EMITTER International Journal of Engineering Technology
Vol. 9, No. 1, June 2021, pp. 182~203
DOI : 10.24003/emitter.v9i1.613

Indian Sign Language Recognition through Hybrid ConvNet-LSTM Networks

Muthu Mariappan H, Gomathi V*

Computer Science and Engineering Department, National Engineering College, Kovilpatti, Tamil Nadu, India.
Corresponding Author: vgcse@nec.edu.in

Received April 12, 2021; Revised May 13, 2021; Accepted June 15, 2021

Abstract

Dynamic hand gesture recognition is a challenging task of Human-Computer Interaction (HCI) and Computer Vision. The potential application areas of gesture recognition include sign language translation, video gaming, video surveillance, robotics, and gesture-controlled home appliances. In the proposed research, gesture recognition is applied to recognize sign language words from real-time videos. Classifying the actions from video sequences requires both spatial and temporal features. The proposed system handles the former by the Convolutional Neural Network (CNN), which is the core of several computer vision solutions and the latter by the Recurrent Neural Network (RNN), which is more efficient in handling the sequences of movements. Thus, the real-time Indian sign language (ISL) recognition system is developed using the hybrid CNN-RNN architecture. The system is trained with the proposed CasTalk-ISL dataset. The ultimate purpose of the presented research is to deploy a real-time sign language translator to break the hurdles present in the communication between hearing-impaired people and normal people. The developed system achieves 95.99% top-1 accuracy and 99.46% top-3 accuracy on the test dataset. The obtained results outperform the existing approaches using various deep models on different datasets.

Keywords: Sign Language Recognition, CasTalk-ISL Dataset, Convolutional Neural Network, Recurrent Neural Network.

1. INTRODUCTION

"Sign language can be considered as a collection of gestures, movements, postures, and facial expressions corresponding to letters and words in natural languages" [1]. Sign languages are the prime mode of communication for hearing and speech impaired people. Sign languages have several variants like American Sign Language (ASL), British Sign Language (BSL), Chinese Sign Language (CSL), and Indian Sign Language (ISL) [2,3,4], based on the originated countries. The communication between the hearing impaired and ordinary people are always idle since the latter do not know these signs [5]. Fingerspelling, writing text messages, and using visual media are some

alternatives to sign language. However, they are pretty much lack of expressing feelings and emotions. To overcome the inefficiency in those communication methods, the deaf-mute people prefer only sign languages for expressing their moods and thoughts to others.

Conversing in sign language brings more comfort for the hearing-impaired community. So, there is a need for real-time sign language interpreters when they communicate with ordinary people. Hence, intelligent machine-vision systems are preferred. Developing a real-time sign language interpreter majorly relies on the hand gesture recognition task, which is implemented as a hand-gloves based approach and vision-based pattern recognition approach [4, 6, 7]. In Glove-based gesture recognition, the users are supposed to wear the data gloves carrying sensors and cables. Even though the recognition rate is higher in the sensor-based approach, it is significantly dominated by its hardware requirements and the interaction with the user [6, 8]. Vision-based systems overcome these issues, and they can handle texture and colour properties too. This section concludes that the non-invasive real-time video capturing vision-based solutions are most suitable for the casual talk environment.

In recent days, vision-based sign language recognition is addressed by researchers in three levels [6]. Alphabets and numbers (0-9) recognition are carried out at the primary level, and in the moderate level, symbolic words are recognized, whereas, in the advanced level, sentence recognition is carried out. Most of the existing systems are at the primary level only [9]. Currently, moderate level research is going to the peak, and the advanced level has several challenges to be addressed by the researchers. The alphabets and numbers can be recognized by static hand gesture recognition systems with images, whereas the words and sentences can only be recognized by dynamic hand gesture recognition systems from real-time videos. Most of the machine learning algorithms suffer when a large amount of data is needed to be processed [7]. Also, feature extraction and feature selection play a vital role in traditional algorithms since they are associated with the system's performance. Hence, neural networks are preferred to avoid the manual feature extraction process. The proposed system is developed using the highly efficient hybrid CNN-RNN architecture [4, 10, 11].

The proposed research was experimented with to recognize signs of the CasTalk-ISL dataset with the divine goal of developing a real-time sign language recognition system. The research team has conducted few data collection camps to create the CasTalk-ISL dataset with the support of trained student volunteers. The volunteers have been trained with the signs of the ISL dictionary launched by the Indian Sign Language Research and Training Center [www.islrtc.nic.in], Department of Empowerment of Persons with Disabilities, Ministry of Social Justice & Empowerment, Government of India. This dictionary was released in DVD form with 3,000 specialized terms from legal, academic, medical, and technical fields. The rest of the sections are organized as follows; section 2 portrays the literature survey with state-of-the-

art methodologies and the current trend in the proposed research area. Section 3 depicts the proposed gesture recognition system in detail. The implementation results are explained and analyzed in section 4 and section 5. Section 6 concluded the proposed research and gives scope for further improvements.

2. RELATED WORKS

2.1 Hand Gesture Datasets

2.1.1. SLVM [2]

SLVM dataset contains 6,800 manually labelled continuous video sequences belonging to 20 categories of frequently used words in Chinese museums. Each vocabulary has 340 video samples recorded from 17 persons with 20 repetitions of each gesture using Microsoft Kinect.

2.1.2 SLR_dataset [12]

SLR dataset has 25,000 RGB videos with a resolution of 1280 x 720 pixels with a frame rate of 30 fps. 50 signers were involved in recording 250 instances for 100 CSL gestures. The videos are recorded using Microsoft Kinect 2.0 with a distance of 1.5 meters between the signers and the device. The dataset has three modalities viz. RGB videos, depth videos, and skeleton joints.

2.1.3 DHG-14/28 [13]

DHG-14/28 is a Dynamic Hand Gesture dataset with 2,800 video sequences of 14 gestures performed in two modes: using one finger and using the whole hand. Each gesture is performed by 20 participants in two ways with 10 repetitions. The videos are recorded using Intel RealSense short-range depth camera with a resolution of 640 x 480 at 30 fps. The frame length of the video samples varies from 20 to 50 frames.

2.1.4 DEVISIGN-D [14]

DEVISIGN-D dataset is a subset of the DEVISIGN dataset. The subset is composed of 6,000 gesture videos of 500 daily used words of CSL. Totally 8 signers (4 male and 4 female) were involved in the data collection process, where 4 signers have performed each gesture 2 times, and 4 signers have performed each gesture 1 time, resulting in 12 samples for each CSL gesture word.

2.1.5 LSA64 [15]

LSA64 is an Argentinian Sign Language (LSA) database created to develop an automatic LSA sign recognizer. The videos are recorded in a white background, and the signers wore a black dress and fluorescent-coloured gloves. 10 signers were employed to perform 5 repetitions of 64 LSA words resulting in 3,200 video samples. The videos are recorded with a Sony HDR-CX240 camera with a resolution of 1920 x 1080 with 60 frames per second.

2.1.6 ISL dataset [16]

The dataset has 5,041 image samples of 140 static hand signs picked from ISL vocabularies. Each gesture sign has 36 RGB images and 36 depth images. The dataset contains image samples for alphabets, numbers, technical, and common words. The data collection was performed using Microsoft Kinect with the help of 18 signers. Each image of the dataset has a resolution of 640 x 480 pixels.

