Spring 5-2017

Statistical Learning Methods for Facial Recognition

Mengyi Jia
Washington University in St. Louis

WASHINGTON UNIVERSITY IN ST. LOUIS

Department of Mathematics

Statistical Learning Methods for Facial Recognition

by

Mengyi Jia

A thesis presented to

The Graduate School

of Washington University in

partial fulfillment of the requirements

for the degree of Master of Arts

May, 2017

St. Louis, Missouri

Contents

List of Tables

List of Figures

Acknowledgements

I'd like to express my deepest gratitude to my advisor, Prof. Kuffner, for his excellent guidance, patience, and providing me with an encouraging atmosphere for doing research. I would never have been able to finish my thesis without his gradual guidance of how to research into a new field for me.

I'd like to thank Prof. Ding agree to attend my thesis defense, and she gave me a lot of good suggestions for my first year's study here.

And I really appreciate Prof. Wickerhauser agreed straightaway to be in my thesis defense committee.

This research uses and modifies on the basis of the package of 'PCA Based Face Recognition System Using ORL Database'. Special thanks goes to the package author, Shujaat Khan from Iqra University, Pakistan.

This research uses AT&T face database, which was collected by AT&T Laboratories Cambridge.

Mengyi Jia

Washington University in St. Louis

May 2017

ABSTRACT OF THE THESIS

Statistical Learning Methods for Facial Recognition

by

Mengyi Jia

Master of Arts in Statistics

Under the Direction of

Professor Todd Kuffner

Washington University in St. Louis, May 2017

Facial recognition techniques have become increasingly popular in recent decades. This thesis investigates the performance of several methods applied to two different face databases, under a variety of poses and illumination settings. PCA, LDA and KNN are compared and contrasted in terms of their accuracy and processing time.

Chapter 1

Introduction

Human face recognition is now a very useful tool, involving statistical and mathematical models, together with computer implementation, which is capable of identifying a person from a digital image or video source. Among existing approaches, facial recognition techniques can be divided into two groups based on the face representation they use:

1. Appearance-based, which uses holistic texture features and is applied to either whole-face or specific regions in a face image;
2. Feature-based, which uses geometric facial features (mouth, eyes, brows, cheeks etc.) and geometric relationships between them [Delac et al., 2005].

When one image is converted to one observation of a dataset, it usually has hundreds of thousands of variables, each representing one pixel value. Among many approaches to the problem of face recognition, appearance-based subspace analysis still gives the most promising results [Delac et al., 2005]. Subspace analysis is aimed at projecting the images into a lower dimensional space (subspace). Finding an adequate subspace is the most challenging part of subspace analysis. Also we can measure distances between images of the original space to avoid the challenge of finding an adequate subspace, but it needs much more storage space and computational operations when the dataset is large.

Motivated by the comparisons of efficiency of facial recognition algorithm implementations in detail, this paper presents a comparison study of three appearance-based face recognition methods PCA, LDA and KNN on ORL database, and database made up of photos downloaded online. We study the face recognition accuracy and processing time in equal conditions. By applying these algorithms on two different datasets, we can further differentiate the three methods' advantages and disadvantages and investigate factors which could influence accuracy.

Chapter 2

Database

Along with the development of face recognition algorithms, face image data acquisition and creation of databases have been of great interest for the last few decades. However, many of these databases are tailored to the specific needs of the algorithm under development [Gross, 2011]. The accuracy of results of face recognition research heavily depends upon the versatility (presence of moderately large representative samples) of the database used.

2.1 ORL Database

The AT&T face database, sometimes also known as ORL database of faces, was collected between 1992 and 1994. It contains 10 different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open or closed eyes, smiling or not smiling) and facial details (glasses or no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position with tolerance for some side movement.

Table 2.1 ORL Data Review

Number of Subjects	Number of Pixels	Number of Images
40	92×112	400
http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html		

Figure 2.1 Part of ORL Database

2.2 Self-made Database

Table 2.2 Self-made Data Review

Number of Subjects	Number of Pixels	Number of Images
10	92×112	50

This database is composed of 10 distinct subjects with 5 different images each. It contains 7 females and 3 males, from whom there are 8 Asians and 2 Westerners. The photos were taken on different conditions, there is much variety in background and photo quality between subjects. At the same time, some subjects had more variety in poses, some tilted head, or lowered head, and the facial expression was much more vivid. The database is colorful, which will be converted to gray-scaled photos by MATLAB.

