# Real Time Bangladeshi Sign Language Detection using Faster R-CNN

Oishee Bintey Hoque*, Mohammad Imrul Jubair†, Md. Saiful Islam‡, Al-Farabi Akash§, Alvin Sachie Paulson¶
Department of Computer Science and Engineering,
Ahsanullah University of Science and Technology Dhaka, Bangladesh
{*bintu3003, ‡saiful.somum, §alfa.farabi, ¶sachiekhan}@gmail.com, †mohammadimrul.jubair@ucalgary.ca

***Abstract*—Bangladeshi Sign Language (BdSL) is a commonly used medium of communication for the hearing-impaired people in Bangladesh. Developing a real time system to detect these signs from images is a great challenge. In this paper, we present a technique to detect BdSL from images that performs in real time. Our method uses Convolutional Neural Network based object detection technique to detect the presence of signs in the image region and to recognize its class. For this purpose, we adopted Faster Region-based Convolutional Network approach and developed a dataset – *BdSLImset* – to train our system. Previous research works in detecting BdSL generally depend on external devices while most of the other vision-based techniques do not perform efficiently in real time. Our approach, however, is free from such limitations and the experimental results demonstrate that the proposed method successfully identifies and recognizes Bangladeshi signs in real time.**

***Index Terms*—Bangladeshi Sign Language, Convolutional Neural Network, Faster R-CNN**

## I. Introduction

The field of computer vision is reaching every possible sectors to help human being. In recent days, it is being used to assist the deaf community by facilitating the *sign language detection technique*. In order to liaise with other people, deaf persons widely use Sign languages as the mediums of communication (Fig. 1). Hence, the main contribution of the sign language detection technique is to act as a digital interpreter between the deaf and the hearing people. There are detection strategies where supplementary equipment such as specialized gloves [1], Kinect [2] etc. are used, however, the inputs are not 2D images and the system is much complicated. Most of the other techniques take image (or sequence of images) containing signs as input using camera, and the ultimate step is to detect those signs and present them in a meaningful manner [3]. The later approaches do not depend on any additional devices and the image processing techniques are applied on the input image which locates the position of the gesture for detection. Previous works were solely based on old-fashioned image pre-processing methods such as – morphological operations, color-based foreground segmentation etc. In this era of machine learning, Convolutional neural network provides us more powerful tools for object detection which has surpassed the former approaches by all means. In our work, we explore the area of sign language detection and develop a technique to detect signs in real time by exploiting the *Faster Region-based Convolutional Network* method (faster R-CNN) [4]. Different sign languages are used worldwide,

Fig. 1: Examples of Sign Language. Here different signs from BdSL are shown.

such as – American Sign Language (ASL), Chinese Sign Language (CSL), Parisian Sign Language, etc. ASL recognition has been explored since around 1995 [5] and other languages are also investigated by different researchers [6] [7]. However, a very few work has been done on Bangladeshi Sign Languages (BdSL). Most of the previous works does not take the advantages of convolutional neural network based object detection technique to identify the gestures. In this paper, we investigate BdSL recognition and develop a system using faster R-CNN. For this purpose, we collected images of specific signs with different backgrounds, variations and built a dataset – ***BdSLImset*** (***Bangladeshi Sign Language Image Dataset***) – to train our system. In that case, we do not need to pre-process the images to differentiate the foreground. Our results exhibit satisfactory outcome in identifying the gesture area and recognizing BdSL in real-time.

In summary, the contributions of our work are listed below.

- We generated a dataset called ***BdSLImset*** containing images of Bangladeshi signs with random backgrounds and lighting conditions. The dataset is available here: `https://github.com/imruljubair/bdslimset`.
- We propose a recognition technique to identify the signs and detect BdSL from images in real time. Our system is trained on our ***BdSLImset*** dataset and uses *Faster R-CNN* object detection approach. We have demonstrated the experimental outcomes of our proposed methodology.

