Optical Character Recognition based on Kohonen Map

SIGMA 205 Project Report

Chen Wei, Victor Clement

Instructed by Pr. Laurence Likforman

June 2017

1 Introduction

Optical character recognition (OCR) is an important research field for pattern recognition. It aims to converting images of typed, handwritten or printed text into machine-encoded text.

Self-Organizing Map (Kohonen Map) is one typical approach in the domain of machine learning. It is an artificial neural network that uses unsupervised learning to project a set of multidimensional data into restrained number of dimensions.

In order to apply the Kohonen map, the characters should be firstly summarised into characteristic data sets. In our work those data set are moments of the character images normalized by the root mean square so that the values are all in a similar range.

We are equipped with a set of training data for whose labels are known and a set of test data extracted from characters images of lower quality. Our objective is to apply the Kohonen algorithm to project the data into lower dimension, and try to classify the test data based on the training model in lower dimension.

2 Kohonen Algorithm

The Kohonen algorithm is a useful tool for visualizing data by project it into a constrained dimensional space which is called Kohonen maps (SOMs Self Organizing Maps). To construct the Kohonen maps, on should firstly construct a grid in which each point is a node. A node is physically linked with its neighbouring nodes however not linked with the nodes outside its neighbourhood. In the training process each sample of data will be located in the grid by selecting the best matching node of each sample. The table 1 shows the principle scheme of how to project the training data on the 2-dimension Kohonen card.'

Table 1. Kohonen algorithm of 2 dimension

Initialise $D_0, L_0, \lambda_D, \lambda_L$

Initialise each node (x, y) with a random weights vector $W_0(x, y)$ of the same dimension as training sample.

$for\ n = 1, \dots, N$

1. Randomly select a sample V_n from the training dataset
2. Based on the Euclid distance between the sample and the weights vector attached to each node, determine the nearest node (x_0, y_0)in the grid, which is called the Best Matching Unit (BMU).
3. Update the size of neighbouring area of BMU

$$D_n = D_0 \exp\left(-\frac{n}{\lambda_D}\right)$$

4. Determine the weights of nodes in the neighbouring area.

$$\Theta(x, y) = \begin{cases} 1, & |(x - x_0)^2 + (y - y_0)^2| < D_n^2 \\ 0, & otherwise \end{cases}$$

5. Update the step length for evaluating the weight vectors

$$L_n = L_0 \exp\left(-\frac{n}{\lambda_L}\right)$$

6. Evaluate the weight vectors of the nodes that are in the neighbouring area of the BMU

$$W_{n+1}(x, y) = W_n(x, y) + \Theta(x, y) L_n (V_n - W_n(x, y))$$

Following the process above, we will finally get a grid in which all the attached weight vectors provide a representation the training data and organize it according to its proximity.