2.1.7 CLAP 14 [17]

ChaLearn Looking At People 2014 (CLAP 14) Gesture Spotting dataset has video samples for 20 Italian gestures performed by 27 persons with variations in background, clothes, and lighting in the environment. 6,600 samples in the development set and 3,543 samples in the testing set have been captured using Microsoft Kinect. This Gesture spotting dataset is an enhanced version of ChaLearn 2013 [18] Montalbano gesture dataset.

2.1.8 ChaLearn 2013 [18]

ChaLearn 2013 challenge dataset consists of video samples belonging to 20 Italian gestures related to cultural terms. Samples from 27 signers have collected using the Kinect sensor. The database has RGB, depth maps, skeleton models, and audio. The development set has 11,116 samples, and the testing set has 2,742 samples. The development set and the test set were recorded by a different set of signers.

2.2 Sign Language Recognition

Hand gesture recognition is used as the core module in different applications such as sign language recognition, Augmented Reality (AR), Gaming, and Robotics. Gesture recognition can be achieved in two ways: sensor base approach and vision-based approach [4, 6, 19]. The researchers have used many state-of-the-art approaches and driven some new approaches to recognize hand gestures. Sign language recognition is the extended research of hand gesture recognition, which is the active research area of computer vision for the last three decades.

2.2.1 Sensor Based Approach

Glove based recognition is the most successful methodology of the sensor-based approach or hardware approach. The speed of identification of the gesture is significantly better in the glove-based approach than the vision-based systems. Also, data gloves have preferred in many virtual reality (VR) applications [20]. However, in this approach, the signer has to constantly wear special gloves, consisting of several sensors interconnected through fibre optic cables for data transmission [6]. The users lose their comfort; since the gloves should be connected with the processor all the time. Sometimes the gloves may not fit correctly, as different users have different hand sizes and finger thickness [6]. More importantly, facial expressions cannot be recorded using this approach. In addition to data gloves, leap motion controller [10, 22, 23] and accelerometer [21] are also used to capture human motions. Each sensing

technology varies with several parameters like accuracy, user comfort, range of motion, and cost [8]. The experimental setup requires more cost, and also due to its requirements and connectivity with the user, they are significantly dominated by the vision-based systems [6, 9].

2.2.2 Vision Based Approach

In recent years, vision-based gesture recognition became a dominant focus research area of HCI and has proven results in many real-life applications such as Virtual Reality (VR), Augmented Reality (AR), smart vehicle control, intelligent home control and virtual gaming [7]. The arrival of the Microsoft Kinect sensor with the camera or the video recorder to capture the hand gestures has promoted the vision-based approach to the next level [2, 5, 7, 11, 25]. Static hand gestures defined by the pose or the orientation of the body parts are captured as images, whereas dynamic hand gestures defined by the spatial deformation of the body parts are recorded as videos [4, 7, 8].

To model the hand gesture recognition system, the algorithms such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), K-means clustering, Hidden Markov Models (HMM), Conditional Random Field (CRF), Artificial Neural Networks (ANN) and Deep Neural Networks (DNN) were greatly involved in literature [7]. Besides Neural Network, all the other approaches have failed to cross the benchmark. In particular, Convolutional Neural Networks (CNN), a variant of DNN, is a suitable model for recognizing the gestures from images [5]. CNN has several variants like 3D-CNN [25], 3D-ResNet [25], Faster R-CNN [29], VGGNet [28] and Inception ConvNet [29] to perform hand gesture recognition. However, to characterize the temporal relationship of sign languages, few researchers suggested incorporating CNN with Recurrent Neural Networks (RNN) [9]. Even though RNN can handle the temporal information, but their short-term memory makes them inefficient for real-time applications [4]. A special variant of RNN known as Long Short-Term Memory (LSTM) [29, 31, 32] handles a wider range of temporal variations and works well for sign language interpretation at moderate or advanced levels of hand gesture recognition.

3. SYSTEM DESIGN

The primary intention of the proposed research system is to recognize the words of ISL. The system achieves this through dynamic hand gesture recognition from real-time videos [3, 9, 39]. The system is developed using deep neural network models since they have proven results in recent gesture recognition research. So, the proposed system uses hybrid CNN-RNN [30, 32] architecture to extract the features and to recognize the words of ISL from real-time videos.

3.1 Convolutional Neural Network

Convolutional Neural Network (ConvNet or CNN) is one of the robust Deep Neural Networks, which can extract and remember spatial features [4, 7,

40]. ConvNet is predominantly used for image processing applications such as object detection, image classification, and face recognition [40].

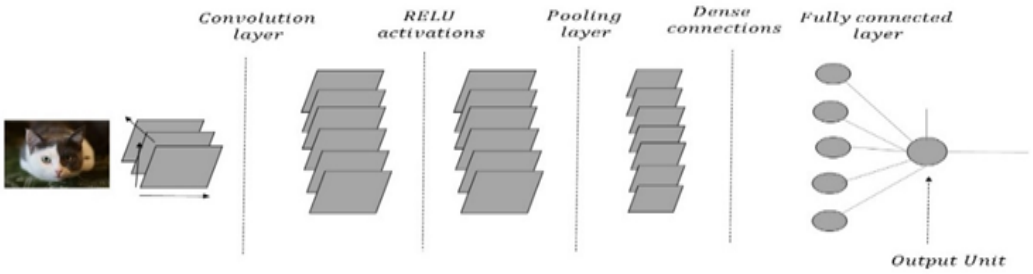

Figure 1. CNN Architecture [37]

The convolution layer produces feature maps for the input images. In the initial convolution layers, the produced feature maps detect simple features such as edges and colour composition variations. In the second convolutional block, the feature map detects complex features such as squares, circles, and other geometrical features. As the network goes deeper, it learns more complicated features like the face, head, hand [37]. The ReLU activations add nonlinearity to the network. The pooling layer [4] summarizes the local neighbourhood information to maintain the translational invariance.

In an ideal CNN, the image passes through the convolution-activation-pooling layers several times before reaching the dense layers. Then the output feature maps are fed into the fully connected layers, followed by the output layer. The output unit varies concerning the problem; if we are doing regression, then the output unit is linear, if it is binary classification, then the output unit is a sigmoid function, and if it is a multiclass classification, then the output unit is softmax layer [37, 39]. The proposed system has effectively used ConvNet to extract the features from every frame of all the videos in the data set. Further, these features are collected as a sequence of data blocks to be fed as the input to the RNN part of the gesture learning model.

3.2 Recurrent Neural Network

Sequence prediction problems are considered complex problems in data science; even Artificial Neural Networks (ANN) [5, 39] cannot guarantee a stable solution. RNNs [9] are remarkable in processing temporal data, where the current input is highly correlated with the input in the previous time step [37]. The chain-like structure of RNN makes them able to solve such sequential problems. The architecture of RNN is illustrated in Fig. 2.

RNN can be defined as multiple copies of the same network, passing the output to a successor. The output of the current node is dependent on the output of its predecessor. RNNs are widely used in applications such as speech recognition, image captioning and various applications of Natural Language Processing (NLP).