The remainder of the paper is organized as follows. We reviewed on previous BdSL recognition methods and datasets followed by an overview of Faster R-CNN in Section II. We present our dataset and pipeline of the proposed technique in Section III. Section IV illustrates our experimental results and

978-1-5386-5229-9/18/$31.00 ©2018 IEEE

comparisons with previous works. Possible avenues of future exploration are discussed in Section V.

## II. Background and Related Works

In the first part of this section, we discuss about previous works on the domain of Bangladeshi sign language detection followed by a study on the existing datasets. The last section provides a brief review on the Faster R-CNN [4] which is used in our proposed methodology.

### A. Existing BdSL Detection Techniques

Rahman, Fatema and Rokonuzzaman used gloves containing dots at each finger position to track the action of signs. The authors collected the dots and mapped the results of the clustered dots to predefined charts. The system can only detect the sign of Bengali numerals from 1 to 10 [8]. In [9], researcher applied image processing operations on input images with any assistance of gloves. They determined relative finger tip positions from image and trained an artificial neural network using those tip-position vectors. In their case, the recognition was not in real time and the authors claimed to have an accuracy of 98.99 percent for Bangla sign letters. In [10], the authors introduced a computer vision-based Bengali sign words recognition system which used contour analysis and *Haar-like* feature based cascaded classifier. They trained their classifier and tested the system using 3600 ($36 \times 10 \times 10$) contour templates for 36 Bengali sign letter separately and achieved 96.46 percent recognition accuracy. In 2017, Yasir, Prasad, Alsadoon and Sreedharan made use of virtual reality by applying leap motion controller to capture hand gestures and implemented CNN for detecting the signs [1] . However, their recognition was not in real time. In [11], the authors also introduced an approach for detecting BdSL letters and digits which applies a fuzzy-logic based model and grid-pattern analysis in real-time. In [12], the authors presented a real-time Bengali and Chinese numeral signs recognition system using contour matching. The system is trained and tested using a total of 2000 contour templates separately for both Bengali and Chinese numeral signs from 10 signers and achieved recognition accuracy of 95.80% and 95.90% with computational cost of 8.023 milliseconds per frame. In [13], the authors introduced a method of recognizing Hand-Sign-Spelled Bangla language. The system is divided into two phase – hand sign classification and automatic recognition of hand-sign-spelled for BdSL using Bangla Language Modeling Algorithm (BLMA). The system is tested for BLMA using words, composite numerals and sentences in BdSL achieving mean accuracy of 93.50%, 95.50% and 90.50% respectively.

The existing BdSL recognition techniques are not trained with same datasets. Different models are trained on author's individual datasets which were not available for further exploration. We're going to briefly discuss about the existing BdSL datasets in the next subsection.

### B. Existing BdSL Datasets

Most of the existing datasets used in BdSL detection models do not have variation in background and different lighting condition. Because of these limitations, those datasets are not suitable to be fed into the CNN model. A comparison between others datasets which were used in previous works and our dataset is presented in Table I in later section.

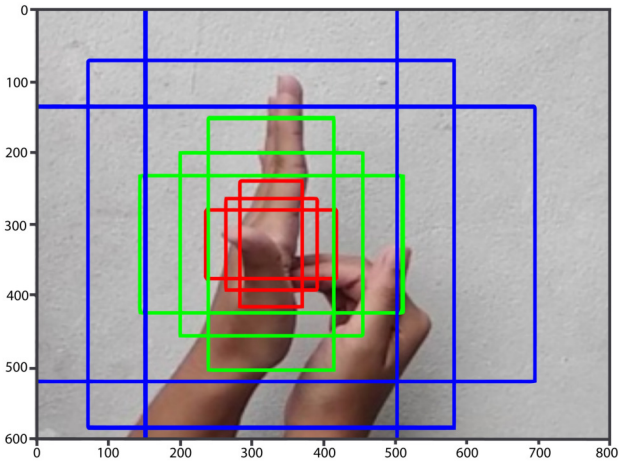


Fig. 2: A conceptual illustration of anchors in CNN feature map for 3 different aspect ratios and sizes. Here, three colors represent three scales or sizes: $128 \times 128$, $256 \times 256$, $512 \times 512$, and three boxes have height width ratios $1:1$, $1:2$ and $2:1$ respectively.

