Layout Analysis on Challenging Historical Arabic Manuscript Using Siamese Network

Reem Alaasam ,Berat Kurar and Jihad El-Sana
The Department of Computer Science Ben-Gurion University of the Negev ,Beer Sheva, Israel

Introduction:

In this paper we present layout analysis method for historical handwritten Arabic documents using a siamese network. Siamese network consists of two identical convolutional neural networks (CNN). It takes as an input a pair of images or patches, extracts their features and ranks the similarity between them. Given pages of a historical Arabic manuscript, we segment them into patches of similar size and train siamese network model. Using the trained model we build a distance matrix among the patches of each testing page. Then we use the distance matrix to cluster the patches into three classes: main text, side text and background.

Method:

Our method is composed of three steps:

1. Converting our dataset which consists of page images, to a dataset of patches taken from each page image from the original dataset.
2. Training siamese network model that can predict the similarity between any two patches from our dataset.
3. Building a distance matrix using the trained model and classifying each patch of the same page to one of the following classes: main text, side text or background

1. Siamese Network

A siamese network consists of two branches that share the same Convolutional Neural Network (CNN) architecture and the same weights. The input is a pair of images and the output is a distance in the range [0, 1], which corresponds to the similarity of the input pair.

2. Architecture

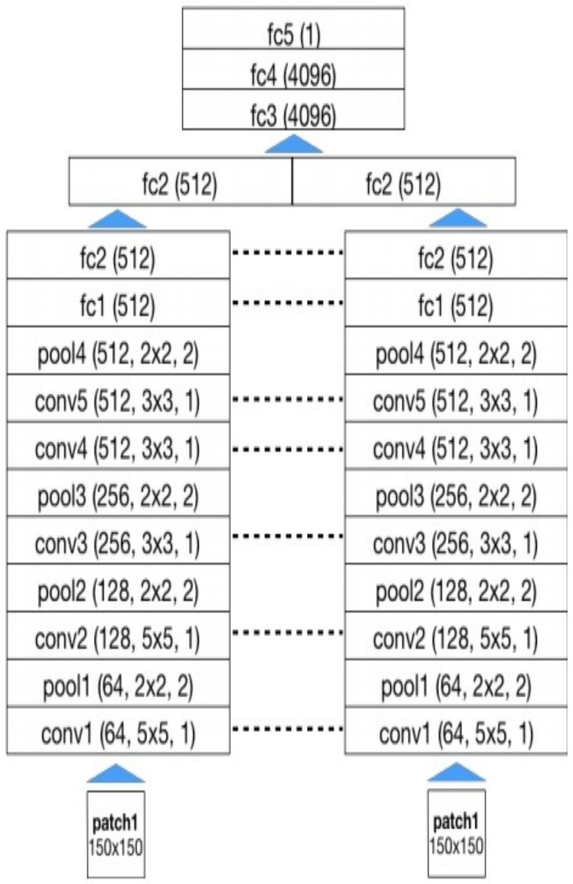

Fig. 1. Siamese architecture for pair similarity. Dotted lines stand for identical weights, conv stands for convolutional layer, fc stands for fully connected layer and pool is a max pooling layer.

Experiments:

1. DataSet

Training and testing were done with challenging historical Arabic manuscripts dataset. The dataset contains various writing styles and different layout structures. It contains 32 documents from 7 different historical Arabic manuscripts. We used 24 pages for training and 8 pages for testing.

Fig. 3. Three pages from the train set

2. Data preparation

We generated patches of 150×150 pixels using a sliding window over page images. In our dataset the average size of a page image is 2800 × 3900, hence each page provides around 470 patches. If a patch contains more than 1000 main text labeled pixels, it is labeled as main text patch, else if it contains more than 1000 side text labeled pixels, it is labeled as side text patch, else it is labeled as background patch. We picked 1000 as the number of pixels to decide the patch class through experiments.