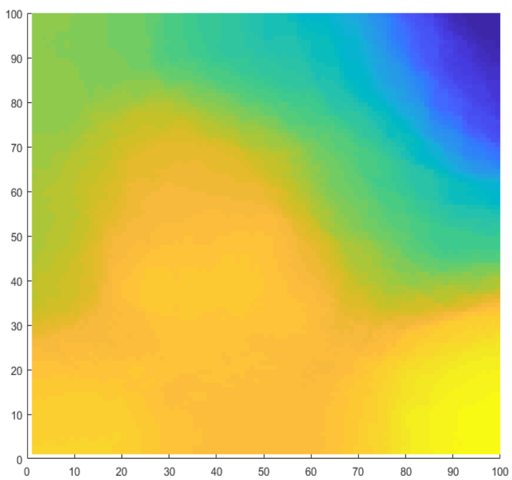

Figure 1a. Coloured Kohonen map

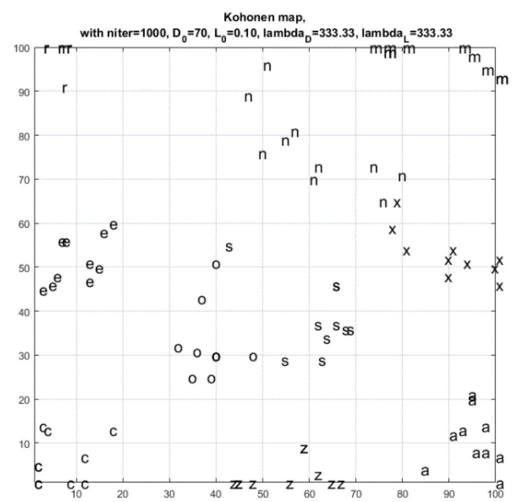

Figure 1b. Locate the labelled training samples in the Kohonen map

3 Classification approach

As stated in last section, training a Kohonen map doesn't need the label of each sample, which means the algorithm auto-organise training samples in the map without knowing which class they belong to. However if we attach the label (the character) to each sample, and locate the sample in the map, we will equivalently attach the label to the node. With the best matching principle, we can see in figure 1b that samples with the same label are organised in the same area.

Now we have projected the training data into 2D space with size 100×100, then there are varieties of classification methods that we can take advantage of. In our work, we implement one of the simplest one KNN (K nearest neighbours) classifier to classify the test data.

First we locate the test data samples on the map based on the similarity between the sample and the weight vector of the labelled nodes. Again we will apply the best matching principle meaning that we should find the labelled node which maximizes the similarity. Then search the K nearest neighbours around the test sample. The label of the test sample will be determined by the most frequent neighbours around it.

4 Experiments

4.1 Introduction of data sets

Two data sets are provided respectively for training and test process. Each data set contains 100 samples extracted from 10 classes, and each class represents one English letter from {a, c, e, m, n, o, r, s, x, z}. We choose 10 classes because it will be more practical for following calculation.

Each sample in both training and test set has 8 features which are extracted from a 300dpi image. We present several samples of these images in figure 2.

Figure 2. Training set: 10 samples of image of printing text

The 8 features are the moments of each image containing isolated character: $\{m_{00}, \mu_{02}, \mu_{11}, \mu_{20}, \mu_{03}, \mu_{12}, \mu_{21}, \mu_{30}\}$, which are defined as

$$m_{pq} = \sum_j \sum_i (i^p j^q I(i,j))$$

$$\mu_{pq} = \sum_j \sum_i (i - i_{mean})^p (j - j_{mean})^q I(i,j)$$

where (i_{mean}, j_{mean}) is the coordinate of the image's centroid

$$i_{mean} = m_{10}/m_{00}$$

$$j_{mean} = m_{01}/m_{00}$$

To quantify the image, we consider that the pixel $I(i,j)$ equals to 1 if it is back, and equals to 0 if it is white.

To eliminate the bias of sampling, and ensure that all the samples in the same range, the data should be normalised. For each feature i, we calculate the RMS (Root Mean Square) of all samples in training set,

$$RMS(i) = \sqrt{\frac{\sum_{k=1}^{N}(x_i^{(k)})^2}{N}}\ i = 1, \dots, 8$$

where $x_i^{(k)}$ is the feature i of the k-th sample in training set.

And we divide each feature in both training and test set by the corresponding RMS to normalize the data,

$$\bar{x}_i^{(k)} = x_i^{(k)}/RMS(i)$$

4.2 Parameter optimisation

As stated in the previous section, the Kohonen algorithm is very sensitive to the grid size (s), the window size to select the neighbours of BMU (D_0), the learning step length (L_0), and the number of iterations (*N*). We assume that the influence of each parameter is independent. In order to get the optimal parameters for the classification, we use the grid search method to evaluation the algorithm with different parameters.

First, to verify the influence of N, we vary N from 600 to 2000 with other parameters fixed as $s = 100, D_0 = 40, L_0 = 0.1, \lambda_D = \frac{N}{3}, \lambda_L = \frac{N}{3}$. We find the resulted classification accuracy is highest when $N \in \{800,1000,1200\}$. So in the following experiments we will fix N=1000 and vary other parameters to see their influence on the accuracy.

