Sign Language Recognition

Yunus Can Bilge
yunuscan.bilge@gmail.com

ABSTRACT

Sign language recognition systems aim to improve connection between non-hearing and hearing people. In this study, a sign language recognition system is developed. We approached this study using Inception Convolutional Network and a single-layer LSTM model. Inception Convolutional Network is used to extract features out of the given video. The LSTM model is later used to caption the result. We have performed our experiments on UCF-101, LSA64: A Dataset for Argentinian Sign Language datasets. In order to compare our results we also run a state-of-the-art study on the same datasets and present the results.

CCS CONCEPTS

• **Computer Vision → Sign Language Recognition**; *Action Recognition*; *Gesture Recognition*; Deep Learning;

KEYWORDS

Sing Language Recognition, Action Recognition, Gesture Recognition, Deep Learning, LSTM

ACM Reference format:
Yunus Can Bilge. 1997. Sign Language Recognition. In *Proceedings of ACM Woodstock conference, El Paso, Texas USA, July 1997 (WOODSTOCK'97),* 4 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

Sign language recognition systems are very critical for people who have hearing disabilities. Non-hearing people can understand with the help of the system and integrate into rest of the community more easily. In this work we would like to make an introduction and start by recognizing signs in each videos. Therefore our main approach is learning to recognize signs in videos. Automation of the recognizing sign languages is a research problem which has a long history.

There are many variants of the sign languages which means that there is no standardization. Most of the countries have their own sign languages. In our study we started with using Argentinian Sign language dataset[1].

Sign language users mostly combine hand movements(depth), hand shapes, arms, body, and facial expressions[2]. In our approach we assumed that sign languages only include hand shapes. On the other hand sign languages have different grammatical structure than spoken languages[2]. Since we only concern hand shapes grammatical differences do not affect our approach.

There are some related study areas which are gesture recognition, action/activity recognition. Gesture recognition tasks mostly aim to recognize hand and face gestures. Therefore gesture recognition researchers try to model human body language[3]. Activity recognition can be defined as analyzing the activities in videos[4]. According to [4] the difference between gestures and action/activity is that; gestures which are elementary movements that describe motion such as raising a hand, while action recognition deals with one person activites such as; waving. These activities can be composed of many gestures.

In this study we use Google's Inception Architecture to extract features out of the videos[5]. There are many other architectures exist for feature extraction such as LeNet-1998 [9], AlexNet-2012 [10], ZFNet-2013 [11], VGG-2014 [12], GoogLeNet-2014 [13], ResNet-2015 [14].The basic idea of Inception architecture is that you do not need to know which convolution features map is better to apply or test for. After that LSTM(Long Short-Term Memory) model is used to declare the sign of the given video[6]. LSTM is a kind of Recurrent Neural Network architecture. The main difference is that LSTM units include memory cell which maintains information for a long period of time. In our approach we combine a Convolutional Neural Network with a Recurrent Neural Network.

We have done our experiments on UCF-101[7] and Argentinian Sign language datasets[1]. In order to compare our method we use Temporal Segment Networks study[8]. Our results show that the approach is correct but it needs to be improved.

2 RELATED WORK

2.1 Recurrent Models of Visual Attention[18] - 2014

In this study instead of doing computationally expensive tasks on large images via CNNs they select sequence of regions and process only on that locations. The results of the study outperforms CNN baseline on images.

2.2 Multi-Scale Deep Learning for Gesture Detection[22] - 2014

Researchers use a deep learning approach based on multi-scale and multi-modal approaches. The researchers also integrated intensity, depth and pose information. Their deep convolutional multi modal architecture operates 3 temporal scales. In each of these scales independent learning is happening. Predictions of different scales are then combined using late fusion. Multi-scale deep neural network is composed of CNN, pose descriptor and meta-classifiers.

