A DEEP NEURAL FRAMEWORK FOR CONTINUOUS SIGN LANGUAGE RECOGNITION BY ITERATIVE TRAINING

P. Sai Hamshika[1], S. Aishwarya[1], V. Ashritha[1], Ayub Baig[2]

[1,2]Department of Information Technology

[1,2]Malla Reddy Engineering College for Women (A), Maisammaguda, Medchal, Telangana.

ABSTRACT

This work develops a continuous sign language (SL) recognition framework with deep neural networks, which directly transcribes videos of SL sentences to sequences of ordered gloss labels. Previous methods dealing with continuous SL recognition usually employ hidden Markov models with limited capacity to capture the temporal information. In contrast, our proposed architecture adopts deep convolutional neural networks with stacked temporal fusion layers as the feature extraction module, and bi-directional recurrent neural networks as the sequence learning module. We propose an iterative optimization process for our architecture to fully exploit the representation capability of deep neural networks with limited data. Our proposed neural model consists of two modules for spatiotemporal feature extraction and sequence learning, respectively. Due to the limited scale of the datasets, we find an end-to-end training cannot fully exploit the deep neural network of high complexity. To address this problem, this work investigates an iterative optimization process to train our convolutional neural network based bidirectional long-short-term-memory (CNN-BILSTM) architecture effectively. We use gloss-level gestural supervision given by forced alignment from end-to-end system to directly guide the training process of the feature extractor. Afterwards, this work fine-tunes the BILSTM system with the improved feature extractor, and the system can provide further refined alignment for the feature extraction module. Through this iterative training strategy, the proposed CNN-BILSTM can keep learning and benefiting from the refined gestural alignments. To implement this project, ‘SignumDataset’ dataset is used, which contains 24 different signs or signatures.

Keywords: Sign language, iterative learning, deep neural networks.

1. INTRODUCTION

Sign language (SL) is commonly known as the primary language of deaf people, and usually collected or broadcast in the form of video. SL is often considered as the most grammatically structured gestural communications [1]. This nature makes SL recognition an ideal research field for developing methods to address problems such as human motion analysis, human-computer interaction (HCI) and user interface design, and makes it receive great attention in multimedia and computer vision.

Typical SL learning problems involve isolated gesture classification [2] sign spotting and continuous SL recognition. Generally speaking, gesture classification is to classify isolated gestures to correct categories, while sign spotting is to detect predefined signs from continuous video streams, with precise temporal boundaries of gestures provided for training detectors. Different from these problems, continuous SL recognition is to transcribe videos of SL sentences to ordered sequences of glosses (here we use “gloss” to represent a gesture with its closest meaning in natural languages), and the continuous video streams are provided without prior segmentation. Continuous SL recognition concerns more about learning unsegmented gestures of long-term video streams and is more suitable for processing continuous gestural videos in real-world systems. Its training also does not require an expensive annotation on temporal boundary for each gesture. Recognizing SL indicates simultaneous analysis and integration of gestural movements and appearance features, as well as disparate body parts, and therefore probably using a multimodal approach. In this paper, we focus on the problem of continuous SL recognition on videos, where learning the spatiotemporal representations as well as their temporal matching for the labels is crucial.

Many studies have made their efforts on representing SL with hand-crafted features. For example, hand and joint locations are used in local binary patterns (LBP) is used in, histogram of oriented gradients (HOG) is utilized in, and its extension HOG-3D is applied in. Recently, deep convolutional neural networks have achieved a tremendous impact on related tasks on videos, e.g. human action recognition gesture recognition [3] and sign spotting and recurrent neural networks (RNNs) have shown significant performance on learning the temporal dependencies in sign spotting. Several recent approaches taking advantage of neural networks have also been proposed for continuous SL recognition. In these works, neural networks are restricted to learning frame-wise representations, and hidden Markov models (HMMs) are utilized for sequence learning. However, the frame-wise labelling adopted in is noisy for training the deep neural networks, and HMMs might be hard to learn the complex dynamic variations, considering their limited representation capability.

This project therefore develops a recurrent convolutional neural network for continuous SL recognition. Our proposed neural model consists of two modules for spatiotemporal feature extraction and sequence learning respectively. Due to the limited scale of the datasets, we find an end-to-end training cannot fully exploit the deep neural network of high complexity. To address this problem, we investigate an iterative optimization process to train our recurrent deep neural architecture effectively. We use gloss-level gestural supervision given by forced alignment from end-to-end system to directly guide the training process of the feature extractor. Afterwards, we fine-tune the recurrent neural system with the improved feature extractor, and the system can provide further refined alignment for the feature extraction module. Through this iterative training strategy, our deep neural network can keep learning and benefiting from the refined gestural alignments.

2. LITERATURE SURVEY

Wu et al. [4] employ a deep belief network to extract high-level skeletal joint features for gesture recognition. Convolutional neural networks (CNNs) and 3D convolutional neural networks (3D-CNNs) have also been employed to capture visual cues for hand regions.

Molchanov et al. [5] apply 3D-CNNs for spatiotemporal feature extraction from video streams on color, depth and optical flow data. Neverova et al. [9] present a multi-scale deep architecture on color, depth data and handcrafted pose descriptors.

Pigou et al. [6] propose an end-to-end neural model with temporal convolutions and bidirectional recurrence for sign spotting, which is taken as frame-wise classification in their framework. However, with only weak supervision in sentence level, recurrent neural networks are hard to learn to match the over-length input sequence frame by frame with the ordered labels. Different from their model, we use temporal pooling layers to integrate the temporal dynamics before the bidirectional recurrence.