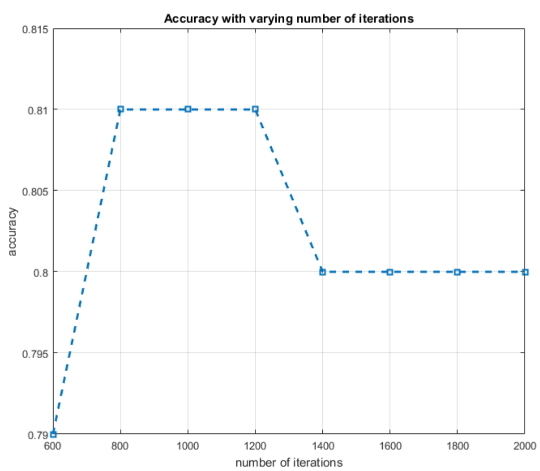

Figure 3a. Accuracy with varying number of iterations, $s = 100, D_0 = 40, L_0 = 0.1, \lambda_D = \frac{N}{3}, \lambda_L = \frac{N}{3}$

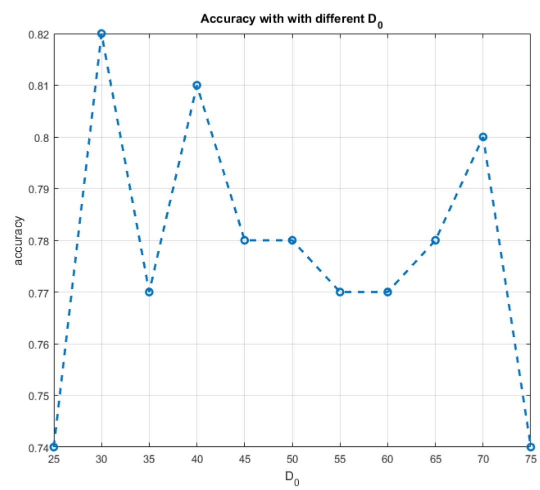

Figure 3b. Accuracy with varying neighbour window size, $N = 1000, s = 100, L_0 = 0.1, \lambda_D = \frac{N}{3}, \lambda_L = \frac{N}{3}$

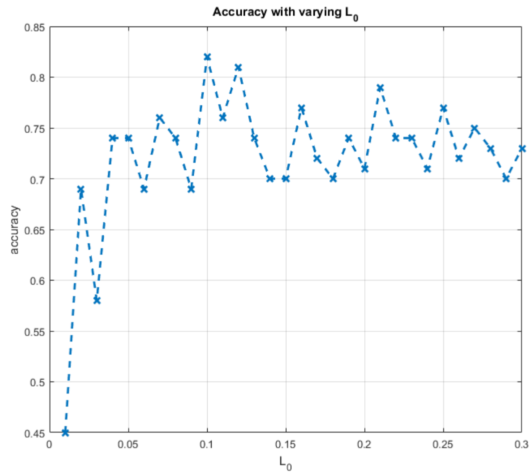

Figure 3c. Accuracy with varying learning step length size, $N = 1000,\ s = 100, D_0 = 30, \lambda_D = \frac{N}{3}, \lambda_L = \frac{N}{3}$

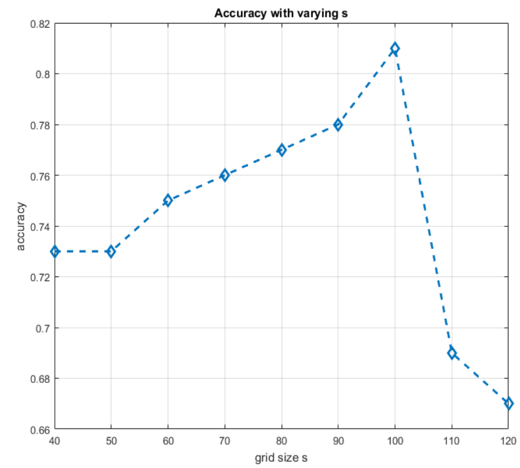

Figure 3d. Accuracy with varying grid size, $N = 1000, D_0 = \frac{s}{3},\ L_0 = 0.1, \lambda_D = \frac{N}{3}, \lambda_L = \frac{N}{3}$

In the similar way, we vary D_0 with other parameters unchanged, and we obtain the optimal $D_0 = 30$.

