Handwritten Text Recognition

Shivam Raval (201501088), Om Thakkar (201501109), Anushree Rankawat (201501007), Jay Mohta (201501036)
School of Engineering and Applied Science, Ahmedabad University

***Abstract*—Text Recognition in handwritten texts is a challenging problem owing to the large amount of variability in language, fonts, lighting conditions, background and style of handwritings. Algorithms developed for this problem have adopted the conventional approach by selecting the hand-engineered features for training the models. However, the models can only be expected to work well on a small range of handwritten texts and would not be generalized for any kind of written text because of the dependence of the algorithm on hand-engineered features. Our work would focus on working with a more all-encompassing framework constituting convolutional neural network and Long Short-Term Memory (LSTMs) for detection of handwritten text in the data and recognition of the text. The generalized framework would help us in grading handwritten assignments with less restrictions for the writer to work upon.**

***Index Terms*—Text Recognition, Convolutional Neural Networks, Long Short Term Memory, Optical Character Recognition**

I. Summary of work executed in previous weeks

In the previous weeks, we have tried to attack the problem at hand with different approaches.

Firstly, we used Convolutional Neural Networks (CNNs) on the dataset available on Kaggle, which contained astronomical number of images of single alphabets (3,70,000 images) and the dataset was trained quite well. As an outcome, the alphabets were detected and classified with a high accuracy. However, the approach used here was size variant and also all the detected contours in the image were passed to the CNN instead of detecting contours containing text which is infeasible in practical scenarios.

Secondly, we came across another approach as given in [1] to overcome this infeasibility by using Histogram of Oriented Gradients (HOG) as features for Linear SVM as mentioned in the previous report using a combination of datasets namely ”CIFAR10”[2] and ”74k chars”[3]. This Linear SVM was used for classifying the contours (obtained bounding rectangles) as containing text and not containing text, hence making the approach size invariant. As a result, the contours classified as those containing text will be passed to the CNN trained earlier in the first approach.

In the next section, we will be discussing about the work done on the second approach with a few supporting results.

Thirdly, we encountered certain cases of cursive writing images where the detected contours contained entire word instead of alphabets and the dataset (Kaggle) used for training CNN contained only alphabets. Hence, we studied few articles that used a complete end to end approach which will also be discussed in the next section.

II. Work executed in this week

In this week, we tried to train the Deep Convolutional Neural Network with LSTMs which uses an approach of end-to-end text recognition without recognizing and classifying contours or any data pre-processing steps. We did write down the code and started training the model, but this turned out to be a computationally intensive task. The dataset used here was IAM Dataset[4] which was highly enriched as indicated in the Project Report 2.

The dataset has been divided as follows :

- Train Samples : 80,000
- Validation Samples : 15,000

Training parameters :

- Batch-size = 200
- Number of Epochs = 50
- Learning rate = 0.001
- Momentum = 0.9

Initial observations of training process : Loss = 70%. After training the model for approximately 12 hours, the Loss reduced to 12% which is quite a reasonable value. However, the entire training process could not be completed and we would showcase the results in the upcoming Project Report.

We also tried to implement the second approach mentioned in the previous section that uses Linear SVM taking HOG as it’s input features. HOG features can be described as taking a nonlinear function of the edge orientations in an image and pooling them into small spatial regions to remove sensitivity to exact localisation of the edges.