3.Training

The input for a siamese network is a pair of images, in our case, it is a pair of patches. We generated the maximum number of positive pairs and for each positive pair a negative pair was generated randomly. We have approximately171652 pairs for training. We trained the model shown in the architecture section from scratch and reached 90%.

3. Postprocessing

After clustering we apply postprocessing step that refines the labels of the patches. For each patch in a page image, we consider its 8-neighbors and by majority vote we choose the new label of a patch.

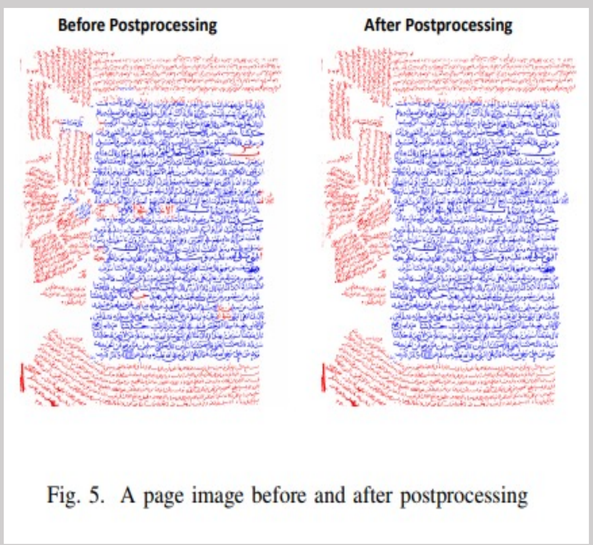

Fig. 5. A page image before and after postprocessing

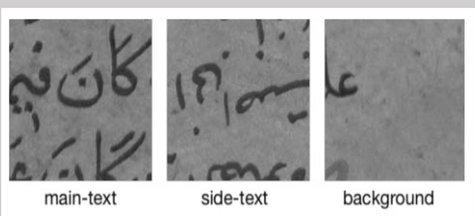

Fig. 4. Sample patches from each class.

TABLE I
COMPARISON WITH F-MEASURES, MT(MAIN TEXT) AND ST(SIDE TEXT)

	MT F-measures(%)	ST F-measures(%)
Bukhari et al [2]	95.02	94.68
Kurar et al. [4]	95	80
Proposed method	**98.59**	**96.89**

Results

We compare our work with others, who use the same dataset. As shown in Table I, we outperform their works in both the main text and side text segmentation. Our method works on patch level instead of page level. This increases the size of train and test sets, and make it possible to have a larger dataset, which is always preferable in. The input images have complex layout structures. In addition, they have a variety of skewed and curved lines, bleed-through and noise. even with these challenges our method is still effective.

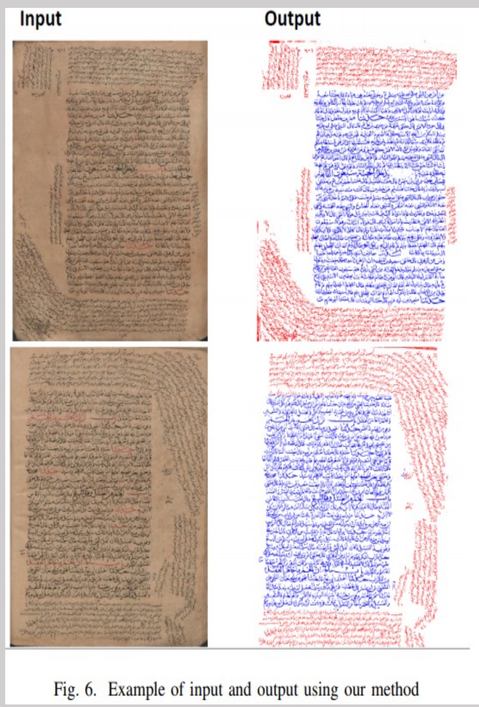

Fig. 6. Example of input and output using our method