Buehler et al. [7] propose a scoring function based on multiple instance learning (MIL) and search for signs of interest by maximizing the score.

Pfister et al. [8] use subtitle text, lip and hand motion cues to select candidate temporal windows, and these candidates are further refined using MISVM [32].

Chung and Zisserman [9] use a ConvNet learned on image encoding representing human keypoint motion for recognition, and they locate temporal positions of signs via saliency map by back-propagation.

Gweth et al. [10] employ a one hidden-layer perceptron to estimate posterior from appearance based features, and use the probabilities as inputs to train an HMM-based recognition system.

Different from the previous work [11], this project proposes a distinctive segment-gloss alignment method to learn from the outputs of our sequence learning module, and we provide an explicit illustration for our iterative training scheme, by proving the training of feature extraction module to be maximizing the lower bound of the objective function, instead of using an intuitive approach. We also contribute by investigating more on the multimodal integration of appearance and motion cues in this work.

3. PROPOSED SYSTEM

Our proposed neural model consists of two modules for spatiotemporal feature extraction and sequence learning, respectively. Due to the limited scale of the datasets, we find an end-to-end training cannot fully exploit the deep neural network of high complexity. To address this problem, we investigate an iterative optimization process to train our recurrent deep neural architecture effectively. We use gloss-level gestural supervision given by forced alignment from end-to-end system to directly guide the training process of the feature extractor. Afterwards, we fine-tune the recurrent neural system with the improved feature extractor, and the system can provide further refined alignment for the feature extraction module. Through this iterative training strategy, our deep neural network can keep learning and benefiting from the refined gestural alignments. The main contributions of our work can be summarized as follows:

1) We develop our architecture with recurrent convolutional neural networks of more learning capacity to achieve state-of-the-art performance on continuous SL recognition, without importing extra supervisory information.

2) We design an iterative optimization process for training our deep neural network architecture, and our approach, with the neural networks better exploited, is proved to take notable effect on the limited training set in contrast to the end-to-end trained system.

3) We propose a multimodal version of our framework with RGB frames and optical flow frames and experiments present that our multimodal fusion scheme provides better representations for the gestures and further improves the performance of the system.

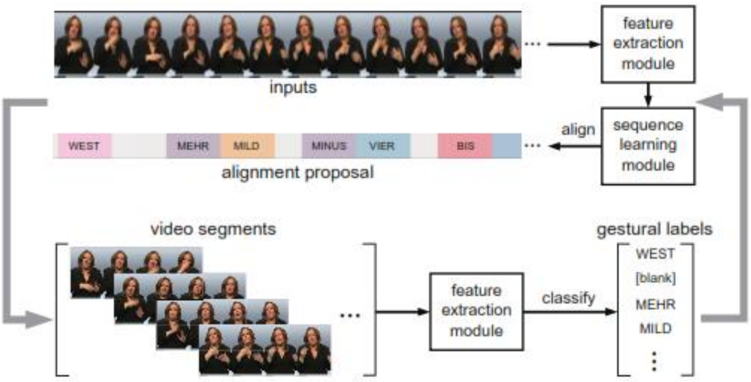


Fig. 1: Iterative training process.

Model design

The proposed deep neural architecture consists of a deep CNN followed by temporal operations for representation learning, and Bi-LSTMs for sequence learning. For experiments with modalities from dominant hands as the inputs, we build the deep convolutional network based on the VGG-S model (from layer conv1 to fc6), which is memory-efficient and shows competitive classification performance on ILSVRC-2012 dataset. The input frames, which are the region of dominant hands signped from original frames, are resized to 101 × 101 in dimension, and they are then transformed to 1024-dimensional feature vectors through the fully connected layer fc6. The stacked temporal convolution and pooling layers are utilized to generate spatiotemporal representation for each segment.

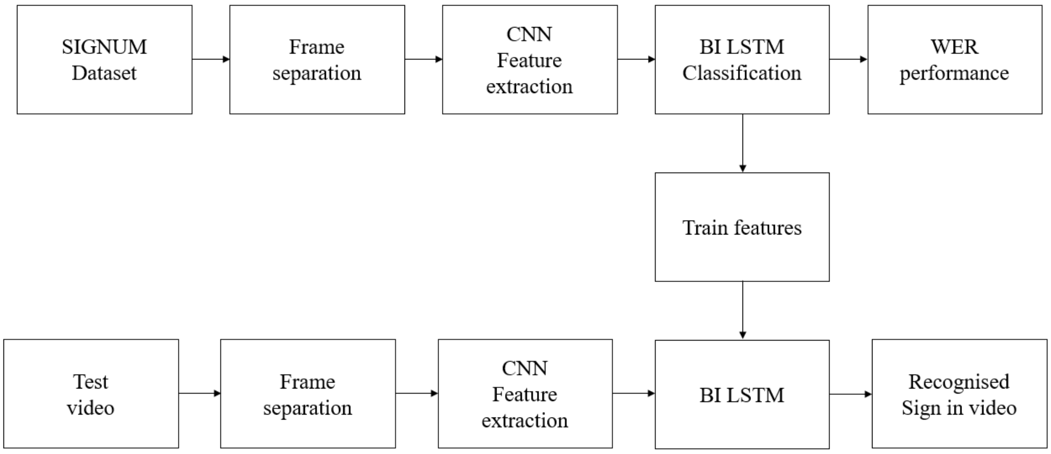


Fig. 2: Proposed block diagram.