Figure 2.2 Self-made Database Converted to Gray-scale

(a) Self-Made Part1 **(b)** Self-made Part2

Chapter 3

Face Recognition Method

In this thesis, before the implementation of LDA, we will do an initial dimension reduction using PCA, due to the limitation of operations on a large-dimensional matrix. This may affect the accuracy of LDA, and a further discussion in Chapter 4 will show that.

3.1 Principle Component Analysis (PCA)

Principal component analysis (PCA), is a classical technique which can be easily understood and applied. It is a statistical method which belongs to the group of factor analysis. The PCA is aimed at reducing the large dimensionality of the data space to a smaller dimensional one, which is quite suitable and efficient for processing the image dataset.

3.1.1 Mathematics of PCA

By concatenating column by column (or row), a 2-D facial image can be converted to a long thin 1-D vector. Let's suppose a random vector X,

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix}, \tag{3.1}$$

with variance-covariance matrix

$$var(X) = \Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \dots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \dots & \sigma_p^2 \end{pmatrix}. \tag{3.2}$$

Consider the following linear combinations:

$$\begin{aligned} Y_1 &= e_{11}x_1 + e_{12}x_2 + \cdots + e_{1p}x_p \\ Y_2 &= e_{21}x_1 + e_{22}x_2 + \cdots + e_{2p}x_p \\ &\vdots \\ Y_p &= e_{p1}x_1 + e_{p2}x_2 + \cdots + e_{pp}x_p, \end{aligned} \tag{3.3}$$

Y_i is a linear combination of $x_1, x_2, \dots, x_p$, and $e_i = (e_{i1}, e_{i2}, \dots, e_{ip})$ is viewed as regression coefficients in Real field, and have the properties:

$$var(Y_i) = e_i^T \Sigma e_i, \quad cov(Y_i, Y_j) = e_i^T \Sigma e_j. \tag{3.4}$$

First Principal Component

The first principal component is the linear combination of x-variables that has maximum variance. More formally, select $e_1 = (e_{11}, e_{12}, \dots, e_{1p})$ that maximizes

$$var(Y_1) = e_1^T \Sigma e_1, \tag{3.5}$$

subject to the constraint that

$$e_1^T e_1 = \sum_{j=1}^{p} e_{1j}^2 = 1. \tag{3.6}$$

Further Principal Components

The second principal component is the linear combination of x-variables that has maximum variance for the remaining data (exclude the variation which the first component accounts for), and it's subject to the constraint,

$$e_2^T e_2 = \sum_{j=1}^{p} e_{2j}^2 = 1, \tag{3.7}$$

along with the additional constraint that these two components will be uncorrelated with one another,

$$cov(Y_1, Y_2) = e_1^T \Sigma e_2 = 0. \tag{3.8}$$

The ith principal component maximizes

$$var(Y_i) = e_i^T \Sigma e_i, \tag{3.9}$$

with the constraints,

$$e_i^T e_i = \sum_{j=1}^{p} e_{ij}^2 = 1 \ and \ cov(Y_1, Y_i) = 0, \ \ldots, cov(Y_{i-1}, Y_i) = 0. \tag{3.10}$$

The solution of coefficients involves the eigenvalues and eigenvectors of the variance-covariance matrix Σ.

Let λ_1 through λ_p denote the eigenvalues of the variance-covariance matrix,

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p, \tag{3.11}$$

and the corresponding eigenvectors e_1 through e_p are

$$\lambda_1 e_1 = \Sigma e_1, \quad \lambda_2 e_2 = \Sigma e_2, \quad \ldots, \quad \lambda_p e_p = \Sigma e_p. \tag{3.12}$$

The PCA chooses the m eigenvectors with the largest eigenvalues of Σ, where $p \gg m$, but it is enough to account for the variation among observations. Thus the goal of dimension reduction is achieved.