2.3 Beyond Temporal Pooling[19] - 2015

"Beyond Temporal Pooling: Recurrence and Temporal Convolutions for Gesture Recognition in Video" study explores deep architectures

WOODSTOCK'97, July 1997, El Paso, Texas USA

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

for gesture recognition in videos. They propose a new end-to-end trainable neural network architecture which includes temporal convolutions and bidirectional recurrence. The models that they tested are Single frame CNN architecture, Temporal Feature Pooling(Many to one), Bidirectional RNN(many to many), Adding temporal convolutions(many to one), and temporal convolutions and RNN. Researchers show that RNNs are very import and when they add temporal convolutions to it performance increases.

2.4 Hand Gesture Recognition using 3D CNN[20] - 2015

The study proposes a 3D-CNN achitecture which extracts discriminative spatio temporal features from raw video. They use multi spatial scales for the final prediction. They actively use color, depth, and trajectory information. The CNN based classifier includes two sub networks; high resolution network and low resolution network. The outputs of the networkk consist of class membership probabilities which later fused to make final prediction.

2.5 Deep learning of mouth shapes for sign language[21] - 2015 :

Researchers model mouth shapes using deep CNNs. They claim that mouth shapes are very import as most people look at face during sign language communication. The method is weakly supervised and they present a way to add neural network classifier outputs into HMM. Researchers learn mouth shape via CNNs and find the most likely frame model state alignment in HMM.

2.6 Show, Attend, and Tell[16] - 2015

Researchers aim to generate a descriptive sentence for an image. They use attention mechanism which learns to select regions while generating a description in their study. In their work they use convolutional neural networks, recurrent neural networks, and attention mechanisms. They mainly use CNNs to encode the given image, attention mechanisms to select a region of the image, rnn to generate words with respect to output of the attention mechanism.

2.7 Long-term Recurrent Convolutional Networks for Visual Recognition and Description[15] - 2015

This research is suitable for image captioning, activity recognition, and video description. The study is about a class of end to end trainable RNNs. The difference of the study from previous studies is that this study is both temporally and spatially deep. The model combines a deep CNN architecture with a model that learns to recognize temporal dynamics for a sequence. Basically a sequence of visual input is given to CNN to extract features. The outputs are fed into stack of LSTM models which provides the output. In order to represent long sequences the weights of LSTM and CNN are shared.

2.8 Deep Sign[17] - 2016

Researchers embed a CNN into a HMM model. They use CNN to extract strong features and they use HMM to model sequences. Another contribution is that they treat CNN outputs as true Bayesian posteriors. Another idea is that generally CNNs are trained on the frame level. In sign language datasets there is no frame level labeling. Therefore they focus on approaches that deal with variable length inputs and outputs. In order to find best fitting sequence of words to the video they use Bayesian decision rule and they try to maximize class posterior probability.

2.9 Temporal Segment Networks[8] - 2016

The study is about discovering the principles to effectively design a CNN architecture which is very effective on visual recognition tasks. Their main contribution is about modeling long range temporal structure. The study has a very good performance on UCF-101 dataset.

3 METHOD

3.1 CNN Model

A Convolutional Neural Network(CNN) in general consists of convolutional layers, pooling layers, and fully connected layers. Convolutional layers are a kind of filters which has a receptive field on the input field. The receptive field extends through the depth. Pooling layers aim to reduce the spatial size. Using pooling layers also help to prevent overfitting. Fully connected layers have full connections with previous layers' activations. CNNs are easier to train than fully connected networks due to the parameter size.

There are many examples of CNNs which are; LeNet-1990, AlexNet-2012, ZF-Net-2013, GoogLeNet-2014, VGGNet-2014, ResNet-2015. In our approach, we used Inception-V3[5] model which is the next version of GoogLeNet.

The inception module solves the problem that what type of convolution use at each layer. The model performs each of them in paralel and concatanates at the end. Another advantage of this model is it both preserves local and high level features using different convolutions.

Inception network has very good accuracy results on classification[5]. In this study it is aimed to use this network to strengthen learning via its features. In general the extracted features from videos are fed into RNN which is explained in the next section. Therefore first of all videos are divided into frames. Each of these frames are fed into Inception network. The first difference is that the classification layer is removed therefore we save the features through last pooling layer. The extracted features are fed into Recurrent Neural Network.