To see the explicit influence of D_0 on the Kohonen map, we show 2 Kohonen maps with different setting of D_0, in which we conclude that D_0 determine the distance between clusters of characters on the map. Basically larger D_0 will lead to larger distance between clusters.

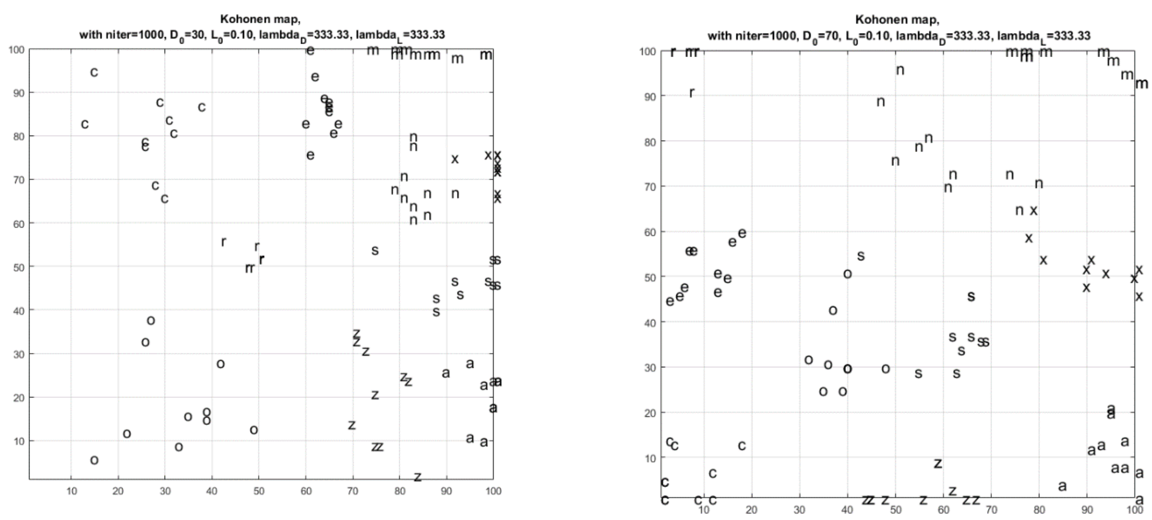

Figure 4. Kohonen map with $D_0 = 30$ and $D_0 = 70$

From figure 5 we can see that L_0 will affect the density of each cluster, i.e. the average distance between elements in each cluster. Smaller L_0 will lead to higher density of the clusters, making the same characters more condensed.

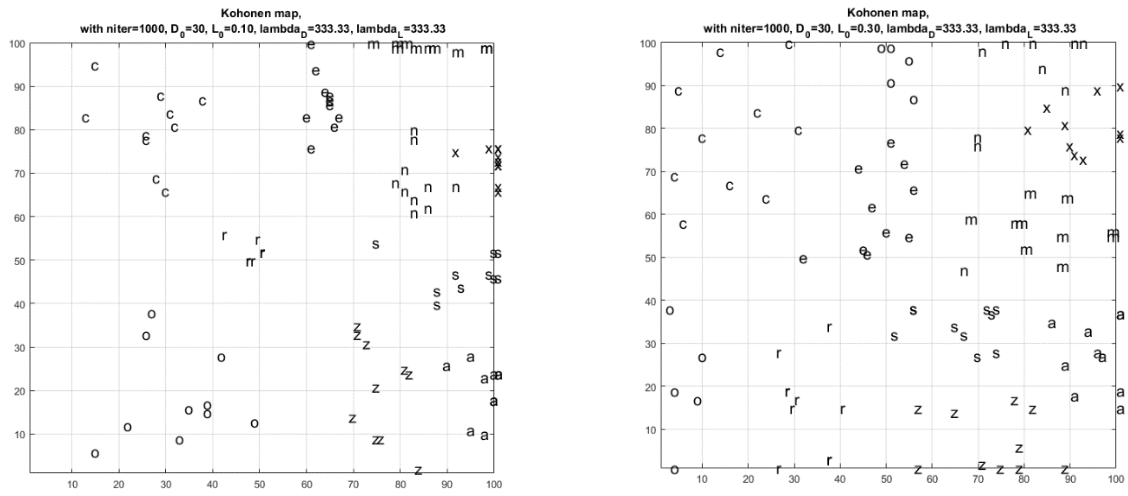

Figure 5. Kohonen map with $L_0 = 0.1$ and $L_0 = 0.3$

4.3 Classification results

According to the optimized parameters obtained in last subsection, we fix the parameters for the final classification as follows:

