Classification and Reconstruction of Chinese Font Style Milestone 3

Yiliang Li,Jiejin Deng, Zijin Shi

October 26, 2017

1 Illustration of dataset

We have found open source software which can transfer standard font files(.ttf) to images of characters in different fonts. Fig.1 is an illustration of the image we generated, where each character is 80*80 pixels.

Figure 1: Illustration of the data set generation

2 Implementation / Algorithmic environment

We will use Matlab as our developing environment. Firstly we will use Matlab Computer Vision System Toolbox to implement Scale-Invariant Feature Transform (SIFT) to extract important local features (curvature, density, strokes, etc.) of each characters. Then we will build the visual 'vocabulary' by our own codes, and write codes to implement SVM and k-NN algorithms to classify different font types. Another approach is to write our own codes to use the convolutional neural network(CNN) so that features can be automatically selected by the network. For each font, about 500 characters will be our training set, and about 50 characters will be our test set, where max-pooling, dropout and softmax techniques will be implemented.

For the character transfer part, we will write our own codes to implement a 3-layer neuron network to minimize the loss function below:

$$L = \Sigma_k \Sigma_{i,j} \left\| (\hat{y}_k)_{i,j} - (y_k)_{i,j} \right\|_1 \tag{1}$$

where $\hat{y}$ is the output pixel matrix at the end of the CNN process we obtained from x_k and y_k is the pixel matrix corresponding to the same character in the target font style.