Fig. 2 shows the block diagram of proposed method, which is used identify the different signs from the test video using CNN feature extraction and BI-LSTM training. We select the temporal stride δ to ensure sufficient overlapping between neighboring segments, as well as pool the representation sequence to a moderate length. In the feature extraction module, rectifier and max pooling are adopted for all the nonlinearity and pooling operations. We use Bi-LSTMs with 2 × 512 dimensional hidden states and peephole connections to learn the temporal dependencies. The hidden states are then fed into the SoftMax classifier, with the dimension equal to the vocabulary size. We also investigate the performance of our training framework with full video frames as the inputs. We use GoogLeNet and VGG-S net as the deep convolutional network in our feature extractor, and we adopt two stacked Bi-LSTMs to build the sequence learning module. Due to the limitations on GPU memory to fit in the whole system, we fix the parameters of CNN at the end-to-end stage and only tune the sequence learning module. The video frames are resized to 224×224 as the inputs of CNN, transformed to feature vectors after the average pooling layer, and then fed into the temporal fusion layers. The employed GoogLeNet is initialized with the weights pretrained on ILSVRC-2014 dataset, and we initialize the feature extractor by fitting it to the alignment proposal generated by the model end-to-end trained on dominant hand frames.

SIGNUM Dataset

The SIGNUM Database was created within the framework of a research project at the Institute of Man–Machine Interaction, located at the RWTH Aachen University in Germany. The SIGNUM (Signer-Independent Continuous Sign Language Recognition for Large Vocabulary Using Subunit Models) project was funded by the Deutsche Forschungsgemeinschaft (German Research Foundation) and aimed to develop a video-based automatic sign language recognition system. In order to ensure user-friendliness, the system utilizes a single-color video camera for data acquisition. Since sign languages make use of manual and facial means of expression both channels are analyzed by means of frame processing. The whole system, particularly the feature extraction and the subsequent classification stage, is designed for signer-independent operation and allows adaptation to an unknown signer. The reader interested in a more detailed description of this recognition system or an in-depth introduction to gesture and sign language recognition is directed to the publication list.

Multimodal Fusion

To incorporate the appearance and motion information, we also take color frame and optical flow for dominant hand regions as the inputs of our deep neural architecture. We adopt sum fusion approach at the conv5 layer for fusing the two stream networks. It computes element-wise sum of the two feature maps at the same spatial location and channel for the fusion. Our intention here is to put appearance and motion cues at the same spatial position in correspondence, without introducing extra filters in order to join the feature maps together. The sum fusion approach also shows a decent performance on the task of action recognition in video compared to other spatial fusion methods.

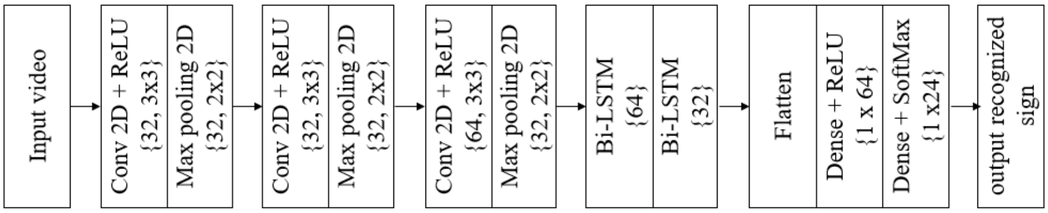


Fig. 3: Deep neural architecture for RGB and optical flow modalities of dominant hands.

Our end-to-end architecture for SL recognition from dominant hands is depicted in Fig. 3. Note that parameters for different modalities are not shared before the sum fusion. In experiments on multiple modalities of full frames, we adopt fusion of color and optical flow at two layers (after inception_3b and inception_4c in GoogLeNet) like.

BI_LSTM-CNN

According to the facts, training and testing of CNN involves in allowing every source data via a succession of convolution layers by a kernel or filter, rectified linear unit (ReLU), max pooling, fully connected layer and utilize SoftMax layer with classification layer to categorize the objects with probabilistic values ranging from.

Convolution layer is the primary layer to extract the features from a source image and maintains the relationship between pixels by learning the features of image by employing tiny blocks of source data. It's a mathematical function which considers two inputs like source image $I(x, y, d)$ where x and y denotes the spatial coordinates i.e., number of rows and columns. d is denoted as dimension of an image (here d=3 since the source image is RGB) and a filter or kernel with similar size of input image and can be denoted as $F(k_x, k_y, d)$..

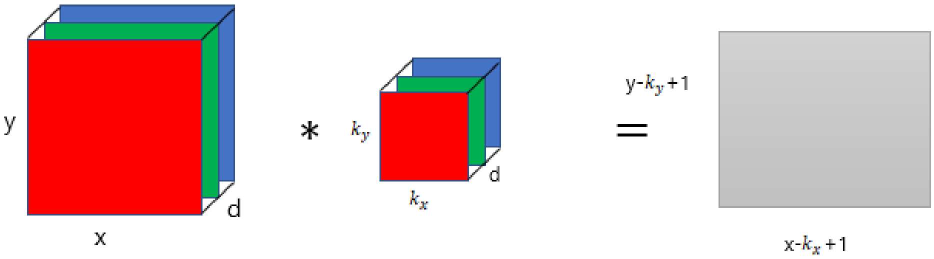


Fig. 4: Representation of convolution layer process.