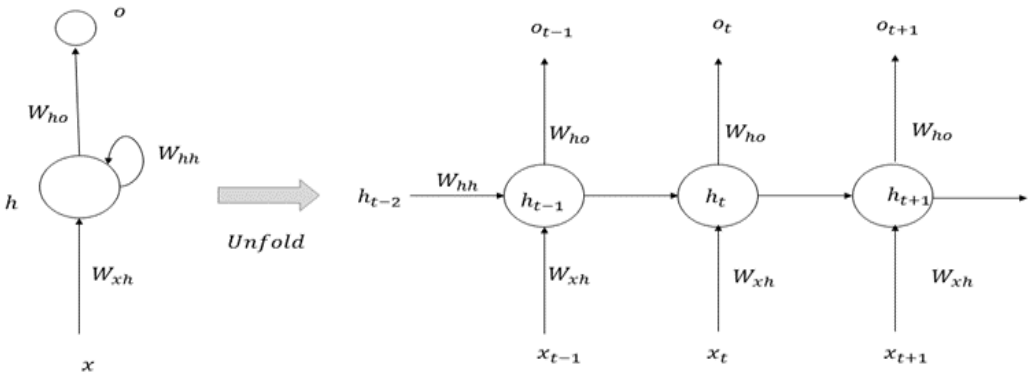

Figure 2. RNN Architecture

The memory state h_t at time step t is calculated using the previous memory state h_{t-1} and the input x_t. The new state h_t is used to predict the output o_t at time step t. The equations of h_t and o_t are derived, as stated below.

$$h_t = f_1\left(W_{hh}\, h_{t-1} + W_{xh}\, x_t + b^{(1)}\right) \quad (1)$$

$$o_t = f_2\left(W_{ho}\, h_t + b^{(2)}\right) \quad (2)$$

For predicting the gesture in a video sequence, the function f_2 can be a softmax function over the ISL words in the vocabulary of ISL gestures and f_1 can be any activation function based on the complexity level that the problem demands.

In RNN, an output error in step t tries to correct the prediction in the previous time steps, generalized by k ∈ 1, 2… t-1 by propagating the error in the previous time steps. This helps the RNN to learn about long dependencies between words that are far apart from each other. In some cases, both feed-forward and back-propagation RNNs are not able to take care of long-term dependencies due to their vanishing gradient problem [37]. The gradient of RNN is derived as,

$$\frac{\partial h_t^{(i)}}{\partial h_k^{(i)}} = \left(u_{ii}\right)^{t-k} \prod_{k=1}^{t-1} \frac{\partial f_2\left(s_{k+1}^{(i)}\right)}{\partial s_{k+1}^{(i)}} \quad (3)$$

In the above equation, the function f2 is generally sigmoid or tanh, which suffers from the saturation problem of having low gradients beyond a specified range of values for the input. Now, since the f2 derivatives are multiplied with each other, the gradient becomes zero if the input is operating in the saturation zone [37]. Thus, the RNN is suffering from the vanishing gradient problem and failed to produce the expected performances sometimes. To solve this vanishing gradient problem, the Long Short-Term Memory (LSTM) networks [7, 11, 37, 40], a special kind of RNN capable of learning long term dependencies are introduced.

3.2.1 Long Short-Term Memory (LSTM) Networks

LSTM has cell state C_t, in addition to the memory state $h_t$of the regular RNNs. The cell state is regulated by three gates: the forget gate f_t, the update gate i_t, and the output gate o_t [11, 34, 42].

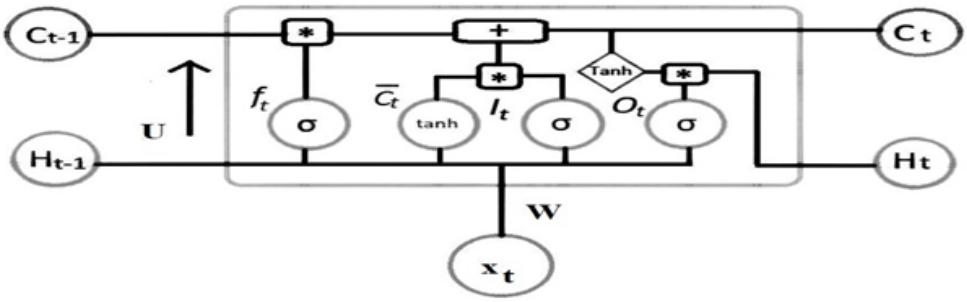

Figure 3. Architecture of an LSTM cell at time t

The forget gate passes the information of the current state and the information from the previous state through the sigmoid function. The information is discarded if the sigmoid value is closer to 0 or kept if the value is closer to 1.

$$f_t = \sigma\left(U_f\, h_{t-1} + W_f\, x_t \right) \quad (4)$$

The input gate also does the same process as of the forget gate.

$$i_t = \sigma\left(U_i\, h_{t-1} + W_i\, x_t \right) \quad (5)$$

To regulate the network, we find the new candidate cell by passing the information of the current state and the previous hidden state through the tanh function. This is expressed as,

$$\tilde{C}_t = \tanh\left(U_c\, h_{t-1} + W_c\, x_t \right) \quad (6)$$

Now, there exists enough information to calculate the value of the cell state. This is calculated by adding the result of the point wise multiplication of the previous cell state and the forget vector and the result of the point wise addition of the input gate value with the potential cell state value.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (7)$$

The output gate decides what the next hidden state should be. At first, it passes the current input and the previous hidden state to calculate o_t. Then it multiplies the tanh output of the new cell state with the sigmoid output $o_t$to decide the information of the hidden state. This is expressed in the following expressions.

$$o_t = \sigma\left(U_o\, h_{t-1} + W_o\, x_t \right) \quad (8)$$

$$h_t = o_t \times \tanh\left(C_t\right) \quad (9)$$

LSTM doesn't suffer from the vanishing gradient problem, and it is proven from the gradient expression.

$$\frac{\partial C_t^{(i)}}{\partial C_k^{(i)}} = \prod_{g=k+1}^{t} f_g^{(i)} \quad (10)$$

From the final expression, it is proven that if the forget cell state is kept closer to 1, the gradients will not get reduced or weakened, and the vanishing gradient problem will not occur [37]. This makes the LSTM more efficient for handling problems, which uses sequences of features to make predictions. With the help of this tremendous remembering power, LSTMs works well in solving the problems of NLP.

To provide a deeper understanding about CNN and the vanishing gradient problem of RNN, the mathematical notations and diagrams are referred from the book, Intelligent Projects using Python [37].

3.3 Proposed Sign Language Recognition System

The flow diagram of the proposed system is depicted in Fig. 4. The gesture videos of the ISL signs are given as input to the system. To process the videos and extract the spatial features, they are converted into a sequence of image frames and passed as the input to the Inception V3-CNN. Feature selection and feature extraction are very much essential processes since they have a direct impact on the recognition rate. This is efficiently completed by the automatic feature extraction abilities of the CNN. The features extracted from each frame are grouped to form the feature sequence of the particular video sample. Then, the generated feature sequences are fed as the input to the LSTM-RNN network, which can learn and remember the sequences for a long time. LSTM network has longer–range memory power, as they do not suffer from vanishing gradient problem. The final layer of the LSTM network returns the probabilities of all classes in the dataset. The class with the highest probability score is declared as the gesture label.

Holistically, the proposed system adapts the transfer learning approach, which is an efficient deep learning technique where the pre-trained model of one application is reused for a new application. It is so common to use transfer learning for predictive modelling problems that deal with images. These pre-trained models have developed for large and challenging image classification competitions, such as the ImageNet 1,000 class classification problem. The best models like the Oxford VGG model, Google Inception model, and Microsoft ResNet model are released under a permissive license for reuse. The proposed system retrains the Inception V3 model for feature extraction from the sign videos of CasTalk-ISL dataset.

