Handwritten Text Recognition

Anushree Rankawat (201501007), Shivam Raval (201501088), Om Thakkar (201501109), 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

We have accomplished the task of character detection using Convolutional Neural Networks (CNN) in the past weeks. Our basic algorithm works by creating boundary boxes around the characters and then breaking the characters into individual portions. We used Mean Shift Filtering to blur the image (denoising) which allowed us to easily adapt otsu threshold to the histogram of pixels. This helped us decide a good threshold for drawing contours around the objects detected. The contours give us 4 points which are used to construct bounding rectangles around the characters. The output of this particular code results in creating boundary boxes over all the characters in the image and separating the characters. A summary of the workflow executed in the previous week is as shown in Fig. (a) on the next page. The diverse nature of the dataset used for training helped train the CNN well and we could get an accurate result almost every time the model was tested. One of the glitches with our model was that it would end up detecting the whole word as a character if words were written in cursive.

II. Work executed in this week

This weeks focus was on correcting glitches of our character recognition algorithm. One of the improvisations we made was to test character recognition using a linear SVM model trained on HoG features. We read a paper which described why linear SVMs trained on HoG features perform better [1]. This week, we implemented it to evaluate the results of this particular algorithm with respect to our existing implementation of character recognition using CNNs. We did write down the code for the same and started training, but this turned out to be a computationally intensive task and we might need dedicated GPUs to train this particular model.

We also started the second half of our problem statement, which involves character recognition using our labeled dataset. This would involve reconstruction of the characters and we are presently working on the same.

We also worked for the task sentence detection and coherent word formation. For this, one task of utmost priority was to find a dataset which had handwritten words present in it, so that the neural network can work towards word detection and finally use Long Short Term Memory (LSTMs) for coherent word formation. IAM Handwriting Database is a dataset we found after extensive research [2]. This is a good database as it is very diverse, containing writings of 657 writers, 1,539 pages of scanned text, 5,685 isolated and labeled sentences, 13,535 isolated and labeled text lines and 115,320 isolated and labeled words. A sample of cursive snapshots given in the dataset is a shown in Fig. 1.