The output obtained from convolution process of input image and filter has a size of $C\left((x-k_x+1),(y-k_y+1),1\right)$, which is referred as feature map. Let us assume an input image with a size of 5×5 and the filter having the size of 3×3. The feature map of input image is obtained by multiplying the input image values with the filter values.

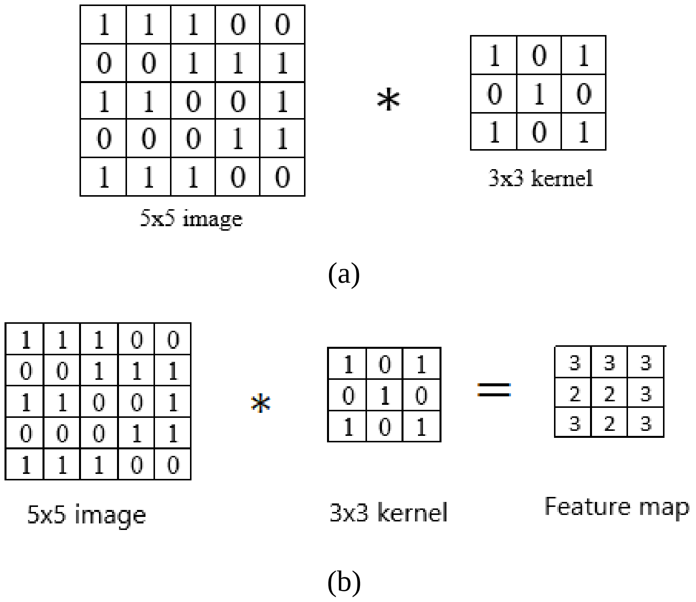
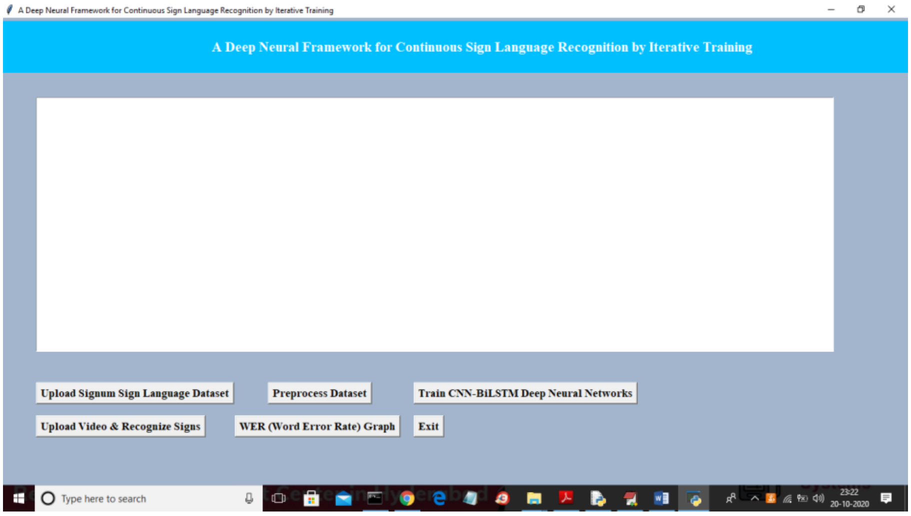


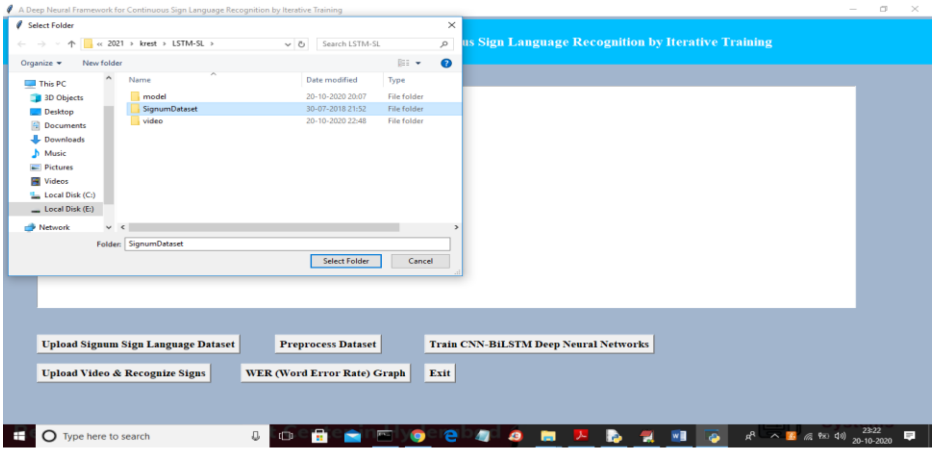
Fig. 5: Example of convolution layer process (a) an image with size 5×5 is convolving with 3×3 kernel (b) Convolved feature map.