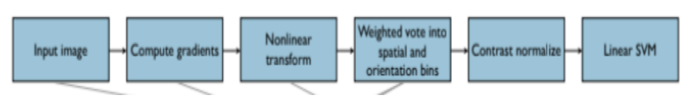
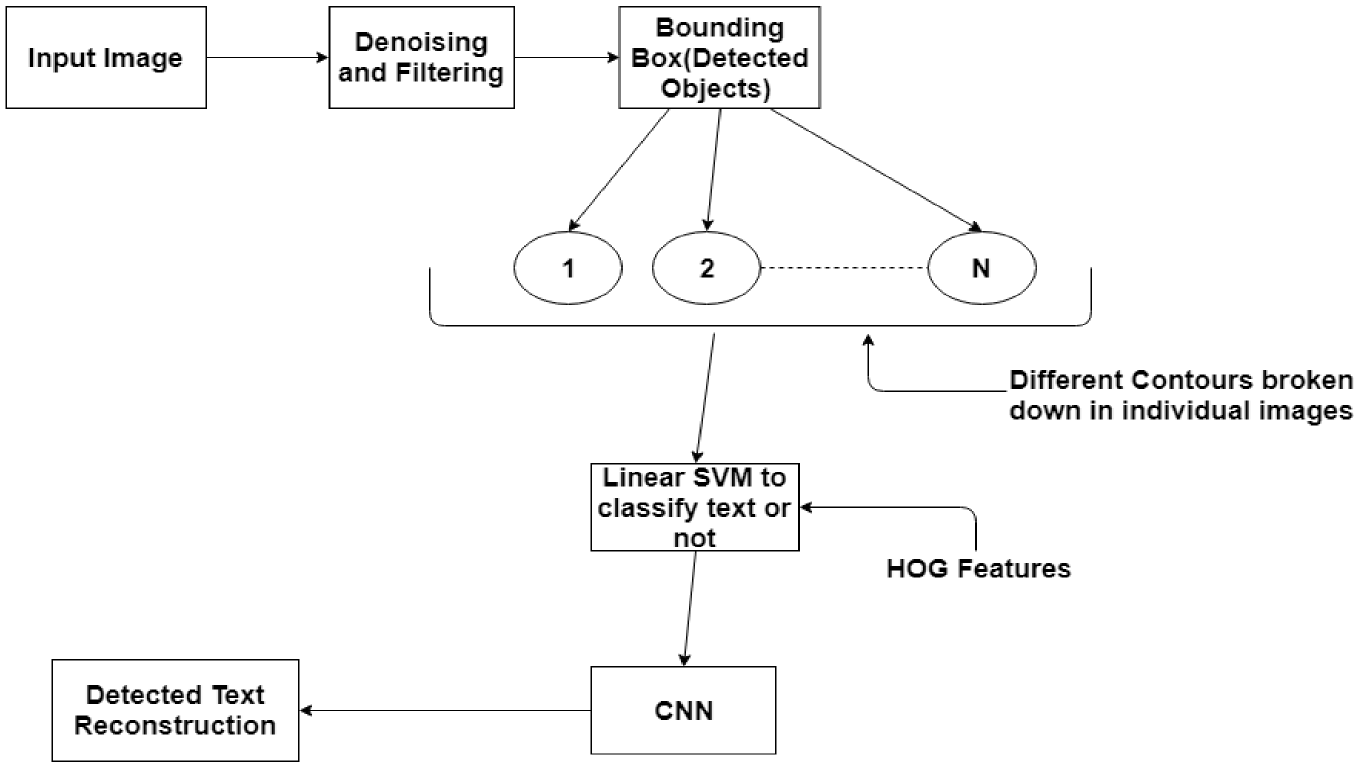


Fig. 1: An illustration of the HOG feature extraction process and how each component maps to our reformulation. Gradient computation is achieved through convolution with a bank of oriented edge filters. The nonlinear transform is the pointwise squaring of the gradient responses which removes sensitivity to edge contrast and increases edge bandwidth. Histogramming can be expressed as blurring with a box filter followed by downsampling.

It is a technique commonly used to extract relevant features from images (object detection for example) and then pass them to a classifier.

A summary of work done in this week is as shown in the flowchart on Page 2.



(a) Flowchart

III. Planned work for upcoming week

Next weeks initial focus would be to tune the parameters for training model of the end-to-end text recognition (CNN with LSTM) approach so that the training time reduces significantly and we can test the model on different images. Also, after completion of image testing, we would like to compare it's results with that of Linear SVM trained using HOG as features. We would first like to focus on the implementation of online handwriting recognition, employing recurrent neural networks for the task of recognition.

We would also work on completing our task of character recognition by reconstruction of the characters which were previously separated out with the help of contours and boundary boxes.

Followed by this, we would like to focus on the implementation of offline handwriting recognition, which seems to be a more challenging and interesting problem, which uses multidimensional recurrent neural networks for performing the task [5]. The most challenging task in this scenario would be to convert two dimensional input images into one-dimensional label sequences.

References

[1] Hilton Bristow, and Simon Lucey. "Why do linear SVMs trained on HOG features perform so well?." arXiv preprint arXiv:1406.2419 (2014).

[2] Alex Krizhevsky, "Learning Multiple Layers of Features from Tiny Images", 2009.

[3] T. E. de Campos, B. R. Babu, and M. Varma,"Character recognition in natural images", In VISAPP, 2009

[4] U. Marti and H. Bunke. The IAM-database: An English Sentence Database for Off-line Handwriting Recognition. Int. Journal on Document Analysis and Recognition, Volume 5, pages 39 - 46, 2002.

[5] Alex Graves, and Jrgen Schmidhuber. "Offline handwriting recognition with multidimensional recurrent neural networks." In Advances in neural information processing systems, pp. 545-552. 2009.