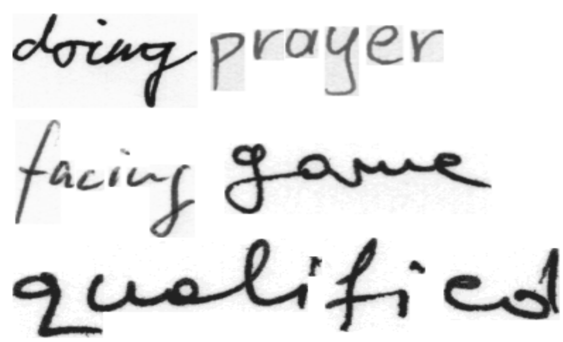
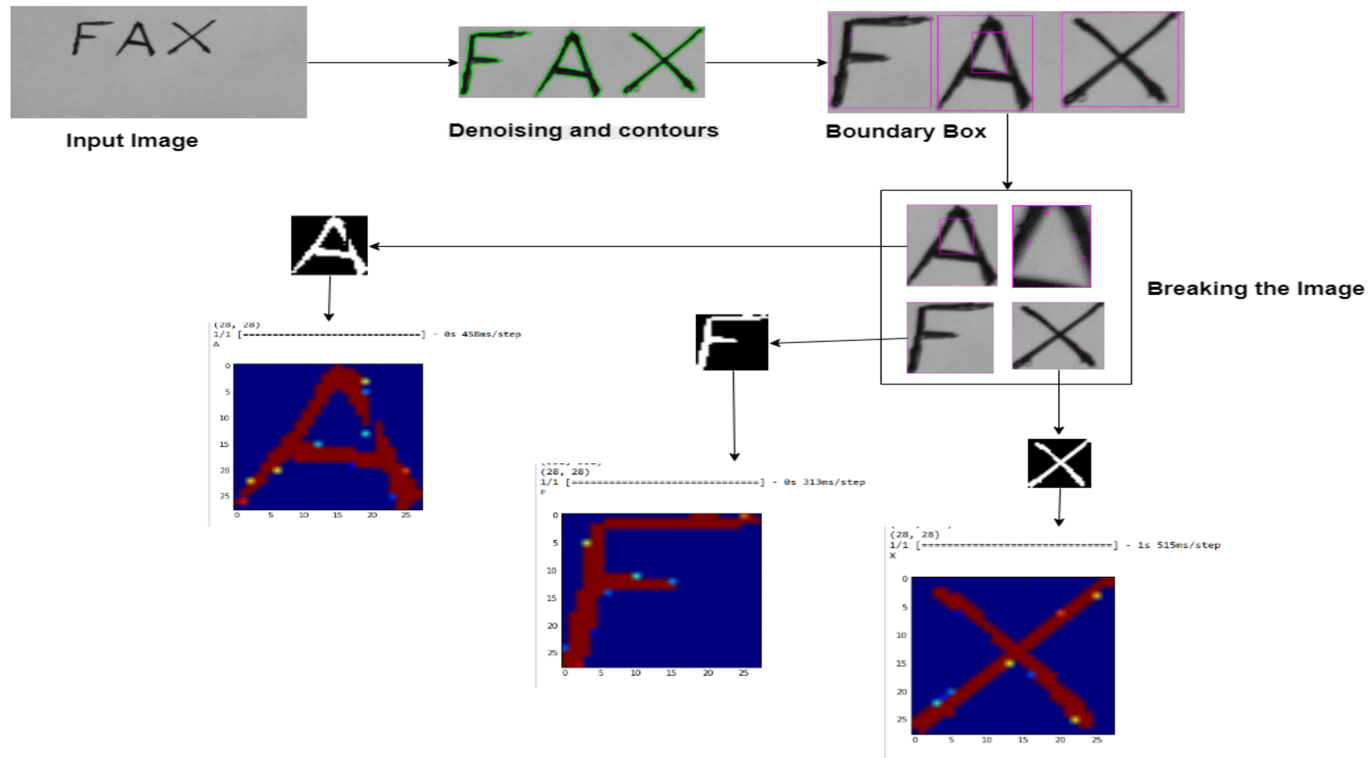


Fig. 1: A Sample of Curvise handwritings given in the IAM Dataset

This kind of hierarchical structure would help us in evaluating our model at these different levels of sentence formation. We plan on using this dataset for our training related to word detection and recognition.

Another focus of this week was to read papers and get a better understanding of how to use LSTMs for recognition of meaningful words and sentences. We found an article which discussed an offline algorithm for performing the task of handwriting recognition using Multidimensional Recurrent Neural Networks (RNN) [3]. This proves to be a challenging task as it can only extract features from the image and not the pen trajectory as is possible in the online case. Online case of the problem proves to be a much simpler one with a 1-D sequence as an input that could simply be fed directly to an RNN. LSTM network architecture could be used for this task, followed by a connectionist temporal output layer which allows data to be recognized without any prior segmentation.

Offline case turns out to be a little tricky, owing to the fact that we deal with handling many dimensions, leading to the need for employing multidimensional RNNs for the same. The



(a) Flowchart

best part of this algorithm/paper is that it can be employed for text recognition for any language without any language specific preprocessing required.

III. Planned work for upcoming week

Next weeks focus would be to read more articles on word detection in handwritten text and get a better understanding of the various prevalent algorithms present in the field for this particular task.

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.

We would like to try running our implementation of linear SVM trained on HoG features on a dedicated GPU for comparing the efficiency of this particular model when compared to our Neural Network.

We also wanted to get an implementation of CNN working in conjecture with LSTMs started so we could make some progress in that front next week. We would first like to focus on the implementation of online handwriting recognition, employing recurrent neural networks for the task of recognition.

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. 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] 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.

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