4. RESULTS AND DISCUSSIONS

To run project double click on 'run.bat' file to get below screen


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search
23:22
20-10-2020

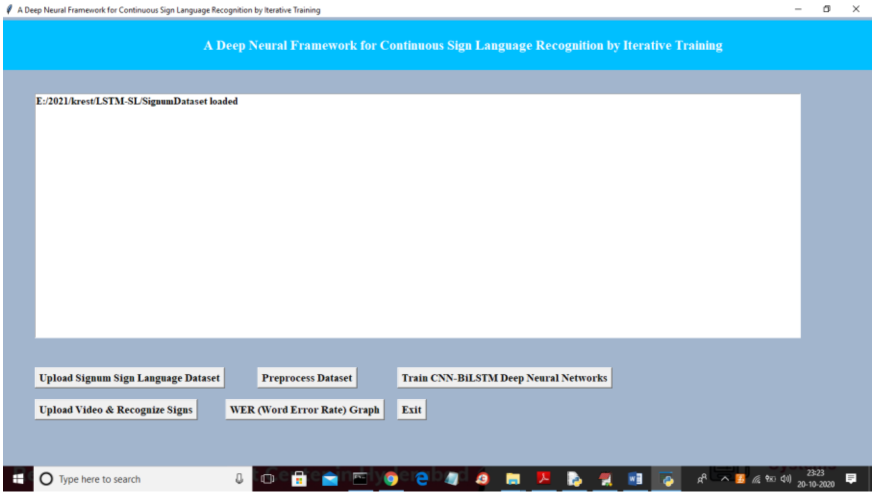
In above screen click on 'Upload Signum Sign Language Dataset' button and upload dataset folder


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Select Folder
« 2021 › krest › LSTM-SL ›
Search LSTM-SL
Organize
New folder
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
Network
Select Folder
Cancel
us Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search

Name	Date modified	Type
model	20-10-2020 20:07	File folder
SignumDataset	30-07-2018 21:52	File folder
video	20-10-2020 22:48	File folder

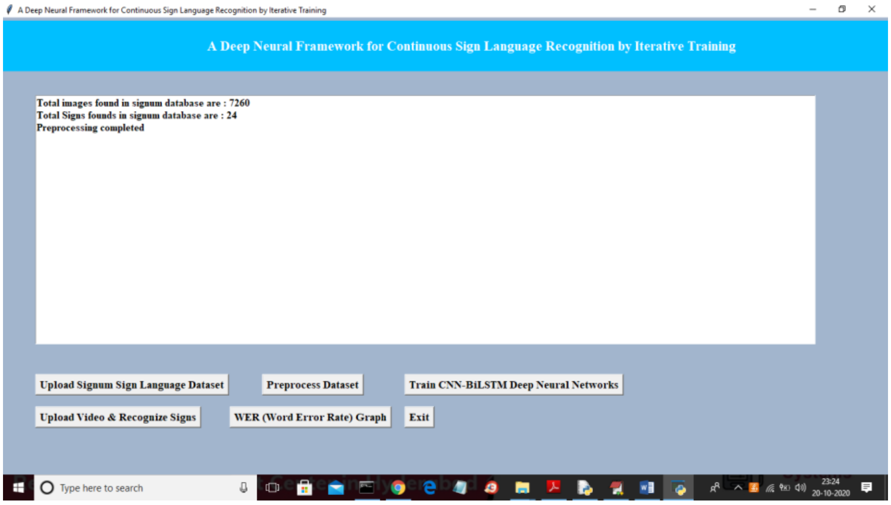
Folder: SignumDataset

In above screen selecting and uploading SignumDataset folder and then click on 'Select Folder' button to load dataset and to get below screen


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search

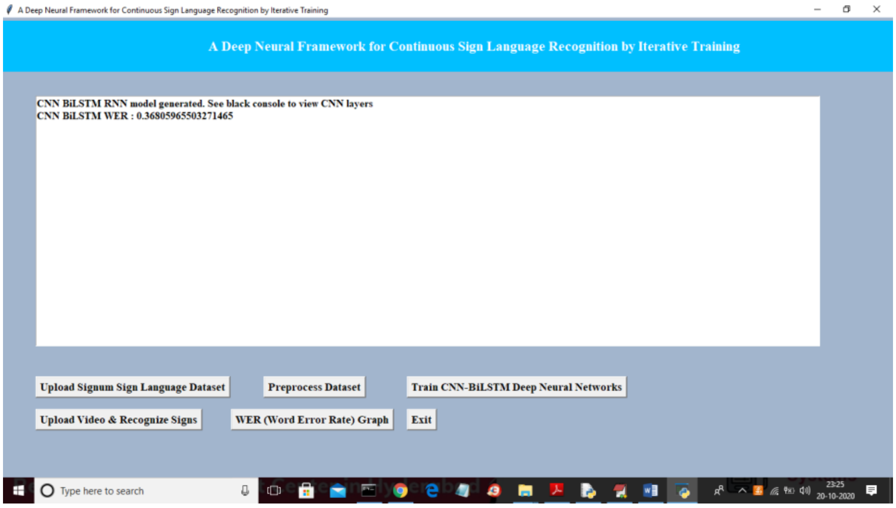
```
E:/2021/krest/LSTM-SL/SignumDataset loaded
```