The variance-covariance matrix can be written as the sum over the p eigenvalues, multiplied by the product of the corresponding eigenvector times its transpose as shown below:

$$\begin{aligned} \Sigma &= \sum_{i=1}^{p} \lambda_i e_i e_i^T \\ &\approx \sum_{i=1}^{m} \lambda_i e_i e_i^T. \end{aligned} \tag{3.13}$$

Let the dataset be

$$X = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}, \tag{3.14}$$

and let $M = mean(X) = (m_1, m_2, \ldots, m_p)$, W is the decentered dataset,

$$W = \begin{pmatrix} x_{11} - m_1 & x_{12} - m_2 & \ldots & x_{1p} - m_p \\ x_{21} - m_1 & x_{22} - m_2 & \ldots & x_{2p} - m_p \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} - m_1 & x_{n2} - m_2 & \ldots & x_{np} - m_p \end{pmatrix}. \tag{3.15}$$

We can find the eigenvalues and corresponding eigenvectors of WW^T instead of $\Sigma = W^T W$ to avoid large number of operations [Kim, 1996],

$$WW^T f_i = \lambda_i f_i, \tag{3.16}$$

by pre-multiplying left W^T to both sides, we have

$$W^T W \left(W^T f_i\right) = \lambda_i \left(W^T f_i\right). \tag{3.17}$$

Let f_i be the eigenvector of WW^T corresponding to the ith eigenvalue (in descending order), $W^T f_i$ is the eigenvector of $W^T W$, which is also called the ith eigenface by PCA.

3.2 Linear Discriminant Analysis (LDA)

Linear discriminant analysis (LDA) attempts to find a linear projection from the image space to a low dimensional space by maximizing the between-class scatter and minimizing the within-class scatter. We assume there are L classes.

3.2.1 Mathematics of LDA

In discriminant analysis, the criterion of class separability is formulated by within-class and between-class scatter matrices.

A within-class scatter matrix shows the scatter of samples around their group's expected vector, and is expressed as below:

$$S_{wx} = \sum_{i=1}^{L} P_i \, E\left\{ (X - M_i)\,(X - M_i)^T \,|X \in Group \; i \right\}. \tag{3.18}$$

A between-class scatter matrix is the scatter of the groups' expected vectors around the overall mean, expressed as below:

$$S_{bx} = \sum_{i=1}^{L} P_i \; (M_i - M)\,(M_i - M)^T \,. \tag{3.19}$$

LDA is aimed at finding a linear transformation from p-dimensional X to an m-dimensional Y ($p \gg m$), which is expressed by

$$Y = A^T X. \tag{3.20}$$

In order to formulate class separability, the typical criterion is defined as $J = tr\left(S_{wy}^{-1} S_{by}\right)$ in m-dimensional subspace, where large between-group deviances and small within-group deviances make it large [Fukunaga, 2013].

To maximize $J = tr(S_{wy}^{-1} S_{by}) = tr\left\{(A^T S_{wx} A)^{-1}(A^T S_{bx} A)\right\}$, A must satisfy

$$\frac{\partial J}{\partial A} = 0 \quad \Rightarrow \quad (S_{wx}^{-1} S_{bx})A = A(S_{wy}^{-1} S_{by}). \tag{3.21}$$

Two matrices S_{by} and S_{wy} can be simultaneously diagonalized to D_m and I_m by a linear transformation,

$$B^T S_{by} B = D_m \quad and \quad B^T S_{wy} B = I_m, \tag{3.22}$$

where B is an $m \times m$ nonsingular matrix and B^{-1} is assumed to exist.

The criterion value is invariant under this nonsingular transformation B:

$$\begin{aligned} & tr\left\{\left(B^T S_{wy} B\right)^{-1}\left(B^T S_{by} B\right)\right\}=tr\left(B^{-1} S_{wy}^{-1} B^{-T} B^T S_{by} B\right) \\ & =tr\left(S_{wy}^{-1} S_{by} B B^{-1}\right)=tr\left(S_{wy}^{-1} S_{by}\right). \end{aligned} \tag{3.23}$$

Using (3.21), (3.22) may be written as:

$$(S_{wx}^{-1}S_{bx})(AB) = (AB)D_m. \tag{3.24}$$

Equation (3.24) shows that the components of D_m and the column vectors of (AB) are the m eigenvalues and eigenvectors of $S_{wx}^{-1}S_{bx}$.

