EE5907 CA2 REPORT

Xin Zhengfang E0427425/A0206597U

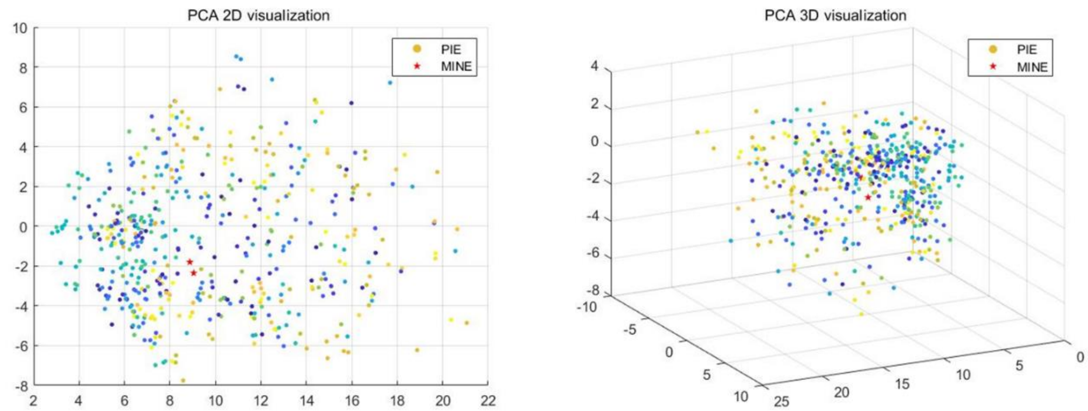
CONTENTS

Introduction

Because photos are taken with some order (face direction) both CMU PIE dataset and my photos, I shuffle the dataset to reduce this stereotype and to see the true properties of different methods.

PART-1 PCA

My visualization results of projected data vector in 2d and 3d plots using PCA show in Fig.1.



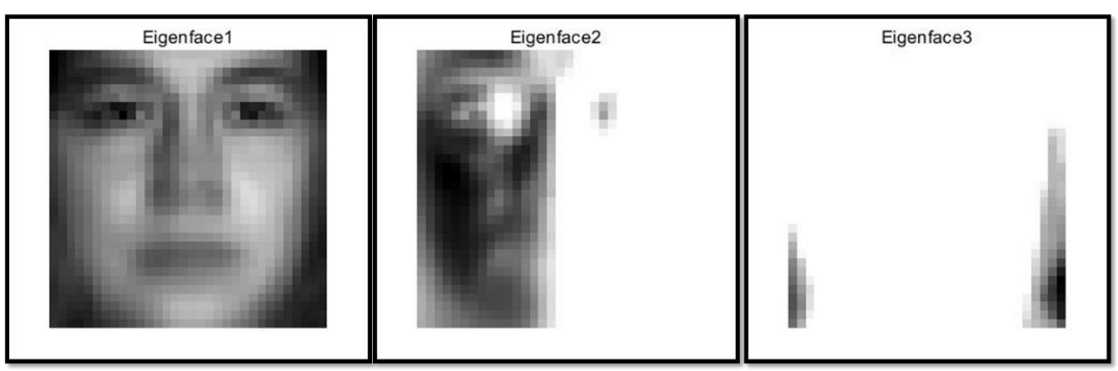
(a) PCA 2D visualization

(b) PCA 3D visualization

Fig.1. Projected data vector using PCA in 2d and 3d visualization. (i) dot point is CMU PIE dataset (ii) red star point is my photos.

Because PCA is unsupervised representation learning, we cannot see obvious distinction between different classes on the 2d or 3d plots.

My corresponding 3 eigenfaces show in Fig.2.



(a) Eigenface 1

(a) Eigenface 2

(a) Eigenface 3

Fig.2. Eigenfaces

Face images reduced by PCA into 40, 80 and 200 dimensionalities are classified by the k-nearest neighbor (knn) method. And the results show in Table 1.

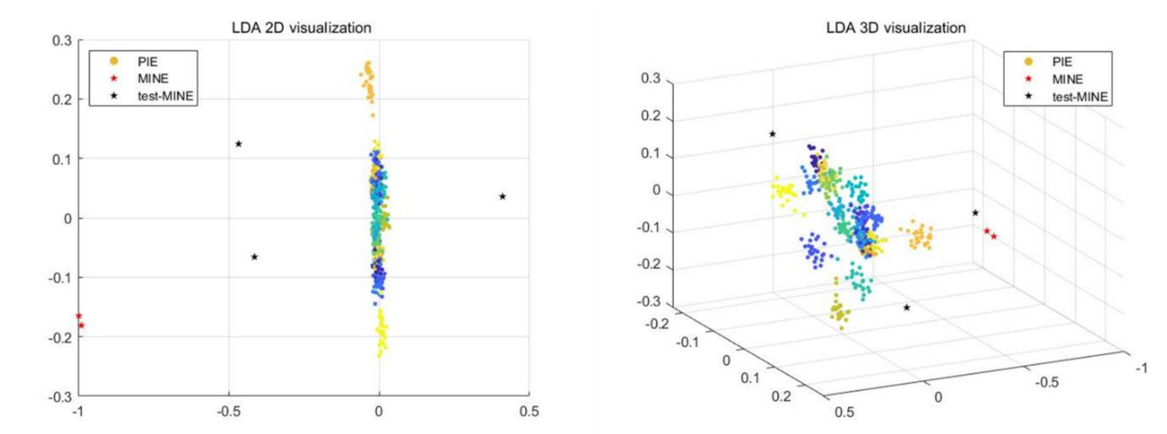
Table 1. Accuracy on face images of 40, 80 and 200 dimensionalities reduced by PCA