3.2 LSTM Model

Recurrent Neural Networks use sequential data as opposed to other models which assumes that inputs are independent. RNNs perform the same task same task for all of the elements of the sequence and the relevent output is depend on the history. RNN models work on vector sequences such as; one to one(image classification), one to many(image captioning), many to one(sentiment analysis), many to many(machine translation), and many to many(video classification)[24]. It is safe to say that RNN models have some kind of a memory. In practice RNNs cannot use information in long sequences.

In order to solve problem of capturing information for long sequences there are LSTM models which are a kind of RNN. LSTM

model has been around for a long time[6]. LSTM(Long Short Term Memory) models learn long term dependencies. LSTM models have a cell state which goes through very small changes through the network. Therefore information is kept during this process. The model also has gates to control the cell state such as; forget, input, and output gate. All of these gates are using logistic functions. The forget gate controls what extend of information will remain on cell. Input gate controls what extend of new information will flow into memory. Output gate controls the extend of information for computing output activation. There are variants of LSTM. In this study we use LSTM model due to the well known fact of capturing information for long sequences.

In this approach we first convert the features that extracted through Inception into sequences of features. We use a single LSTM layer. In tasks like machine translation we might need multi-layer models to learn more complex conditions[26]. However in our case we use a single layer which seems to outperm stacked versions[23]. We also add Flatten, Dense, Dropout, and a Dense layer again afterwards. The output returns the recognition prediction.

3.3 The Combined CNN + RNN Model

Combination of CNN and RNN model is a hybrid model. In this model we are extracting features using Inception network and save it. Later that we train LSTM model by feeding the sequences(features) that saved. In the end the prediction is given. In this model both spatial and temporal information are used.

4 EXPERIMENTS

4.1 Datasets

(1) **UCF-101[7]:** The action recognition dataset which contains 101 action categories. The actions include large variations. The main advantage of the dataset is actions are realistic. The action categories can be divided into 5 main types; Human-Object Interaction, Body-Motion Only, Human-Human Interaction, Playing Musical Instruments, Sports. The dataset includes total of 6 GB data. State of the art accuracy on this dataset is 0.94.

(2) **LSA-64[1] :** The automatic sign recognition dataset includes sign samples for Argentinian Sign language. The dataset contains total of 3200 videos and 64 signs. The selected signs are the most common ones and the signs both correspond to verbs and nouns. In our case we started with using 5 main sub-signs which are; Opaque, Red, Green, Yellow, and Bright. We also use one-handed signs in order to handle and analyze the method more easily. The whole dataset includes 1.9 GB of data. The samples from the dataset can be seen in Figure 1.

Figure 1: LSA-64 samples

4.2 Results

First of all we try the model with different parameters for dropout, weight regularization, early stopping, and learning rate. In the last layer our activation function is softmax. We test different different values for dropout and as a result 0.5 selected. We use RELU activation with dense layer. We set batch size 32 and epoch number is selected as 1000. In inception V3 structure we use imagenet weights.

Moreover we compare our CNN+LSTM model with CNN only model. In CNN only(Inception V3 pretrained on ImageNet) model we fine tune it to work on the new dataset. The top layer is fine tuned.

The selected evaluation metric is Accuracy. Accuracy is calculated as 1 - Error_Rate. Error_Rate is calculated as; number of misclassification/number of samples in validation set. For now we don't calculate F1 score and ROC curve. It will be calculated when we move on to a bigger sign language recognition dataset. As a validation k-fold cross validation is used. In 10-fold cross validation the dataset randomly partitioned into 10 equal sizes. 9 partitions are used for training and 1 partition used for testing and the validation process performed 10 times.

UCF-101 Dataset

As we declared we first run the models on UCF-101 dataset. Top-1 accuracy for CNN only model was 62%. On the other hand our CNN+LSTM model has 73% top-1 accuracy. When we look at the Temporal Segment approach the accuracy is 94%.