Since the trace of a matrix is the summation of the eigenvalues,

$$\begin{aligned} J(p) &= tr(S_{wx}^{-1}S_{bx}) = \lambda_1 + \lambda_2 + \cdots + \lambda_p, \\ J(m) &= tr(S_{wy}^{-1}S_{by}) = d_1 + d_2 + \cdots + d_m, \end{aligned} \tag{3.25}$$

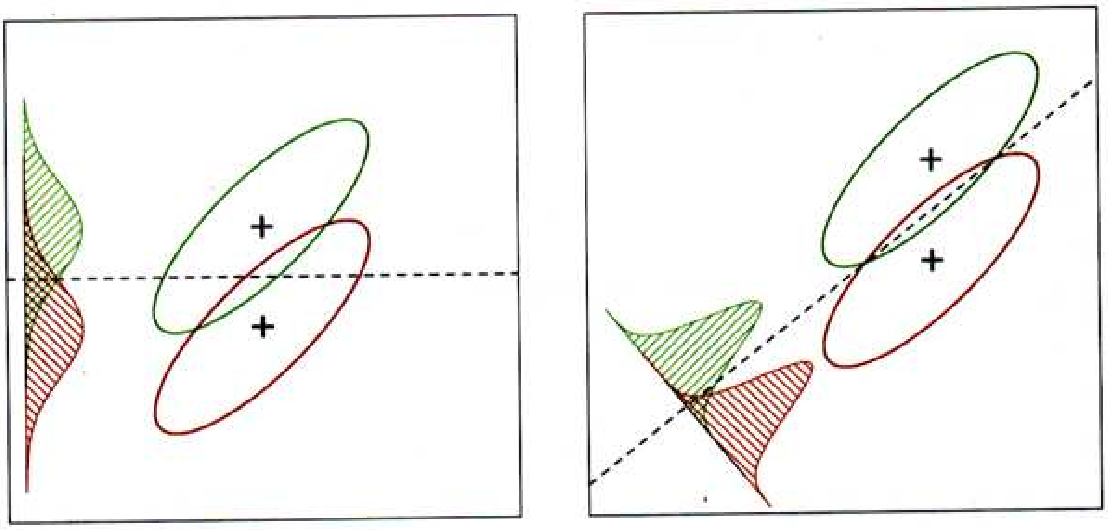
and $d_1, d_2, \ldots, d_m$ are also the eigenvalues of $S_{wx}^{-1}S_{bx}$, we can maximize J(m) by selecting the largest m eigenvalues and the corresponding m eigenvectors of $S_{wx}^{-1}S_{bx}$ to form the transformation matrix.

3.3 K-Nearest Neighbours (KNN)

K-nearest neighbours (KNN) is a nonparametric method used for classification. The single nearest neighbour technique, i.e. $k = 1$, is the simplest method of all. Nonetheless, other simple rules exist which have good statistical properties for various statistical tasks, such as estimation, prediction, and classification.

Let the data be (x, y), where x is the variable representing pixel values, y is the label. Reorder the data according to distances from x. We write $\left(x_{[n]}, y_{[n]}\right)$ for the nth reordered

Figure 3.1 LDA transforms the data to maximize between-class scatter and minimize within-class scatter.



Source: https://www.mathworks.com/matlabcentral/fileexchange/30779-lda--linear-discriminant-analysis-?focused=5183374&tab=function

data point with respect to x. And $d(\cdot)$ is a distance function to be defined that should have some properties, such as non-negativity, symmetry, the triangle inequality:

$$d\left(x, x_{[1]}\right) \leq d\left(x, x_{[2]}\right) \leq \cdots \leq d\left(x, x_{[n]}\right). \tag{3.26}$$

The nearest neighbourhood is

$$g(x) = y_{[1]}\left(x\right), \tag{3.27}$$

x is classified by assigning the label of the nearest data point to x.

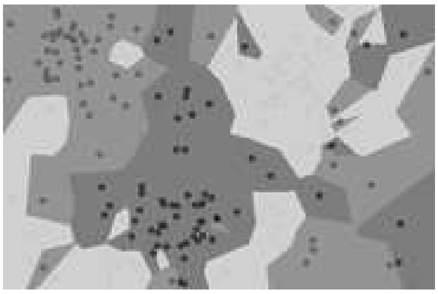
By considering more than just a single neighbour, we can obtain the rule for the K-nearest neighbours. x is classified by a majority vote of its neighbours, i.e.

$$g\left(x\right) = mode\left(y_{[1]}\left(x\right), y_{[2]}\left(x\right), \ldots, y_{[n]}\left(x\right)\right). \tag{3.28}$$