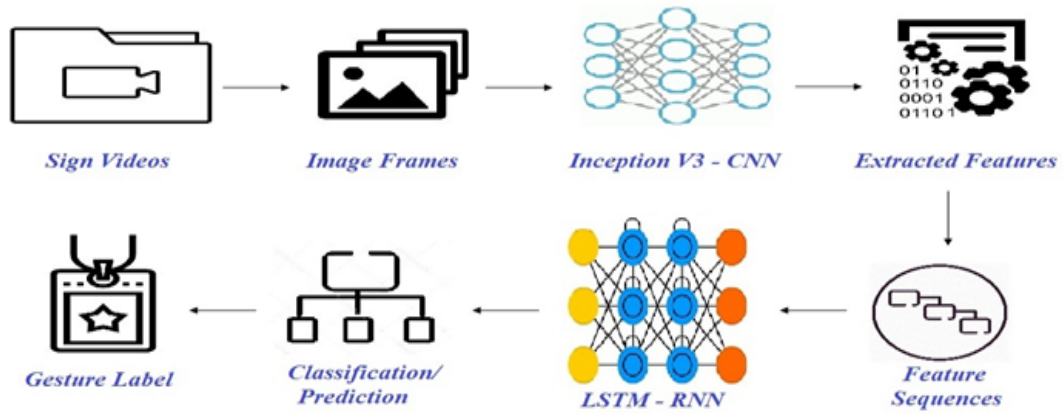

Figure 4. Flow Diagram of the proposed Sign Language Recognition System

The Inception V3-CNN network has a sequence of inception modules comprising the three variants A, B, and C, as suggested in [28]. Each inception module has convolution layers, pooling layers, and ReLU activation units. Convolution layers perform several convolution operations such as edge detection, sharpening, and blurring. Pooling layers involve different pooling operations, such as average pooling and max pooling. Pooling is carried out to reduce the number of parameters when the video frame is too large. ReLU is the most frequently used activation function, which has fewer computations than sigmoid and tanh. This model also has concatenation filters after each module to combine and pass the results to the next module.

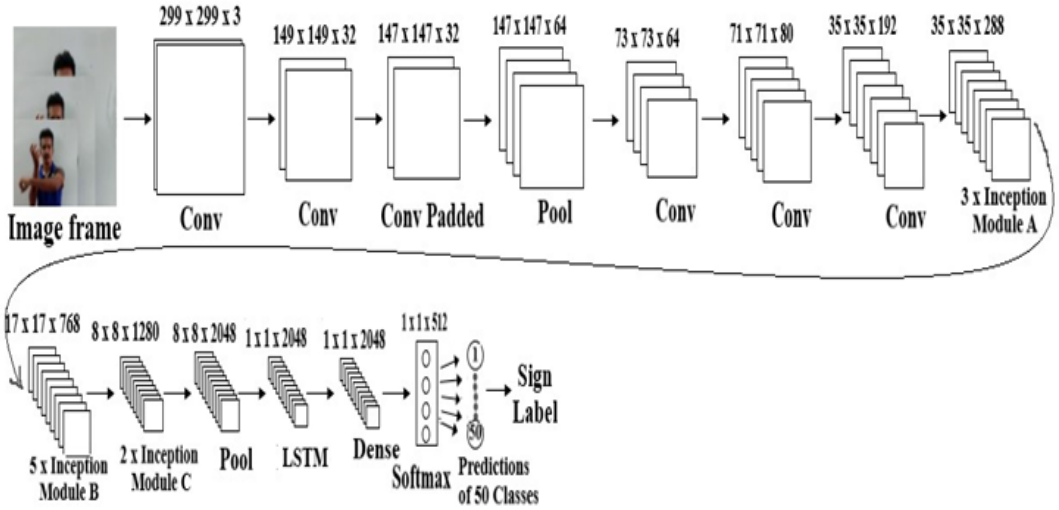

Figure 5. Hybrid CNN-RNN Architecture of the proposed Sign Language Recognition system

All the frames from each video are passed through the Inception V3-CNN model to perform feature extraction. From the final block of the Inception V3-CNN model, the drop out layer, fully connected layer and the softmax classifier are chopped off since the proposed system uses the model for feature

extraction purpose only. So, the 1 x 1 x 2048 dimensional feature vector from the final pooling layer (Global Average Pooling (GAP)) is considered as the output of the Inception V3-CNN model and given as the input to the LSTM-RNN model for learning the temporal dependencies. This stage will produce 2048 x N node values for the given input video, where N is the number of frames fed as the input to CNN.

The feature maps from the Inception V3-CNN model are to be grouped to form feature sequences since feature maps are produced for each video frame. Hence, a feature sequence is created for all the videos of the training dataset. The generated feature sequences are then given to the LSTM-RNN for the training and classification of hand gestures. The proposed system uses an efficient shallow RNN network, as shown in Fig. 5. The initial LSTM layer has 2048 nodes to take the 2048-dimensional feature sequence as the input. The LSTM layer is followed by a 512 nodes dense layer which is connected to the softmax classifier with 50 nodes, as we are classifying the gesture video as one of the ISL words of the proposed CasTalk-ISL dataset, which comprises samples belonging to 50 ISL words.

3.4 Experimental Dataset

The prime objective of the proposed research is to recognize the words of the Indian Sign Language (ISL). Earlier, no public dataset is available for this work since sign language varies from region to region. Researches on ISL have been promoted to the next level after introducing the first-ever ISL dictionary in 2018. The dictionary has one video sample for each of the 3000 ISL words. The dictionary was discussed in detail in section 1. To feed the data hungry neural network models, we need to create many samples for each word to be recognized. Hence, we have created our own dataset named "CasTalk–ISL Dataset." The dataset was created with the help of students of the National Engineering College, Kovilpatti, Tamil Nadu, India.

To start with, we have chosen 50 essential words from the 3000 ISL words. For each ISL word, our dataset has 100 video samples recorded from 10 subjects with 10 repetitions. The dataset has 5000 videos with a resolution of 1280 x 720 and 30 fps. As in Fig. 6, the average frame length varies from word to word since the subjects are given the freedom to articulate the hand gestures at their own pace of time and orientations. Also, there is no restriction for them in the dress code too. The videos are recorded with different backgrounds using smartphone cameras. The subjects are carefully chosen with variations in skin tone and body postures to increase the system's robustness towards handling the soft biometric issues. Table 1 portrays the list of the ISL words used in the proposed research.

Here, i denote the number of sign gestures, j denotes the number of persons involved, and k denotes the number of samples recorded from each person.