LSA-64 Dataset

In this dataset CNN+LSTM model has 95% accuracy. The CNN only method has made 70% accuracy. Due to technical difficulties we couldn't run the Temporal Segment Networks method on this dataset. Therefore it is left as a future work.

5 CONCLUSIONS AND DISCUSSIONS

In this study, we approach the recognition problem using a model which is combined of a CNN and a LSTM architecture. We analysed the performance of this hybrid model on both UCF-101 and a sign language dataset. We compare the hybrid models result with CNN only model and Temporal Segment Network model.Our results

on different datasets show that we need to improve the current architecture. However this introductory study shows that the proposed approach can be used in many recognition tasks. The model performs well on LSA-64 dataset however we need to improve the approach in order to get more accuracy on UCF-101 dataset. In the future ResNets can be used for feature extraction. ResNets are also successful on classification, detection, and recognition tasks. Moreover different RNN architectures can be used such as GRA(Gated Representation Allignment). PCA can also be used after CNN in order to reduce dimensionality of features. Optical-flow images should be used in this application in the future. In overall approach all videos are subsampled into 40 frames. We should use all frames in the future. The last one is that we could do preprocessing for videos such as subtracting mean from each video which keeps the frames more definite.

6 REFERENCES

[1] Ronchetti, F., Quiroga, F., Estrebou, C. A., Lanzarini, L. C., & Rosete, A. (2016). LSA64: An Argentinian Sign Language Dataset. In XXII Congreso Argentino de Ciencias de la ComputaciÃşn (CACIC 2016).

[2] Baker, Anne, Beppie van den Bogaerde, Roland Pfau, and Trude Schermer eds., 2016. The Linguistics of Sign Languages: An introduction. John Benjamins Publishing Company

[3] Ying Wu and Thomas S. Huang, "Vision-Based Gesture Recognition: A Review", In: Gesture-Based Communication in Human-Computer Interaction, Volume 1739 of Springer Lecture Notes in Computer Science, pages 103-115, 1999, ISBN

[4] Aggarwal, J. K., & Ryoo, M. S. (2011). Human activity analysis: A review. ACM Computing Surveys (CSUR), 43(3), 16.

[5] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2818-2826).

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

[7] Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402.

[8] Temporal Segment Networks: Towards Good Practices for Deep Action Recognition, Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, & Luc Van Gool, ECCV 2016, Amsterdam, Netherlands.

[9] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[10] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[11] Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer International Publishing.

[12] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[13] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-9).

[14] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778).

[15] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2625-2634).

[16] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. In International Conference on Machine Learning (pp. 2048-2057).

[17] Koller, O., Zargaran, O., Ney, H., & Bowden, R. (2016). Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition. In Proceedings of the British Machine Vision Conference 2016.

[18] Mnih, V., Heess, N., & Graves, A. (2014). Recurrent models of visual attention. In Advances in neural information processing systems (pp. 2204-2212).

[19] Pigou, L., Van Den Oord, A., Dieleman, S., Van Herreweghe, M., & Dambre, J. (2015). Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. International Journal of Computer Vision, 1-10.

[20] Molchanov, P., Gupta, S., Kim, K., & Kautz, J. (2015). Hand gesture recognition with 3D convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 1-7).

[21] Koller, O., Ney, H., & Bowden, R. (2015). Deep learning of mouth shapes for sign language. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 85-91).

[22] Neverova, N., Wolf, C., Taylor, G. W., & Nebout, F. (2014, September). Multi-scale deep learning for gesture detection and localization. In Workshop at the European Conference on Computer Vision (pp. 474-490). Springer International Publishing.

[23] Five Classification Methods. (2017, June 3). Retrieved from https://hackernoon.com/five-video-classification-methods-implemented-in-keras-and-tensorflow-99cad29cc0b5

[24] Convolutional Neural Networks for Visual Recognition. (2017, June 3). Retrieved from http://cs231n.github.io/

[25] Recurrent Neural Networks. (2017, June 3). Retrieved from http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

[26] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.