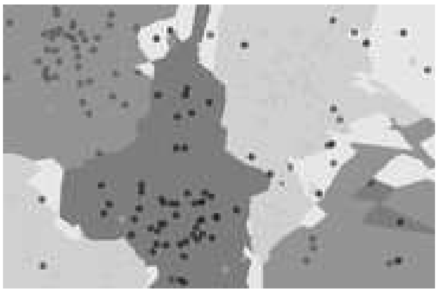
In this thesis, we will use Euclidean distance measurement. For PCA and LDA, after reduction of dimensionality the Euclidean distances between training and test are measured, and then we apply the 1-NN rule for recognition. For the KNN method, we will

Figure 3.2 One Example of Decision Boundary: 1-NN rule results in a complicated decision boundary and the 5-NN's decision boundary is simpler.

(a) The dataset



(b) The 1-NN Classification Map



(c) The 5-NN Classification Map

Source: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

directly measure the distances between test and training images, then apply the KNN rule. If there are multiple modes, for example, if the KNN method has two modes for a test image, then we will drop to the (K-1)-NN rule for recognition.

Euclidean distance between two vectors x,z of p dimensionality:

$$d(x, z) = \sqrt{(x_1 - z_1)^2 + (x_2 - z_2)^2 + \cdots + (x_p - z_p)^2}. \tag{3.29}$$

3.4 Method Comparison

Among the three methods, KNN is the easiest to be implemented and understood, with the intuition that subjects with nearest distance are from the same class with high probability. However, there is no theoretical guarantee for the optimal selection of k. The value of k is usually selected by maximizing accuracy with respect to the training data. Theoretically, PCA reaches the dimensionality reduction target and at the same time it retains the variation of variables as much as possible, which is reliable. Moreover, PCA finds the eigenvectors of WW^T instead of W^TW, which makes the method practical and efficient. The dimensionality can be reduced to no more than the number of training data. If the training size is too small, say not more than 10, it may not be an ideal method. A similar criticism applies to KNN in that the selection of the projection matrix depends on the training data. LDA is also designed to reduce dimensionality, but unlike PCA, its motivating principle is to find the projection matrix which maximizes the intergroup variation, and makes within-group variation as small as possible. Thus the training data must have two or more observations within each class. The projection selection also depends on the training data. However, to perform operations on large matrices, which are usually larger than $1e4 \times 1e4$, is time-consuming, and sometimes hard to be computed. The difficulty lies in creating advanced algorithms for computing inverses and singular value decompositions for large matrices. Another approach to this problem is to combine several methods. In this thesis, we combine PCA and LDA to achieve our goals. First we utilize PCA to obtain a comparatively small dimensional space, but which is still large enough for accuracy, and then we apply LDA.

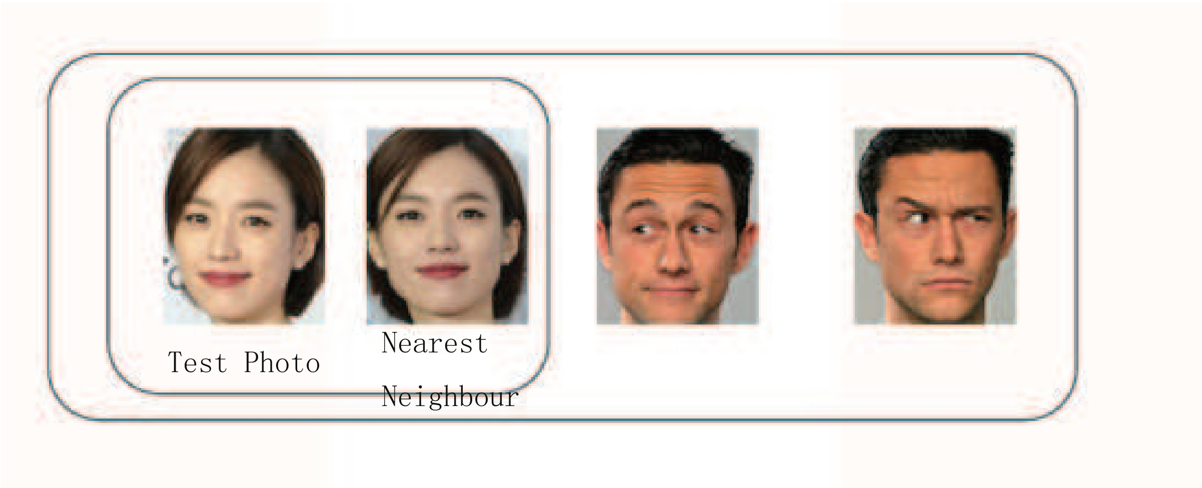


Figure 3.3 One Example of Results of different KNN: Test photo is correctly assigned with the 1-NN rule, but incorrectly assigned with 3-NN.