### C. Faster-RCNN model

In our work, we emphasize on identifying signs in the image. This challenge basically falls under the domain of object detection. Our study on CNN based object detection leads us to focus on using Faster R-CNN. Faster R-CNN is an improved version of its antecedent algorithms – R-CNN [14] and Fast R-CNN [15]. Unlike other models, Faster R-CNN feeds only necessary region to the convolutional neural network. Initially, CNN generates a feature map and a network – Regional Proposal Network (RPN) – proposes regions with high probability of containing desired object. In the network architecture, it applies a *region of interest (RoI)* pooling layer and reshape them into a fixed size to feed into a fully connected layer. From the RoI feature vector, it uses a softmax layer to predict the class of the proposed region. A key factor that plays an important role in Faster R-CNN is the *Anchor*. Anchors are fixed sized bounding box. An input image is divided into several anchors in CNN feature map. In proposed model, there are 4 different scale $(0.25, 0.5, 1.0, 2.0)$ and 3 aspect ratios $(0.5, 1.0, 2.0)$ with height and width stride of 16 anchors. Fig. 2 shows an example illustration of anchors. Therefore, if an image has $w \times h$ ratio and we choose every stride of 16, there will be $(w/16) \times (h/16)$ positions to consider and it will finally have $(w/16) \times (h/16) \times 12$ anchors for each image. The RPN takes all the reference boxes from the feature map and generate a good set of proposals of being objects. The process votes the anchor if it is an object or not. The Anchor is labelled as foreground if it is voted as object,

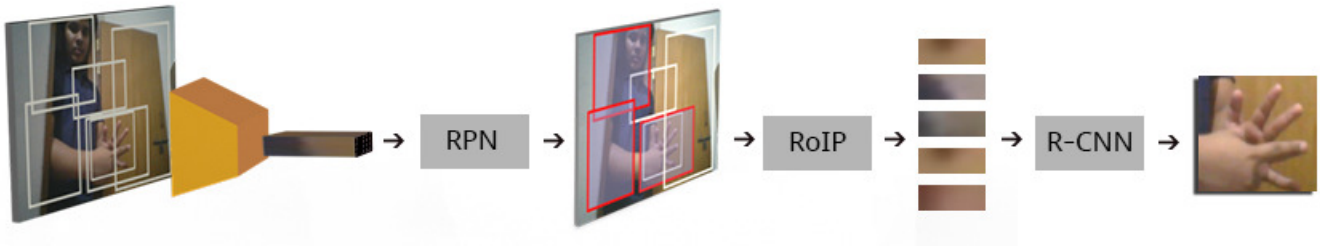


Fig. 3: Detailed architecture of the network. Firstly the input image goes into CNN framework and creates a feature map. RPN proposes anchors with higher probability of being an anchor and the RoI pooling classification is performed at last to finalize the detection.

otherwise labelled as background. Later, the anchors based on the similar criteria are selected and refined. Anchors labelled as background is not included in the regression.

After the RPN stage, we get proposed regions with different sizes. RoI pooling reduce the feature maps into the same size. It eventually splits the input feature map into a fixed number of roughly equal regions then applies Max-Pooling on every region. Hence the output of RoI Pooling is always fixed regardless the size of input.

## III. Proposed Methodology

In this section, we first present our dataset followed by the illustration of our proposed technique.

### A. *BdSLImset (Bangladeshi Sign Language Image Dataset)*

After the investigations on existing works mentioned in the previous section, we found lacking of a proper dataset to integrate a neural network architecture. As we want to develop a real time system, our model must be enriched to train such robust classifier, training images should have variations in signs of letter, in backgrounds and lighting conditions. Therefore, we kept these factors in our considerations while collecting training images for this dataset. In our dataset, some images contain desired gesture of letter which is partially obscured, overlapped with something else, or only halfway in the picture. Each image size is less than $200kb$ and the resolution is not more than $700 \times 1280$. Fig. 4 shows some sample dataset images. Currently, our *BdSLImset* dataset has 10 different labels sign letters. We collected about 100 pictures of each gesture. For each letter about 100 sign images of 10 persons of different ages and genders have been captured with variety of backgrounds. The dataset is divided into training set and testing set with the ratio of 8:2. After gathering images, we selected the region of each of the hand gestures with a bounding box and labeled them (see Fig. 5). Thus the initial training data is prepared. A comparison between ours and previous dataset has been shown in TABLE I.