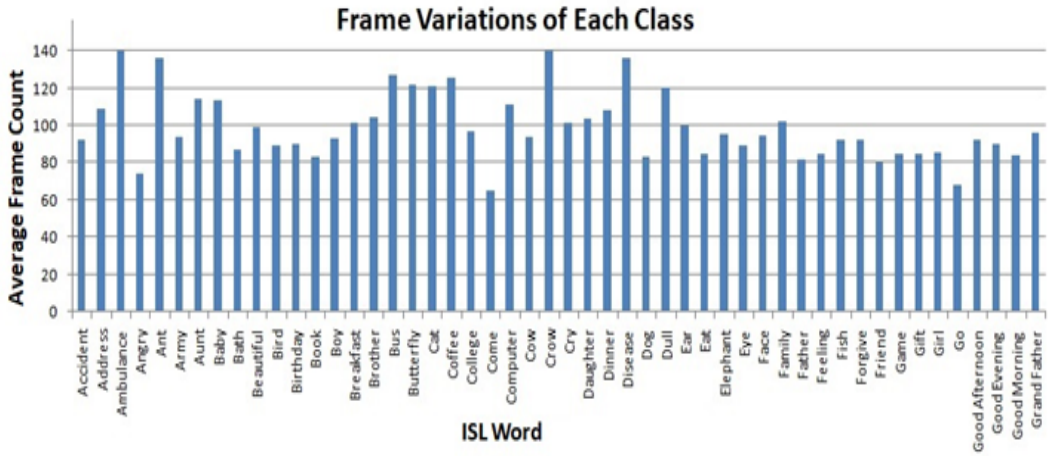

Figure 6. Average frame length of Each Word of the Dataset

Table 1. ISL Words of CasTalk-ISL Dataset

Sign_Id	ISL Word	Sign_Id	ISL Word	Sign_Id	ISL Word
S1	Accident	S18	Butterfly	S35	Eye
S2	Address	S19	Cat	S36	Face
S3	Ambulance	S20	Coffee	S37	Family
S4	Angry	S21	College	S38	Father
S5	Ant	S22	Come	S39	Feeling
S6	Army	S23	Computer	S40	Fish
S7	Aunt	S24	Cow	S41	Forgive
S8	Baby	S25	Crow	S42	Friend
S9	Bath	S26	Cry	S43	Game
S10	Beautiful	S27	Daughter	S44	Gift
S11	Bird	S28	Dinner	S45	Girl
S12	Birthday	S29	Disease	S46	Go
S13	Book	S30	Dog	S47	Good Afternoon
S14	Boy	S31	Dull	S48	Good Evening
S15	Breakfast	S32	Ear	S49	Good Morning
S16	Brother	S33	Eat	S50	Grand Father
S17	Bus	S34	Elephant		

The above table represents the words of ISL Gestures of CasTalk-ISL dataset. Here S1 to S50 indicates the sign id and they are mapped with the class label of the proposed sign language recognition system.

3.5. Implementation

The dataset is organized in such a way that all the P_j (j =1, 2….100) samples of sign S_i (i=1, 2…50) are grouped in the same directory, which is named with the corresponding class label. The dataset is further classified into the training set and testing set in 70:30 ratios.

The raw videos are given as the input to the system, as mentioned in Fig. 4. These videos are processed, and the frames are extracted from all the videos and stored separately. The videos with less than 25 frames and greater than 230 frames are skipped and excluded from the training and the testing dataset. Here, v is any video with the satisfied frame count, as mentioned above. W, H represents the width and height and N_f is the number of frames in the video v .

$$v^{W \times H \times N_f} \in \{V_{i,j,k}\} \quad (11)$$

In the videos having 30 fps, there will not be many changes from each frame to the next frame; hence to avoid the redundancy of features, key frames are extracted from the videos [43]. Every fifth frame (r = 5) from the N_f frames is taken from each video v with the assumption that those five frames are highly similar to each other. So, N_f frames are reduced to N_f/r frames, here r is the redundancy reduction coefficient.

$$v_r = v^{W \times H \times \frac{N_f}{r}} \quad (12)$$

The selected key frames from all the videos are passed through the Inception V3 ConvNet. Before passing to the network, the image frames are resized to 299x299x3. The resized image frames are passed through the Inception V3 network. The inception modules have convolutional layers and pooling layers with ReLU activators. The concatenated result from each inception block is given as input to the successor block.

The transitions happening to the input in the inception network are explained in Fig. 5. As the network goes deeper, the number of filters is also increased. Also, Ref. 7 states that any n x n convolution can be replaced by a 1 x n convolution followed by a n x 1 convolution, and the computational cost decreases as n increases. In the consecutive inception blocks, 35 x 35 grid is reduced to 17 x 17 and further reduced to 8 x 8 [7]. The final result of the Inception module 11 of size 8 x 8 x 2048 is passed to the Global Average Pooling (GAP) layer to produce a 2048 dimensional feature map.

$$v_{f_m} = V3\_CNN(v_r) \quad (13)$$

Here v_{f_m} is the feature map generated for each of the key frames of the video v. This feature map v_{f_m} is a 2048 dimensional output of the final pooling layer (GAP) with 1 x 1 x 2048 dimension. The feature maps of all the frames of each video are stitched together to form a feature sequence. These sequences have the feature maps of all the selected key frames. Hence, a feature sequence v_{f_s} is created for all the key frames in the gesture video.

$$v_{f_s} = v_{f_m} \times \frac{N_f}{r} \quad (14)$$

The extracted feature sequences are passed to the LSTM network to learn the temporal dependencies of the features. The proposed LSTM network is a three-layered shallow network with the initial LSTM layer and a middle dense layer followed by a softmax classification output layer. The output of the LSTM layer is passed to the immediate dense layer (MLP). The dense layer passes the data sequence (d_s) to the 50 class softmax layer. The softmax layer calculates the prediction probabilities for all the classes as the desired class for the given d_s. The class with the highest probability is treated as the desired gesture class (g_c), and the corresponding sign label is returned. These processes are explained through the Eq. (15), (16), and (17).

$$o_t = LSTM(v_{f_s}) \quad (15)$$

$$d_s = MLP(o_t) \quad (16)$$

$$g_c = \sigma(d_s), \qquad g_c \in S,\ s.t\ g_c = S_i \quad (17)$$

The feature extraction from the videos of the CasTalk –ISL dataset using Inception V3-CNN took 28 hours of computation on Intel i5 8th generation processor with 8 GB RAM. Training of the LSTM took 2 hours of computation on Google Colaboratory GPU [44] environment. Training, validation, and testing phases of the proposed sign language recognition system are carried out on the Google Colaboratory GPU environment since it is very much faster than the usual CPU environment.

4. EXPERIMENT AND ANALYSIS

The implementation results have been analysed and discussed by considering Top-1 accuracy and Top-k accuracy. Top-k accuracy is calculated for two trails by initialising k as 3 and 5, respectively. In each trial, the model with the most negligible validation loss is saved as the final model. From Table 2, it is observed that there is only a marginal difference in the accuracy values for trail 1 and trail 2. This shows the stability of the system. The Top-1 accuracy of the Top-3 approach in the training phase is lesser than that of the Top-5 approach since the training loss of the Top-3 approach is slightly higher than that of the Top-5 approach. Eventually, the Top-3 approach is considered for evaluating the performance metrics of the proposed sign language recognition system since it has better Top-1 accuracy in the validation and testing phases. Also, the validation loss 0.105 is better than 0.119 of the Top-5 approach. So, the model with 0.105 validation loss is chosen as the final model of the proposed system. The trained model has correctly classified 1438 videos of 1498 videos in the test set of the CasTalk-ISL dataset when the Top-1 label is considered. Also, the system correctly classified 1490 gesture videos of the test

set when the Top-3 predicted labels are considered. Table 2 provides a detailed analysis of the performance metrics in the training, validation and testing phases.

The signs "S10-Beautiful", "S31-Dull", and "S36-Face" involve the same sequence of actions and have more similarity. Despite the similarity, the system correctly classified all the samples of the gesture signs "S10-Beautiful" and "S36-Face" but wrongly classified 3 samples of "S31-Dull" as "S36-Face". Similarly, the actions involved in the "S21-College" and "S22-Come" are closer to each other, the hand movements of the signs "S28-Dinner", "S33-Eat" and "S34-Elephant" are similar to each other, and the signs "S47-Good Afternoon", "S48-Good Morning" and "S49-Good Evening" are very much similar to each other and have only slight variations.