Now in above screen click on 'Preprocess Dataset' button to read dataset images


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search

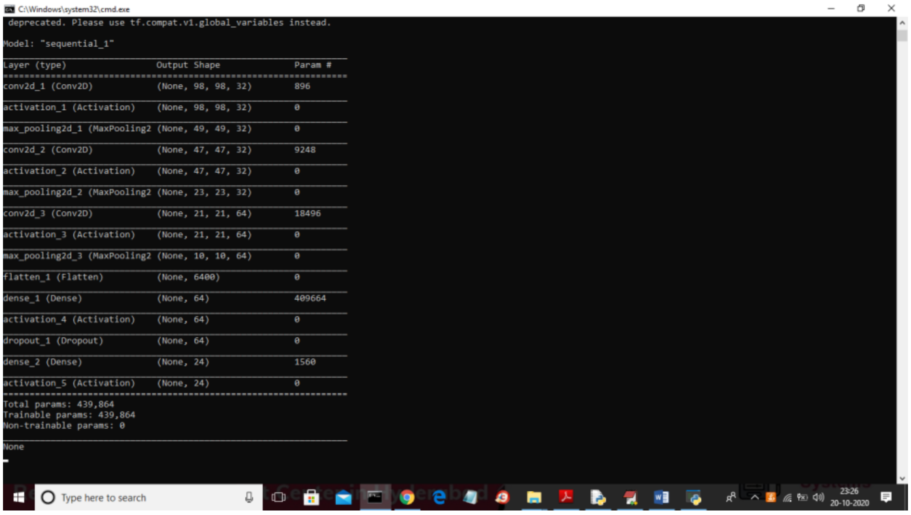
```
Total images found in signum database are : 7260
Total Signs founds in signum database are : 24
Preprocessing completed
```

In above screen application has found 7260 images of 24 different sign languages and now dataset images are ready and now click on 'Train CNN-BILSTM Deep Neural Networks' button to generate sign language recognition model


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search

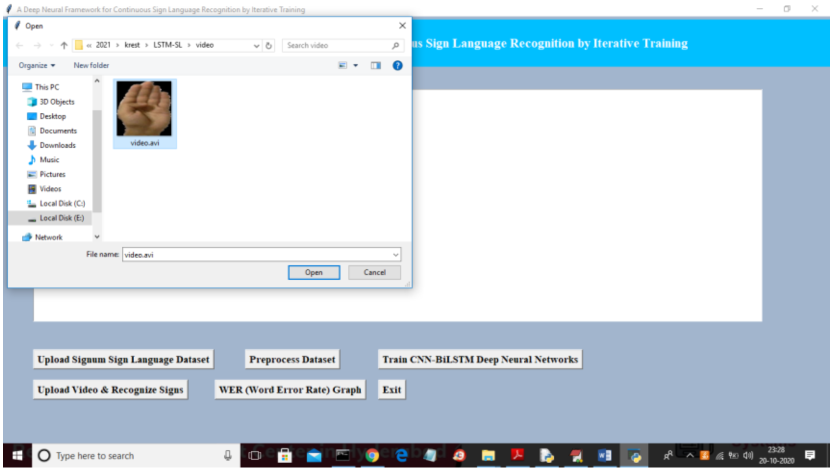
CNN BiLSTM RNN model generated. See black console to view CNN layers
CNN BiLSTM WER : 0.36805965503271465

In above screen model is generate and the detected error rate 0.36% and now we can see below black screen to see CNN layers details



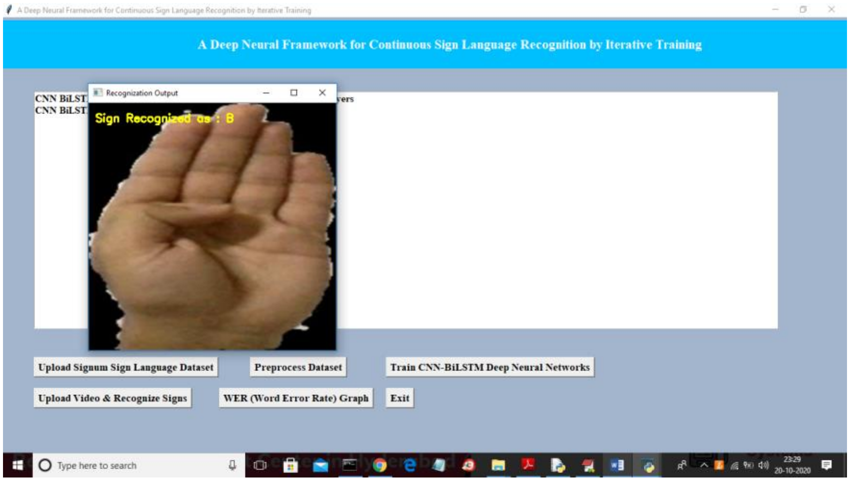
```
 deprecated. Please use tf.compat.v1.global_variables instead.

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 98, 98, 32)        896
_________________________________________________________________
activation_1 (Activation)    (None, 98, 98, 32)        0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 49, 49, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 47, 47, 32)        9248
_________________________________________________________________
activation_2 (Activation)    (None, 47, 47, 32)        0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 23, 23, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 21, 21, 64)        18496
_________________________________________________________________
activation_3 (Activation)    (None, 21, 21, 64)        0
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 10, 10, 64)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 6400)              0
_________________________________________________________________
dense_1 (Dense)              (None, 64)                409664
_________________________________________________________________
activation_4 (Activation)    (None, 64)                0
_________________________________________________________________
dropout_1 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_2 (Dense)              (None, 24)                1560
_________________________________________________________________
activation_5 (Activation)    (None, 24)                0
=================================================================
Total params: 439,864
Trainable params: 439,864
Non-trainable params: 0
_________________________________________________________________
None
```