Chapter 4

Results

4.1 Recognition Performance with Increasing Training Size

Table 4.1 Running Time on ORL Database

	PCA Reduced to dim 13	LDA Reduced to dim 30	1-NN	3-NN
Time	12.5250	70.9384	4.4231	7.0387

Each of three methods were run 10 times on the ORL database under equal conditions. Within each iteration, we respectively selected 40, 80, ..., 360 images (1, 2, ..., 9 images of each subject) to be training data and the remaining images (360, 320, ..., 40 images) to be testing data. The training sample size is increasing by one for each iteration, but the selected images to be tested are changed between each run.

The accuracy is computed as (correctly recognized total) / total.

Table 4.2 Correctly Recognized Total on ORL
1 test $\times$ 40 subjects

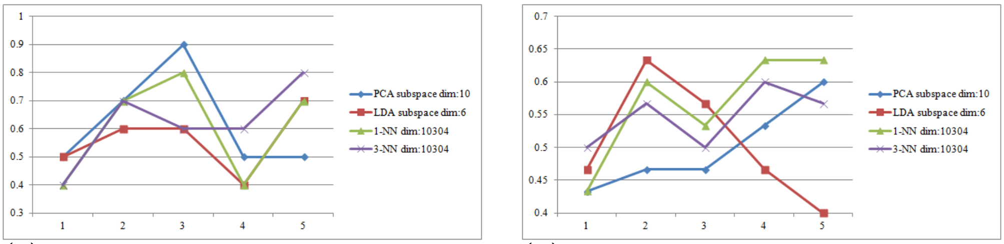
	Select the ith image of each subject as test										
	10th	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	standard deviation
PCA	36	37	37	39	40	39	39	39	37	38	1.28668
LDA	38	39	39	40	38	40	37	37	36	36	1.49071
1-NN	37	39	40	40	39	39	40	39	39	39	0.87560
3-NN	37	38	39	39	38	38	39	40	39	37	0.96609

We also assessed the three methods by running 5 iterations on the Self-made dataset. Within each iteration, 10, 20[1], 30, 40 images were respectively selected as training. The algorithms' performance on the Self-made data is not as good as their performance on the ORL data. Moreover, the recognition accuracy has a bigger deviance between each run compared to the ORL database. That is, the recognition performance differs greatly when different images are selected, even with the same training size.

Table 4.3 Correctly Recognized Total on Self-made
1 test × 10 subjects

	Select the ith image of each subject as test					
	5th	1st	2nd	3rd	4th	standard deviation
PCA	5	7	9	5	5	1.78885
LDA	5	6	6	4	7	1.14018
1-NN	4	7	8	8	7	1.87083
3-NN	4	7	6	6	8	1.48324

Figure 4.1 Recognition Accuracy on Self-made Data: 5 runs with subjects' different images selected as test



(a)
Select 1 image of each subject as test set

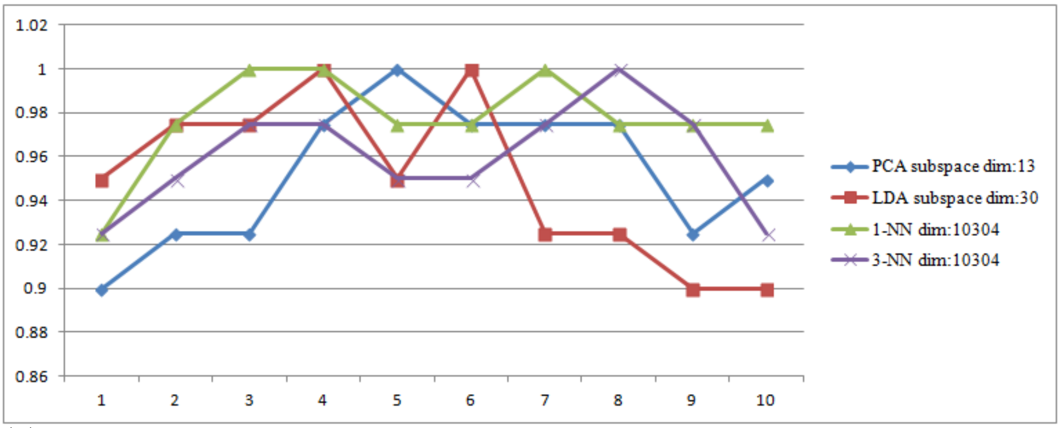
(b)
Select 3 images of each subject as test set