In these cases, the system wrongly classifies some samples one sign as the other sign. Similarities among the gestures are also played a vital role in the classification rate. The classification rate for all the classes is ranging from 83% to 100%. Since we have used the equally distributed dataset, the system completely avoids the overfitting problem, and it can be proved from the depicted confusion matrix in Fig. 7. Apart from those similar signs, the signs "S2-Address" and "S12-Birthday", "S14-Boy" and "S16-Brother", and "S41-Forgive" and "S46-Go" have significant similarity between them. Despite this much similarity among the signs in the dataset, the developed dynamic hand gesture recognition system achieves as high as the 96% recognition rate.

Table 2. Comparison between Top-3 and Top-5 Approaches

Phases	Performance Metric	Top-3 Approach	Top-5 Approach
Training	Accuracy	**85.52**	87.09
	Loss	0.432	0.410
	Top k accuracy	99.62	99.25
Validation	Accuracy	**97.34**	96.17
	Loss	0.105	0.119
	Top k accuracy	99.92	100
Testing	Accuracy	**95.99**	95.32
	Top k accuracy	99.46	99.93

In the confusion matrix, the labels S1 to S50 represents the ISL words as given in Table 1. 1 sample out of 30 samples of the "S21-College" sign is classified as "S22-Come", and 3 samples out of 30 samples of "S22-Come" are classified as "S21-College". One out of 30 samples of the "S35-Eye" class is classified as "S32-Ear," and vice versa. The actions of class "S46-Go" is similar to the classes "S41-Forgive," and "S47-Good Afternoon," and this is reflected in the classification as some samples are wrongly classified among the three classes. In the signs "S47-Good Afternoon," "S48-Good Morning," and "S49-Good Evening," the right hand is moved upwards and downwards at the same angle. Only a few frames at the end are changed to give a particular meaning. Hence, the system struggled a little bit to classify few samples, as it classified

2 out of 30 samples of "S48-Good Evening" as "S47-Good Afternoon" and 4 out of 30 samples of "S47-Good Afternoon" as "S48-Good Evening".

The performance evaluation metrics, precision, and recall values are calculated using the given confusion matrix as,

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{18}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{19}$$

Both precision and recall are useful metrics for evaluating the performance measures of the system. Sometimes we need to combine both precision and recall to produce another metric to evaluate the performance as,

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{20}$$

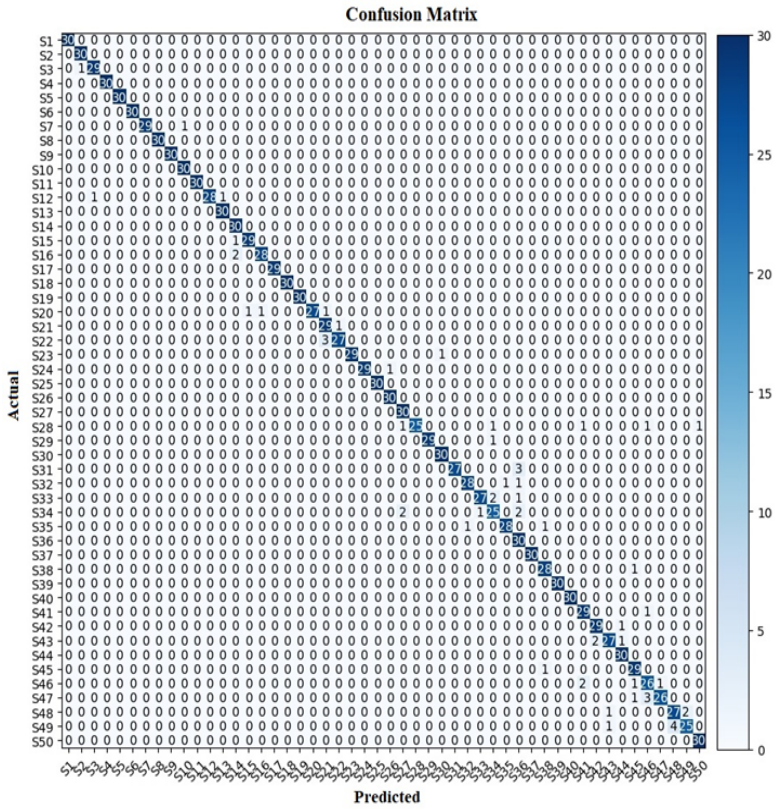

Figure 7. Confusion matrix of the proposed SLR system

F1 score is the harmonic mean of precision and recall, and it gives equal importance to the two metrics. The optimal classification model is chosen as the model with balanced precision and recall values and maximized F1 score. Also, it is stated that if the F1 score is high, then the precision and recall values of the model indicates good results. The overall performance measurement

values of the proposed hybrid CNN-RNN network for sign language recognition are given in Table 3.

Table 3. Performance metrics of the proposed sign language recognition system

Accuracy	Precision	Recall	F1-Score
95.99%	96.10%	95.99%	96.05%

5.1 Comparison with state-of-the-art Approaches

As discussed in the beginning note, gesture recognition has many applications. Concerning the application needs, model type, and the dataset is chosen. Some researchers have used publicly available datasets, while others have created new datasets like the proposed CasTalk-ISL dataset. The details about the state-of-the-ar have evaluated the performance of their model against various datasets. The table includes both static image datasets and dynamic video datasets used in the literature with the implementation results. The models recognizing static images achieve a higher recognition rate when compared with models deal with dynamic videos. But real-time gesture recognition systems should have the capability to recognize the gestures from dynamic videos.

Table 4. Comparison with the state-of-the-art gesture recognition researches

	Architecture	Dataset	Result
Our Approach	**Inception V3 (CNN) + LSTM (RNN)**	**CasTalk-ISL**	**95.99**
Yanqiu Liao [42]	B3D ResNet	DEVISIGN-D	89.8
		SLR_Dataset	86.9
Danilo Avola [22]	DLSTM	DHG-14 Gest	97.62
		DHG-28 Gest	91.43
Zhi-Jie Liang [2]	3DCNN + Contour	SLVM	87.6
	3DCNN + Infrared		88.3
	3DCNN + multi model fusion		89.2
Marwa Elpeltagy [6]	(HOG–PCA) + (CCA)	Indian	60.4
	(HOG–PCA) + (COV3DJ) + (CCA)	ChaLearn	83.12
Sarfaraz Masood [4]	CNN + LSTM	LSA	80.87
	CNN + LSTM + Pooling		95.21
Lionel Pigou [5]	CNN + ReLU	CLAP 14	95.68

Table 4 gives a comparative analysis of the recognition rate of the proposed system with some state-of-the-art gesture recognition systems. On keeping an eye on dynamic gesture recognition, the hybrid CNN-RNN architecture was chosen, and it paid the price by achieving 96% recognition accuracy. By considering the efficiency and classification rate, Inception V3-CNN has been selected, and LSTMs also have proven results in handling the sequences; hence, it was chosen. DLSTM [22] has 97.62% accuracy on the DHG-14 Gest

dataset, which has 14 gestures, but the accuracy is dropped when the model was tested with the DHG-28 Gest dataset. Masood's [4] system achieved 95.21% recognition rate on 46 gestures LSA64 [15] dataset, which is better than Piagou's [5] system, which was trained to recognize 20 gestures. The proposed system recognizes 50 gestures of the CasTalk-ISL dataset with 95.99% recognition rate is better than the similar systems considered in the table.

