset under varying illumination and backgrounds. Currently, the algorithm generates the nearest possible word from a sequence of input gestures. However, it must be capable of generating sentences for better practical usages. The optimization process in the algorithm can be also studied for generation faster outputs from longer sequences of input gestures. All of these problems can be addressed in the future as a refinement of this work.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Agrawal SC, Jalal AS, Bhatnagar C (2012) Recognition of Indian Sign Language using feature fusion. In: 2012 4th international conference on intelligent human computer interaction (IHCI), pp 1–5
2. Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: 2017 International conference on engineering and technology (ICET), pp 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186
3. Ansari ZA, Harit G (2016) Nearest neighbour classification of Indian Sign Language gestures using kinect camera. Sadhana 41(2):161–182. https://doi.org/10.1007/s12046-015-0405-3
4. Beena V, Namboodiri A, Thottungal R (2019) Hybrid approaches of convolutional network and support vector machine for American sign language prediction. Multimed Tools Appl. https://doi.org/10.1007/s11042-019-7723-0
5. Boulares M, Jemni M (2016) Learning sign language machine translation based on elastic net regularization and latent semantic analysis. Artif Intell Rev 46(2):145–166. https://doi.org/10.1007/s10462-016-9460-3
6. Cheok MJ, Omar Z, Jaward MH (2019) A review of hand gesture and sign language recognition techniques. Int J Mach Learn Cybernet 10(1):131–153. https://doi.org/10.1007/s13042-017-0705-5
7. Feng K, Yuan F (2013) Static hand gesture recognition based on hog characters and support vector machines. In: 2013 2nd international symposium on instrumentation and measurement, sensor network and automation (IMSNA), pp 936–938. https://doi.org/10.1109/IMSNA.2013.6743432
8. Hietanen JK, Leppänen JM, Lehtonen U (2004) Perception of emotions in the hand movement quality of Finnish sign language. J Nonverbal Behav 28(1):53–64. https://doi.org/10.1023/B:JONB.0000017867.70191.68
9. Hore S, Chatterjee S, Santhi V, Dey N, Ashour AS, Balas V, Fuqian S (2017) Indian Sign Language recognition using optimized neural networks, vol 455, pp 553–563. https://doi.org/10.1007/978-3-319-38771-0_54
10. Kong WW, Ranganath S (2010) Sign language phoneme transcription with rule-based hand trajectory segmentation. J Signal Process Syst 59(2):211–222. https://doi.org/10.1007/s11265-008-0292-5
11. Krak IV, Barmak OV, Romanyshyn SO (2014) The method of generalized grammar structures for text to gestures computer-aided translation. Cybernet Syst Anal 50(1):116–123. https://doi.org/10.1007/s10559-014-9598-4
12. Luqman H, Mahmoud SA (2018) Automatic translation of Arabic text-to-Arabic sign language. Universal Access in the Information Society. https://doi.org/10.1007/s10209-018-0622-8
13. Morrissey S, Way A (2013) Manual labour: tackling machine translation for sign languages. Mach Transl 27(1):25–64. https://doi.org/10.1007/s10590-012-9133-1
14. Patil SB, Sinha GR (2017) Distinctive feature extraction for Indian Sign Language (ISL) gesture using scale invariant feature transform (SIFT). J Inst Eng (India) Ser B 98(1):19–26. https://doi.org/10.1007/s40031-016-0250-8
15. Pigou L, Dieleman S, Kindermans PJ, Schrauwen B (2015) Sign language recognition using convolutional neural networks. In: Agapito L, Bronstein MM, Rother C (eds) Computer vision—ECCV 2014 workshops. Springer, Cham, pp 572–578
16. Quesada L, López G, Guerrero L (2017) Automatic recognition of the American sign language fingerspelling alphabet to assist people living with speech or hearing impairments. J Ambient Intell Humaniz Comput 8(4):625–635. https://doi.org/10.1007/s12652-017-0475-7
17. Raghuveera T, Deepthi R, Mangalashri R, Akshaya R (2020) A depth-based Indian Sign Language recognition using Microsoft Kinect. Sadhana 45(1):34. https://doi.org/10.1007/s12046-019-1250-6
18. Raheja JL, Mishra A, Chaudhary A (2016) Indian Sign Language recognition using SVM. Pattern Recogn Image Anal 26(2):434–441. https://doi.org/10.1134/S1054661816020164
19. Rajam PS, Balakrishnan G (2011) Real time Indian Sign Language recognition system to aid deaf-dumb people. In: 2011 IEEE 13th international conference on communication technology, pp 737–742
20. Rogers KD, Pilling M, Davies L, Belk R, Nassimi-Green C, Young A (2016) Translation, validity and reliability of the British Sign Language (BSL) version of the EQ-5D-5L. Qual Life Res 25(7):1825–1834. https://doi.org/10.1007/s11136-016-1235-4
21. Sehgal S, Singh H, Agarwal M, Bhasker V, Shantanu (2014) Data analysis using principal component analysis. In: 2014 international conference on medical imaging, m-health and emerging communication systems (MedCom), pp 45–48. https://doi.org/10.1109/MedCom.2014.7005973
22. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: 3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference track proceedings. arxiv:1409.1556
23. Sinha GR (2017) Indian Sign Language (ISL) biometrics for hearing and speech impaired persons: review and recommendation. Int J Inf Technol 9(4):425–430. https://doi.org/10.1007/s41870-017-0049-0
24. Stein D, Schmidt C, Ney H (2012) Analysis, preparation, and optimization of statistical sign language machine translation. Mach Transl 26(4):325–357. https://doi.org/10.1007/s10590-012-9125-1
25. Tolba M, Samir A, Aboul-Ela M (2012) Arabic sign language continuous sentences recognition using PCNN and graph matching. Neural Comput Appl 23:999–1010. https://doi.org/10.1007/s00521-012-1024-0
26. Wadhawan A, Kumar P (2020) Deep learning-based sign language recognition system for static signs. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04691-y
27. Xu L, Gao W (2000) Study on translating Chinese into Chinese sign language. J Comput Sci Technol 15(5):485–490. https://doi.org/10.1007/BF02950413
28. Yang Y, Li J, Yang Y (2015) The research of the fast SVM classifier method. In: 2015 12th international computer conference on wavelet active media technology and information processing (ICCWAMTIP), pp 121–124. https://doi.org/10.1109/ICCWAMTIP.2015.7493959
29. Zafrulla Z, Brashear H, Starner T, Hamilton H, Presti P (2011) American sign language recognition with the kinect. In: Proceedings of the 13th international conference on multimodal interfaces, association for computing machinery, New York, NY, USA, ICMI'11, pp 279–286. https://doi.org/10.1145/2070481.2070532
30. Zeshan U (2003) Indo-pakistani sign language grammar: a typological outline. Sign Lang Stud 3:157–212. https://doi.org/10.1353/sls.2003.0005
31. Zhang Z (2018) Improved ADAM optimizer for deep neural networks. In: 2018 IEEE/ACM 26th international symposium on quality of service (IWQoS), pp 1–2. https://doi.org/10.1109/IWQoS.2018.8624183
32. Zhang Z, Sabuncu MR (2018) Generalized cross entropy loss for training deep neural networks with noisy labels. CoRR arxiv:1805.07836