Dataset\Dimensionalities	**40**	**80**	**200**
PIE	92.06%	94.31%	95.20%
MINE	100%	100%	100%

With the dimensionality increase, the classification accuracy achieves better performance, because of more information kept. PCA is also a kind of compressed technology and it can project data into less dimensionalities with the richest information. We should balance the computation cost and performance when we use PCA.

PART-2 LDA

My visualization results of projected data vector in 2d and 3d plots using LDA show in Fig.3. Because knn classification results on LDA reduced dimensionality is not good on my photos, I also plot my test photos on Fig.3.



(a) LDA 2D visualization (b) LDA 3D visualization

Fig.3. Projected data vector using LDA in 2d and 3d visualization. (i) dot point is CMU PIE dataset (ii) red star point is my photos. (iii) black star point is my test photos

CMU PIE data point all almost compressed into one line at 0 on horizontal axis in 2d and 3d. I think this is because of my photos. Moreover, I think PIE is taken by the same criterion, and my photo is taken causally without this criterion. Because of this “stereotype”, my photo is really distinctive from PIE photos, we can easily tell my photo and PIE photos in 2d and 3d plots. Generally, all data point successfully are clustered with their class group supervised by LDA.

Face images reduced by LDA into 2, 3 and 9 dimensionalities are classified by the knn method. And the results show in Table 2.

Table 2. Accuracy on face images of 2, 3 and 9 dimensionalities reduced by LDA

Dataset\Dimensionalities	**2**	**3**	**9**	**40**
PIE	32.84%	53.53%	85.59%	96.76%
MINE	0%	0%	0%	0%

As the dimensionality increase, the classification accuracy achieves better performance except on my photos. This is because of my personal training photo is too small to cover all my characters. We can also observe this problem from Fig.3. My test photo didn’t closely surround my training photo. Moreover, the dimensionality is also too small to get better performance than PCA. When LDA dimensionalities increase to 40, the accuracy on PIE is even much better than PCA 200.

PCA vs LDA

LDA takes class label into consideration, which makes LDA has more information and more makes sense. But shall LDA always work better than PCA?

From the two above experiments, the answer is no! We can see that LDA doesn’t do well on my photos, but PCA achieve 100% accuracy on my photo. Therefore, we can conclude that PCA performs better in case where number of samples per class is less. Whereas LDA works better with large dataset having multiple classes; class separability is an important factor while reducing dimensionality.

PART-3 SVM

In this part, we use Support Vector Machine (SVM) to classify the reduced dimensionality of face images by PCA.

The classification accuracies on 80 dimensionalities reduced by PCA show in Table 3.

Table 3. SVM classification accuracy on face images of 80 dimensionalities reduced by PCA

Penalty C\ Dimensionalities	**80**
2	98.63%
1	98.83%
0.1	98.05%
0.01	82.11%

I add extra experiment that penalty C equals to 2. Because I want to show when penalty become larger it will also degrade the performance. We can also see this phenomenon in Table 4.

Table 4. SVM classification accuracy on face images of 200 dimensionalities reduced by PCA

Penalty C\ Dimensionalities	200
2	99.22%
1	99.32%
0.1	98.63%
0.01	83.87%

The penalty should not be too big and too small. We should balance this hyper-parameter to get best result. Moreover, because 200 dimensionalities have more information, the classification accuracy achieves better performance than 80 dimensionalities with the same penalty value.

KNN vs SVM

Comparing the results of KNN and SVM, it is clear that SVM works better than KNN.

KNN has no training time, but cost the memory to remember all the dataset and has more test time. This is not what we want, but implement KNN algorithm is really simple.

SVM cost time on training processing, but it saves the memory and test time comparing with KNN. Moreover, soft-margin SVM is more robust with noise comparing with KNN. These are reasons why SVM works better than KNN.

PART-4 CNN

My training batch is 10 and training iteration is 20,000. I use Adam to optimize the weights. The learning rate is 1e-3 and the weight decay is 1e-6. My loss function is cross-entropy loss.

The training processing shows in Fig. 4.

```
step:17800  loss:0.000021
step:18000  loss:0.000020
step:18200  loss:0.000018
step:18400  loss:0.000014
step:18600  loss:0.000016
step:18800  loss:0.000015
step:19000  loss:0.000012
step:19200  loss:0.000011
step:19400  loss:0.000011
step:19600  loss:0.000010
step:19800  loss:0.000010
step:20000  loss:0.000010
Finished Training
```

Fig. 4. The training processing

The classification accuracy on test set is **99.61%**.

Because this task is relatively simple for CNN, the result is not surprising.