5. CONCLUSIONS

This paper presents the research carried out to achieve dynamic hand gesture recognition from real-time videos. This is achieved by using the hybrid architecture of Inception V3–CNN and LSTM-RNN to perform sign language recognition at a moderate level by considering the 50 categories of ISL words. The proposed work is part of our product development research. The goal of the product is to provide a real-time Sign Language to speech translation service. The proposed system achieves a 96% recognition rate on the dynamic video dataset by infusing various challenges such as background subjectivity, sensor resolutions to adopt the real-world requirements. The obtained results conclude the system's robustness and stability, hence making it suitable for real-time gesture video translation. The results that we obtained are better than the similar systems used to perform dynamic gesture recognition. These results are evident to show that CNN, along with RNN, could be a successful solution to video classification tasks. In this version, the system can recognize 50 signs of ISL words, and it is planned to work with 100 signs with lesser data samples; this will reduce the model complexity and the processing time by a higher margin. Also, it is planned to work with systems capable of recognizing the sentences of sign languages.

ACKNOWLEDGEMENTS

The proposed research was carried out with the support of the Institutional Research Fellowship (IRF) scheme of National Engineering College, Tamil Nadu, India, and the EPICS in IEEE project grant (Grant reference number: 2016-8) NJ, USA. The research team expresses heartfelt thanks to the student volunteers of National Engineering College for their significant contribution to the CasTalk-ISL dataset. Also, the team thanks Google for its Google Colaboratory GPU environment and acknowledges the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] Shweta Dour, **Real time recognition of Indian sign language**, Ph.D. dissertation, Dept. Elect. Comm. Eng., Bhagwant Univ., Ajmer, Rajasthan, India, 2017. http://shodhganga.inflibnet.ac.in/handle/10603/153744.

[2] Zhi-jie Liang, Sheng-bin Liao, Bing-zhang Hu, **3D Convolutional Neural Networks for dynamic sign language recognition**, *The Computer Journal*,

vol. 61, no. 11, Nov. 2018, 1724-1736. https://doi.org/10.1093/comjnl/bxy049.

[3] P. V. V. Kishore, D. Anil Kumar, A. S. Chandra Sekhara Sastry, E. Kiran Kumar, **Motionlets matching with adaptive kernels for 3-D Indian Sign Language Recognition**, *IEEE Sensors Journal*, vol. 18, no. 8, Apr 2018, 3327-3337. https://doi.org/10.1109/JSEN.2018.2810449.

[4] S. Masood, A. Srivastava, H.C. Thuwal, M. Ahmad, **Real-time sign language gesture (word) recognition from video sequences using CNN and RNN**, *Intelligent Engineering Informatics*, Advances in Intelligent Systems and Computing, vol. 695, Apr. 2018, 623-632. https://doi.org/10.1007/978-981-10-7566-7_63.

[5] Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, Benjamin Schrauwen, **Sign language recognition using Convolutional Neural Networks**, *European Conference on Computer Vision- ECCV 2014 Workshops*, Lecture Notes in Computer Science, vol. 8925, Mar. 2015, 572-578. https://doi.org/10.1007/978-3-319-16178-5_40.

[6] Marwa Elpeltagy, Moataz Abdelwahab, Mohamed E. Hussein, Amin Shoukry, Asmaa Shoala, Moustafa Galal, **Multi-modality-based Arabic Sign Language recognition**, *IET Computer Vision*, vol. 12, no. 7, Oct 2018, 1031-1039. https://doi.org/10.1049/iet-cvi.2017.0598.

[7] Biplab Ketan Chakraborty, DebajitSarma, M.K. Bhuyan, Karl F MacDorman, **Review of constraints on vision-based gesture recognition for human–computer interaction**, *IET Computer Vision*, vol. 12, no. 1, Jan. 2018, 3-15. https://doi.org/10.1049/iet-cvi.2017.0052.

[8] Sushmita Mitra, Tinku Acharya, **Gesture recognition: A survey**, *IEEE Transactions on Systems, Man, and Cybernetics*, Part C (Applications and Reviews), vol. 37, no. 3, Apr. 2007, 311-324. https://doi.org/10.1109/TSMCC.2007.893280.

[9] Kouichi Murakami, Hitomi Taguchi, **Gesture recognition using Recurrent Neural Networks**, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Louisiana, USA, May. 1991, 237-242. https://doi.org/10.1145/108844.108900.

[10] Kenneth Lai, Svetlana N. Yanushkevich, **CNN+RNN depth and skeleton based dynamic hand gesture recognition**, *24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, Aug. 2018. https://doi.org/10.1109/ICPR.2018.8545718.

[11] Juan C. Nunez, Raul Cabido, Juan J. Pantrigo, Antonio S. Montemayor, Jose F. Velez, **Convolutional Neural Networks and Long Short-Term Memory for skeleton-based human activity and hand gesture recognition**, Pattern Recognition, vol. 76, Apr. 2018, 80-94. https://doi.org/10.1016/j.patcog.2017.10.033.

[12] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li and Weiping Li, **Video-based Sign Language Recognition without Temporal Segmentation**, *AAAI Conference on Artificial Intelligence (AAAI)*, 2018. http://home.ustc.edu.cn/~pjh/openresources/cslr-dataset-2015/index.html.

[13] Q. De Smedt, H. Wannous, J.P. Vandeborre, **Skeleton-based dynamic hand gesture recognition**, *The IEEE Conference on Computer Vision and Pattern*

Recognition Workshops (CVPRW), Jun. 2016, 1206–1214. https://doi.org/10.1109/cvprw.2016.153.

[14] Hanjie Wang, Xiujuan Chai, Xiaopeng Hong, Guoying Zhao, and Xilin Chen, **Isolated sign language recognition with Grassmann covariance matrices**, *ACM Transactions on Accessible Computing (TACCESS)*, vol. 8, no. 4, May 2016. http://dx.doi.org/10.1145/2897735.

[15] Ronchetti, F., Quiroga, F., Estrebou, C.A., Lanzarini, L.C., Rosete, A, **LSA64: an Argentinian sign language dataset**, *XXII Congreso Argentino de Ciencias de la Computación (CACIC)*, (2016). http://facundoq.github.io/unlp/lsa64/.

[16] Zafar Ahmed Ansari, Gaurav Harit, **Nearest neighbour classification of Indian sign language gestures using kinect camera**, Sadhana, 41, Feb. 2016, 161 – 182. https://doi.org/10.1007/s12046-015-0405-3.

[17] Escalera, S., Bar, X., Gonzlez, J., Bautista, M.A., Madadi, M., Reyes, M., Ponce, V., Escalante, H.J., Shotton, J., Guyon, Cha, **Learn Looking at People Challenge 2014: Dataset and Results, Computer Vision - ECCV 2014 Workshops**, Lecture Notes in Computer Science, vol 8925. https://doi.org/10.1007/978-3-319-16178-5_32.

[18] S. Escalera, J. Gonzàlez, X. Baró, M. Reyes, O. Lopes, I. Guyon, V. Athistos, H.J. Escalante, **Multi-modal Gesture Recognition Challenge 2013: Dataset and Results**, *ICMI* 2013. http://www.maia.ub.es/~sergio/linked/ps-p445-escaleraps.pdf.

[19] Shugao Ma, Sarah Adel Bargal, Jianming Zhang, Leonid Sigal, Stan Sclaroff, **Do less and achieve more: Training CNNs for action recognition utilizing action images from the Web**, Pattern Recognition, vol. 68, Aug 2017, 334-345. https://doi.org/10.1016/j.patcog.2017.01.027.