TABLE I: Comparisons between datasets used in previous work and in our method.

| Related Works | Background & Lighting | Image per Class × Total Classes | No. of Signers |
|---|---|---|---|
| Rahman et al. [3] | Static | $36 \times 10$ | 10 |
| Rahman et al. [12] | Static | $10 \times 10$ | 10 |
| Ahmed et al. [9] | Static | $37 \times 14$ | 3 |
| Yasir et al. [1] | *N/A* | *N/A* | *N/A* |
| Our (BdSLImset) | Randomized | $10 \times 10$ | 10 |

### B. Pipeline of the Proposed Technique

Our proposed methodology follows a certain pipeline. At first, our training data is fed into the convolutional neural network as mentioned earlier in the overview of Faster R-CNN and eventually makes a feature map based on given aspect ratio and sizes of the model. Then the feature map goes to the RPN and proposes regions with probability of that region being a sign gesture of a letter. Fig. 3 shows the entire working process of the network. Here, RoI pooling resizes the feature map into fixed sizes for each proposal in order to classify them into a fixed number of classes.

## IV. Experiments and Results

This section represents the experimental results of our proposed BdSL recognition with Faster R-CNN model on the prepared *BdSLImset* dataset. Firstly, we start with experimental setup and then the final outcome and comparisons with other models are presented.

### A. Experimental Setup

To employ Faster-RCNN model based training in our module, each image were re-sized and their resolution were kept minimal for training as mentioned earlier. The image set were divided into testing and training set. Training set contains 80 percent of the images and test set contains 20 percent.

Our technique is implemented in `Tensorflow-GPU V1.5` and `cuda V9.0`. The training was performed by adopting the `Faster RCNN Inception V2` model. The experiment has been conducted on a machine having CPU from Intel ®. Core™ i7-7500U of 2.7 GHz, GPU Nvidia 940mx with 4.00GB and with 8.00GB memory on a Windows 10 operating system.

Fig. 4: Sample sign language images from our *BdSLImset* dataset for different Bengali letters with different background.

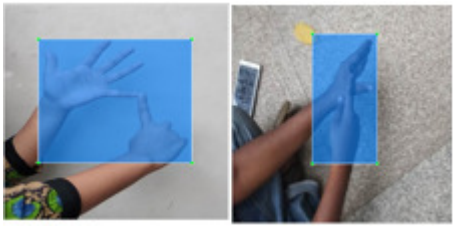

Fig. 5: Images with labels of different signs (labelled inside the blue boxes).

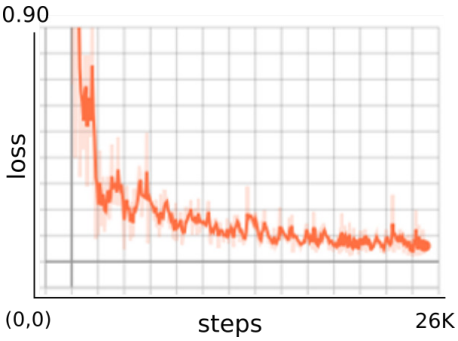


Fig. 6: Loss graph generated from *tensor board*. Initially the loss was about 3.0 and quickly dropped to below 0.8 and the training was stop after 26k iterations at loss of 0.075.

### B. Experimental Result

Training was completed with loss of 0.07538 and accuracy was about 98.2 percent in average. The detection time was really fast of about 90.03 milliseconds. The training included about 100 images for each label. As sign letter gesture have similarities among them and take wide range of area of hand and may have different background, sometimes it becomes difficult for the model to differentiate between background and foreground of the image. The experimental process had been conducted for different background and persons, and the recognition time was stable for every test but confidence rate varies in different lighting conditions. Fig. 7 shows some result of our testing process.