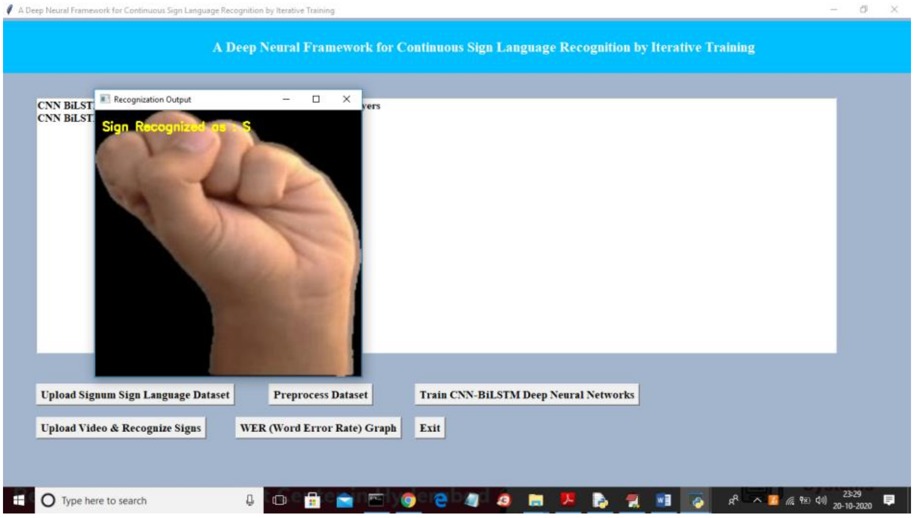
In above screen we can see CNN different layers with different size of images to filter features and to choose best features for prediction or recognition. In above screen we can see in first layer CNN has used 98 X 98 image height and width and in next layer it uses 49 X 49 and goes on. Now model is ready and now click on ‘Upload Video & Recognize Signs’ button to upload video and to identify signs

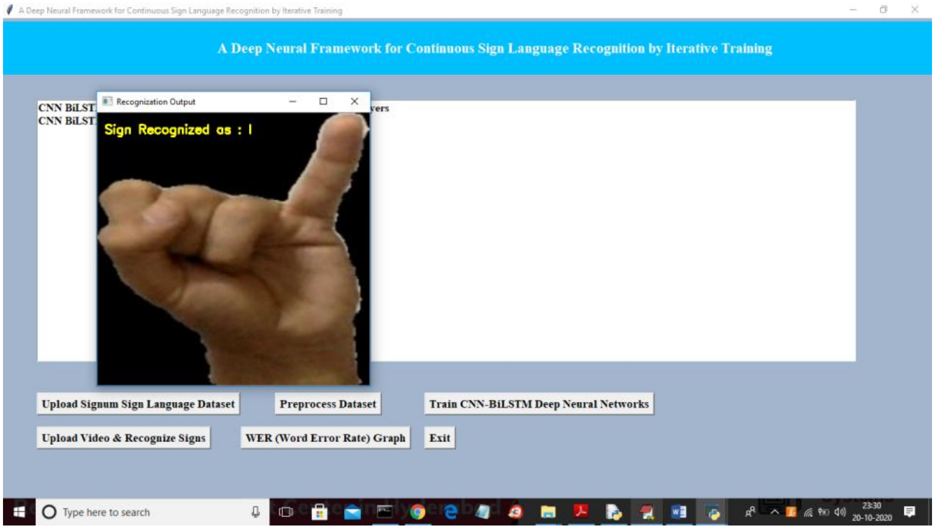

A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Open
« 2021 › krest › LSTM-SL › video
Search video
Organize ▾
New folder
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
Network
video.avi
Open
Cancel
us Sign Language Recognition by Iterative Training
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search

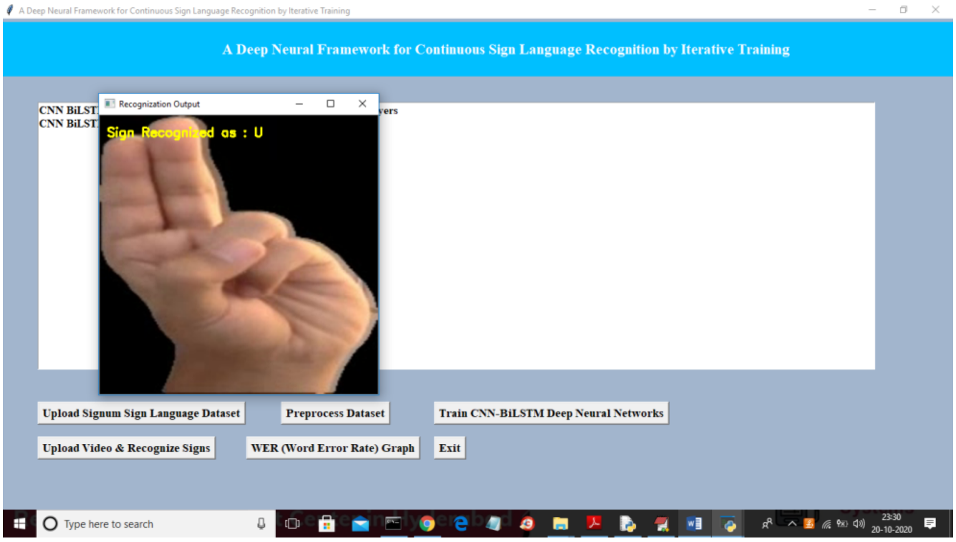
File name: video.avi

In above screen uploading 'video.avi' file and then click on 'Open' button to load video and to identify signs