[20] Julien Maitre, Clement Rendu, Kevin Bouchard, Bruno Bouchard, Sebastien Gaboury, **Basic daily activity recognition with a data glove**, *The 10th International Conference on Ambient Systems, Networks and Technologies (ANT)*, vol. 151, May 2019, 108-115. https://doi.org/10.1016/j.procs.2019.04.018.

[21] Wen-Ren Yang, Chau-Shing Wang, Chien-Pu Chen, **Motion-pattern recognition system using a wavelet-neural network**, *IEEE Transactions on Consumer Electronics*, vol. 65, no. 2, May 2019, 170-178. https://doi.org/10.1109/TCE.2019.2895050.

[22] Danilo Avola, Marco Bernardi, Luigi Cinque, Gian Luca Foresti, Cristiano Massaroni, **Exploiting Recurrent Neural Networks and leap motion controller for the recognition of sign language and semaphoric hand gestures**, *IEEE Transactions on Multimedia*, vol. 21, no. 1, Jan. 2019, 234-245. https://doi.org/10.1109/TMM.2018.2856094.

[23] *Bo Li,* Chao Zhang, Cheng Han, Baoxing Bai, **Gesture Recognition Based on Kinect v2 and Leap Motion Data Fusion**, *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 33, no. 05, 1955005 (2019). https://doi.org/10.1142/S021800141955005X.

[24] Shuiwang Ji, Wei Xu, Ming Yang, Member, Kai Yu, **3D Convolutional Neural Networks for human action recognition**, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, Jan. 2013. https://doi.org/10.1109/TPAMI.2012.59.

[25] Kensho Hara, Hirokatsu Kataoka, Yutaka Satoh, **Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition**, Aug 2017. [https://arxiv.org/abs/1708.07632v1].

[26] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, **Faster R-CNN: Towards real-time object detection with region proposal networks**, Jan. 2016. [https://arxiv.org/abs/1506.01497v3].

[27] Karen Simonyan, Andrew Zisserman, **Very deep Convolutional Networks for large-scale image recognition**, *ICLR 2015*, Apr. 2015. https://arxiv.org/abs/1409.1556v6.

[28] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna, **Rethinking the Inception architecture for computer vision**, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, Dec. 2016. https://doi.org/10.1109/CVPR.2016.308.

[29] Ilias Papastratis, Kosmas Dimitropoulos, Dimitrios Konstantinidis, Petros Daras, **Continuous Sign Language Recognition Through Cross-Modal Alignment of Video and Text Embeddings in a Joint-Latent Space**, *IEEE Access*, Vol. 8, May 2020, 91170-91180. https://doi.org/10.1109/ACCESS.2020.2993650.

[30] Muneer Al-Hammadi, Ghulam Muhammad, Wadood Abdul, Mansour Alsulaiman, Mohamed A. Bencherif, Mohamed Amine Mekhtiche, **Hand Gesture Recognition for Sign Language Using 3DCNN**, *IEEE Access*, Vol. 8, May 2020, 79491-79509. https://doi.org/10.1109/ACCESS.2020.2990434.

[31] Safa Ameur, Anouar Ben Khalifa, Med Salim Bouhlel, **A novel hybrid bidirectional unidirectional LSTM network for dynamic hand gesture recognition with Leap Motion**, *Entertainment Computing*, Vol 35, June 2020. https://doi.org/10.1016/j.entcom.2020.100373.

[32] Elahe Rahimian, Soheil Zabihi, Seyed Farokh Atashzar, Amir Asif, Arash Mohammadi, **Surface EMG-Based Hand Gesture Recognition via Hybrid and Dilated Deep Neural Network Architectures for Neurorobotic Prostheses**, Journal of Medical Robotics Research, Vol. 5, No. 1, Mar. 2020. https://doi.org/10.1142/S2424905X20410019.

[33] Sepp Hochreiter, Jurgen Schmidhuber, **Long Short-Term Memory**, *Neural Computation*, vol. 9, no. 8, Nov. 1997, 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735.

[34] Zhigang Tu, Hongyan Li, Dejun Zhang, Justin Dauwels, Baoxin Li, Junsong Yuan, **Action-stage emphasized spatiotemporal VLAD for video action recognition**, *IEEE Transactions on Image Processing*, vol. 28, no. 6, Jun. 2019, 2799-2812. https://doi.org/10.1109/TIP.2018.2890749.

[35] Caihua Liu, Jie Liu, Zhicheng He, YujiaZhai, Qinghua Hu, Yalou Huang, **Convolutional neural random fields for action recognition**, *Pattern Recognition*, vol. 59, Nov. 2016, 213-224. https://doi.org/10.1016/j.patcog.2016.03.019.

[36] PichaoWang, Wanqing Li, Philip Ogunbona, Jun Wan, Sergio Escalera, **RGB-D-based human motion recognition with deep learning: A survey**, *Computer Vision and Image Understanding*, vol. 171, Jun. 2018, 118-139. https://doi.org/10.1016/j.cviu.2018.04.007.

[37] Santanu Pattanayak, **Foundations of artificial intelligence-based systems**, in Intelligent Projects using Python: 9 real-world AI projects leveraging

machine learning and deep learning with TensorFlow and Keras, Packt Publishing, Birmingham, UK, 2019, Ch. 1. Sec. 13-14. [https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781788996921/1.

[38] Festus Osayamwen, Jules-Raymond Tapamo, **Deep learning class discrimination based on prior probability for human activity recognition**, *IEEE Access*, vol. 7, Feb. 2019, 14747 - 14756. https://doi.org/10.1109/ACCESS.2019.2892118.

[39] Di Wu, Lionel Pigou, Pieter-Jan Kindermans, Nam Do-Hoang Le, Ling Shao, Joni Dambre, Jean-Marc Odobez, **Deep Dynamic Neural Networks for multimodal gesture segmentation and recognition**, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 38, no. 8, Aug. 2016, 1583-1597. https://doi.org/10.1109/TPAMI.2016.2537340.

[40] Runpeng Cui, Hu Liu, Changshui Zhang, **A deep neural framework for continuous sign language recognition by iterative training**, *IEEE Transactions on Multimedia*, vol. 21, no. 7, Jul. 2019, 1880-1891. https://doi.org/10.1109/TMM.2018.2889563.

[41] Fangxin Wang, Wei Gong, Jiangchuan Liu, **On spatial diversity in wiFi-based human activity recognition: A deep learning-based approach**, *IEEE Internet of Things Journal*, vol. 6, no. 2, Apr. 2019, 2035-2047. https://doi.org/10.1109/JIOT.2018.2871445.

[42] Yanqiu Liao, PengwenXiong, WeidongMin, Weiqiong Min, Jiahao Lu, **Dynamic sign language recognition based on video sequence with BLSTM-3D residual networks**, *IEEE Access*, vol. 7, Mar. 2019, 38044 – 38054. https://doi.org/10.1109/ACCESS.2019.2904749.

[43] Hao Tang, Hong Liu, Wei Xiao, Nicu Sebe, **Fast and robust dynamic hand gesture recognition via key frames extraction and feature fusion**, *Neurocomputing*, vol.331, Feb. 2019, 424-433. https://doi.org/10.1016/j.neucom.2018.11.038.

[44] **Google Colaboratory** [https://colab.research.google.com/]