Faster R-CNN model takes hours to train the sets of image data. We stopped the training when the loss rate became almost constant to 0.07538. It took almost 26K iterations to decrease loss to this rate (see Fig 6). Each step of training reports the loss. It started high and gets lower and lower as training progresses. For our training model, it started at about 3.0 and quickly dropped below 0.9.

Table II presents the comparisons between proposed and other BdSL detection methods. This comparison is based on their individual dataset and experiment. It is obvious from the table that our method is efficient than the others while considering the performance in real time as well as the accuracy rate.

During the testing process, one letter 'ই' was taking more time to be recognized and was giving faulty outcome. The reasons can be due to extreme variation with the background and illumination. The sign letters have much similarities among them which make it complicated to differentiate between these letters. Such as, 'ই' having very similar gesture as 'এ' which may lead the classifier to a wrong decision. In order to overcome these issues, number of samples in our dataset needs to be enriched more. Fig. 8 shows some failed situations during the recognition of 'ই'.

TABLE II: Comparisons between our method and other works

| Related Works | Methodology | Recognition Time | Accuracy |
|---|---|---|---|
| Rahman et al. [3] | Haar-like feature | Real Time (93.55 ms) | 96.46% |
| Ahmed and Akhand [9] | ANN | Not Real Time | 98.99% |
| Yasir et al. [1] | CNN | Not Real Time | 97.00% |
| Rahman et al. [12] | Contour Matching | Real Time | 95.80% |
| Proposed Methodology | Faster R-CNN | Real Time (90.03 ms) | 98.20% |

## V. Conclusion and Future Work

In this paper, we have developed a system that would recognize Bangla Sign Letters in real time. Images of different BdSL signs from our *BdSLImset* dataset were trained by Faster R-CNN based model to solve the problem of sign language recognition. We obtained average accuracy rate of 98.2 percent


Fig. 7: Some results of real time detection 'এ','গ','ঊ' & 'ত' using our proposed technique. This experiment had been done on various situation, i.e. different background, illumination and angles. In (a), 'এ' has detection confidence rates of 98% and 86% (*top & bottom row in left respectively*). For each row, 1st column shows the detection and the (2nd column) has a zoom version of detected portion for better investigation. A reference sign is included at the rightmost column as ground truth collected from *BdSLImSet*. Similarly, (b) shows that 'গ' has detection confidence rates of 90% and 89% (*top & bottom*). In (c) and (d), 'ঊ' has 94% & 99% (*top & bottom*), and 'ত' has 90% & 90% confidence rate (*top & bottom*) respectively.

and recognition time was 90.03 milliseconds. Different possible avenues of future exploration of our research are discussed below.

- Our system has limitations while recognizing the letters, which have many similarities among their pattern. The problem might be overcome with more image data for those letters.
- Also the image size is a factor, as data training requires huge amount of time. Our research is still ongoing progress. We are collecting more data and training the system to recognize the patterns better.
- In future, our plan is to evaluate our model by genuine users to sort out its limitations and improve the system. This will also help us see how the system reacts on real life situation and how clearly it can recognize the pattern and interpret effectively.
- One of our targets is to make a system that recognizes a sequence of signs, concatenate them and translate it into a phrase. In that case, not only a single frame will be considered, but a series of actions, providing a meaningful word or phrase. For this purpose, we are planning to apply recurrent neural network [16] in future. This attempt will also require our dataset to be refined and processed further.