Sign Recognized as : B
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Recognization Output
CNN BiLST
CNN BiLST
vers
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search
23:29
20-10-2020


Sign Recognized as : S
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Recognization Output
CNN BiLST
CNN BiLST
vers
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search
23:29
20-10-2020

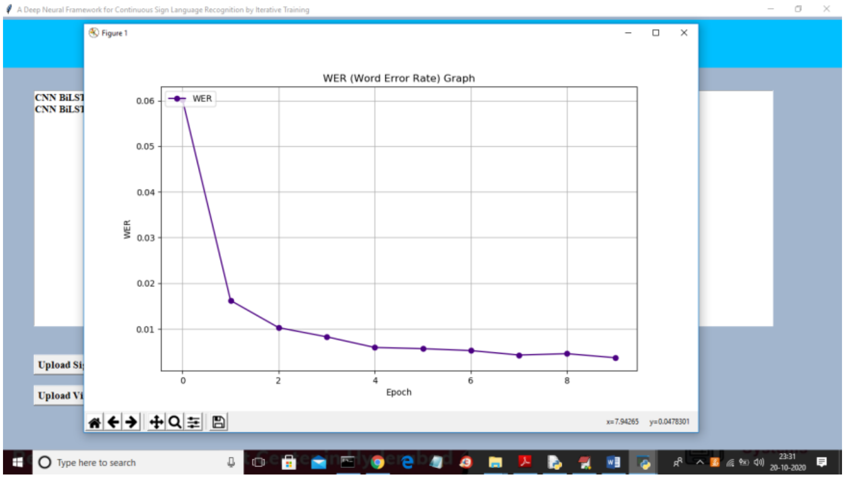

Sign Recognized as : I
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Recognization Output
CNN BiLST
CNN BiLST
vers
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search
23:30
20-10-2020


A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training
Recognization Output
Upload Signum Sign Language Dataset
Preprocess Dataset
Train CNN-BiLSTM Deep Neural Networks
Upload Video & Recognize Signs
WER (Word Error Rate) Graph
Exit
Type here to search
23:30
20-10-2020

CNN BiLST
CNN BiLST

Sign Recognized as : U

In above screens we can see while video playing CNN and BILSTM starts recognizing signs. Now click on ‘WER (Word Error Rate) Graph button to view below error rate graph



In above graph x-axis represents iterative training values and x-axis represents error rate and in above graph we can see when algorithm proceeds further with iterative training then word identifying error rate goes down.

5. CONCLUSION

In this proposal, we have developed a continuous SL recognition system with recurrent convolutional neural networks on multimodal data of RGB frames and optical flow images. In contrast to previous state-of-the-art methods, our framework employs recurrent neural networks as the sequence learning module, which shows a superior capability of learning temporal dependencies compared to HMMs. The scale of training data is the bottleneck in fully training a deep neural network of high complexity on this task. To alleviate this problem, a novel training scheme is proposed to make our feature extraction module fully exploited to learn the relevant gestural labels on video segments and keep on benefitting from the iteratively refined alignment proposals. In addition, a multimodal fusion approach also developed to integrate appearance and motion cues from SL videos, which presents better spatiotemporal representations for gestures. Further, our model is evaluated on two publicly available SL recognition benchmarks, and experimental results present the effectiveness of our method, where both the iterative training strategy and the multimodal fusion contribute to a better representation and the performance improvement

REFERENCES

[1] S. C. Ong and S. Ranganath, “Automatic sign language analysis: A survey and the future beyond lexical meaning,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 6, pp. 873–891, 2005.

[2] C. Wang, Z. Liu, and S.-C. Chan, “Superpixel-based hand gesture recognition with Kinect depth camera,” IEEE Trans. Multimedia, vol. 17, no. 1, pp. 29–39, 2015.

[3] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, “Hand gesture recognition with 3D convolutional neural networks,” in IEEE Conf. Comput. Vis. Pattern Recog. Workshops, 2015, pp. 1–7.

[4] D. Wu and L. Shao, “Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 724–731.

[5] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, “Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2016, pp. 4207–4215.

[6] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, “Sign language recognition using convolutional neural networks,” in Eur. Conf. Comput. Vis. Workshops, 2014, pp. 572–578.

[7] P. Buehler, A. Zisserman, and M. Everingham, “Learning sign language by watching TV (using weakly aligned subtitles),” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2009, pp. 2961–2968.

[8] T. Pfister, J. Charles, and A. Zisserman, “Large-scale learning of sign language by watching TV (using co-occurrences),” in Proc. Brit. Mach. Vis. Conf., 2013.

[9] J. S. Chung and A. Zisserman, “Signs in time: Encoding human motion as a temporal image,” in Eur. Conf. Comput. Vis. Workshop on Brave New Ideas for Motion Representations, 2016.

[10] Y. L. Gweth, C. Plahl, and H. Ney, “Enhanced continuous sign language recognition using pca and neural network features,” in IEEE Conf. Comput. Vis. Pattern Recog. Workshops, 2012, pp. 55–60.

[11] R. Cui, H. Liu, and C. Zhang, “Recurrent convolutional neural networks for continuous sign language recognition by staged optimization,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2017.

[12] H. Cooper, E. J. Ong, N. Pugeault, and R. Bowden, “Sign language recognition using sub-units,” J. Mach. Learning Research, vol. 13, pp. 2205–2231, 2012.