Table 2. Optimal parameters for classification

N	1000
s	100
D_0	30
L_0	0.1
λ_D	$N/3$
λ_L	$N/3$

In figure 6, the samples in training and test set are located in the fitted Kohonen map according to the best matching principle stated in the previous section.

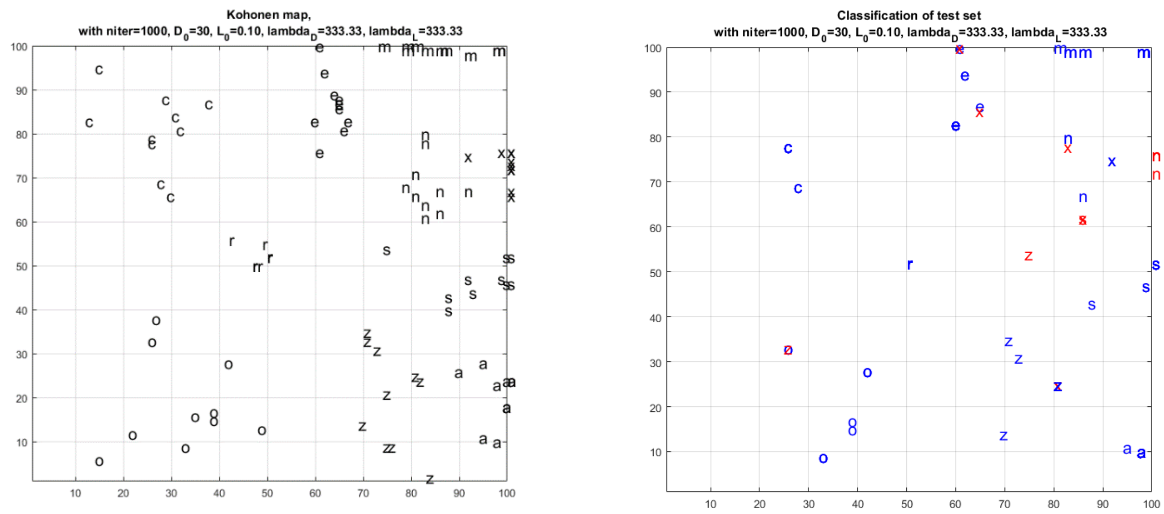

Figure 6. Kononen map with location of training samples and classification results for test samples

We can see from figure 6 that the same characters are located closed to each other in the map, which indicates that the sample in the same class are well cluster by their location on the map. The whole map can be segmented into 10 sections corresponding to 10 different classes. If a sample in the test set is located in a certain section, we will determine its label as the label of this section.

Table 3. Prediction results of test set

	a	*c*	*e*	*m*	*n*	*o*	*r*	*s*	*x*	*z*	accuracy
a	10										1.0
c		10									1.0
e			10								1.0
m				10							1.0
n					3				7		0.3
o						10					1.0
r							10				1.0
s					1			9			0.9
x			2		2				2	4	0.2
z					1	1				8	0.8
Average											**0.82**

The prediction results of test set are shown in table 3 where we can see that most prediction errors occur for the character ‘x’ and ‘n’. If we verify the image for x in the data set, we will find the character x in the training set is seriously distorted, which makes it hard to predicted for the test set.

5 Conclusion

Kohonen algorithm provides as a practical tool for visualize the high dimensional data. If all types of data can be viewed in 2D or 3D space, it will be more intuitional for us to observe the characteristic, the similarity and the diversion of the data, and then classify them.

In our work this approach is applied to the optical character recognition, and we have obtained 82% accuracy for 10 classes of scanned characters. Further work including comparing the performance of different classifiers, implementing for handwriting characters recognition will be done.