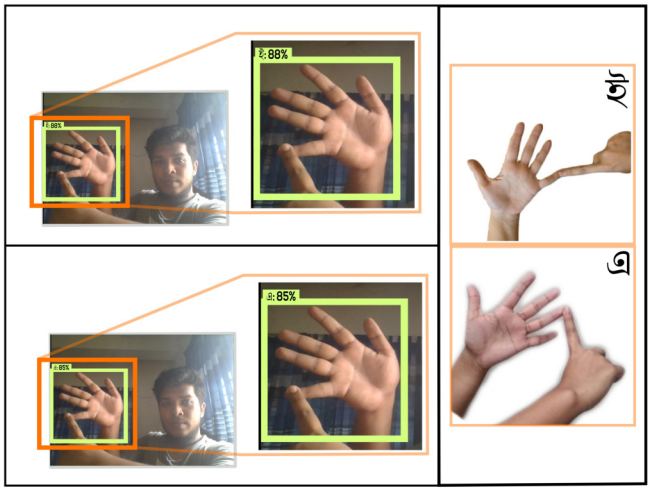

Fig. 8: False recognition because of the similarity in the gestures of 'এ' and 'ই'. Here, for both rows, the 1st column shows the detection and the 2nd column has a zoom version of detected portion for better investigation. Both the rows show the same signs of 'ই' while our approach performs perfectly in once case (*top row in left*), while fails in another situation (*bottom row*). A reference sign is included at the rightmost columns as ground truth.

## REFERENCES


[1] F. Yasir, P. Prasad, A. Alsadoon, A. Elchouemi, and S. Sreedharan, "Bangla sign language recognition using convolutional neural network," in *Proceedings of the 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies*. United States: IEEE, Institute of Electrical and Electronics Engineers, 4 2018, pp. 49–53.

[2] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *Computer Vision - ECCV 2014 Workshops*, L. Agapito, M. M. Bronstein, and C. Rother, Eds. Cham: Springer International Publishing, 2015, pp. 572–578.

[3] M. A. Rahaman, M. Jasim, M. H. Ali, and M. Hasanuzzaman, "Real-time computer vision-based bengali sign language recognition," in *2014 17th International Conference on Computer and Information Technology (ICCIT)*, Dec 2014, pp. 192–197.

[4] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," vol. abs/1506.01497, 2015.

[5] M. B. Waldron and S. Kim, "Isolated asl sign recognition system for deaf persons," vol. 3, no. 3, Sept 1995, pp. 261–271.

[6] A. Karami, B. Zanj, and A. K. Sarkaleh, "Persian sign language (psl) recognition using wavelet transform and neural networks," vol. 38, no. 3. Tarrytown, NY, USA: Pergamon Press, Inc., Mar. 2011, pp. 2661–2667.

[7] C. Wang, W. Gao, and Z. Xuan, "A real-time large vocabulary continuous recognition system for chinese sign language," in *Advances in Multimedia Information Processing — PCM 2001*, H.-Y. Shum, M. Liao, and S.-F. Chang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 150–157.

[8] S. Rahman, N. Fatema, and M. Rokonuzzaman, "Intelligent assistants for speech impaired people," in *Intl. Conf. on Computer and Information Technology*. Dhaka, Bangladesh: East West University, 2002.

[9] S. T. Ahmed and M. A. H. Akhand, "Bangladeshi sign language recognition using fingertip position," in *2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec)*, Dec 2016, pp. 1–5.

[10] M. A. Rahaman, M. Jasim, M. H. Ali, and M. Hasanuzzaman, "Computer vision based bengali sign words recognition using contour analysis," in *2015 18th International Conference on Computer and Information Technology (ICCIT)*, Dec 2015, pp. 335–340.

[11] M. A. RAHAMAN, M. JASIM, M. ALI, T. ZHANG, and M. HASANUZZAMAN, "A real-time hand-signs segmentation and classification system using fuzzy rule based rgb model and grid-pattern analysis."

[12] M. A. Rahaman, M. Jasim, T. Zhang, M. H. Ali, and M. Hasanuzzaman, "Real-time bengali and chinese numeral signs recognition using contour matching," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Dec 2015, pp. 1215–1220.

[13] M. A. Rahaman, M. Jasim, M. H. Ali, and M. Hasanuzzaman, "Bangla language modeling algorithm for automatic recognition of hand-sign-spelled bangla sign language." Front. Comput. Sci., 2018, p. 0.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 580–587.

[15] R. Girshick, "Fast r-cnn," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 1440–1448.

[16] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *INTERSPEECH*, 2010.