For the ORL database, when the test number is 1 for each subject (and 9 training images for each subject), PCA, LDA, and KNN all perform well. They all exceed 90% accuracy, which means out of 40 subjects, at least 36 subjects are correctly classified. PCA reduces to 13 dimensional space, and LDA reduces to 30, which is enough to carry enough information compared to the original 10,304-dimensional space. When the test set is enlarged (and the training set shrinks), 1-NN performs most efficiently with the shortest operation time and the best accuracy. Although LDA takes more time, it performs better

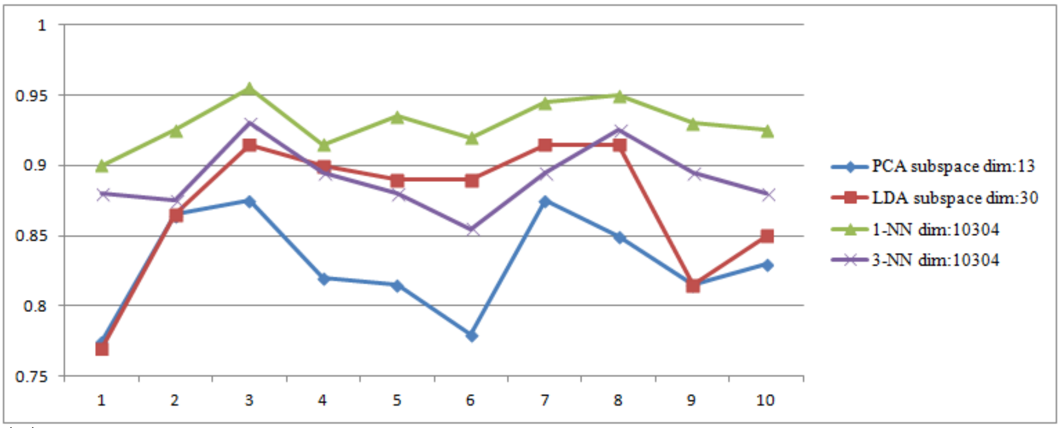
[1]There must be at least 2 images for each subject as training set for the LDA method, so the training size starts from 20 on Self-made data, 80 on ORL data.

than PCA when the training set is small.

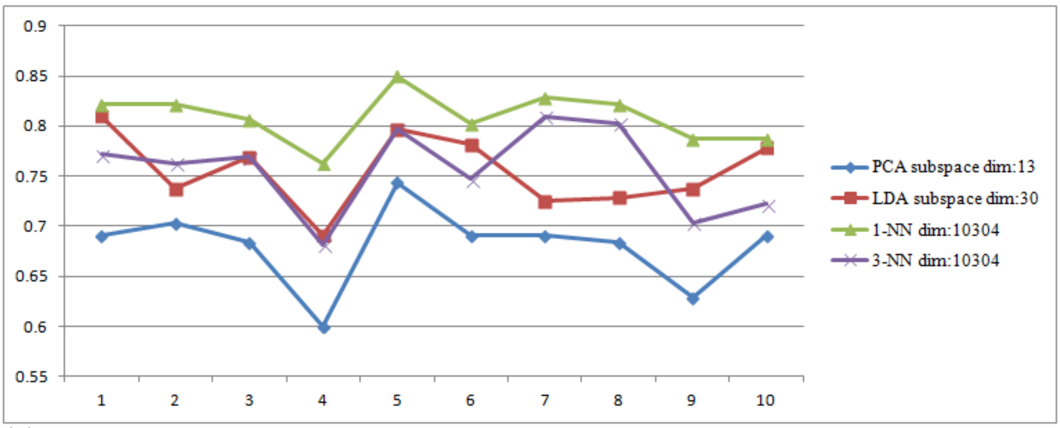
Figure 4.2 Recognition Accuracy on ORL Data: 10 runs with subjects' different images selected as test



(a)
Select 1 image of each subject as test set.



(b)
Select 5 images of each subject as test set

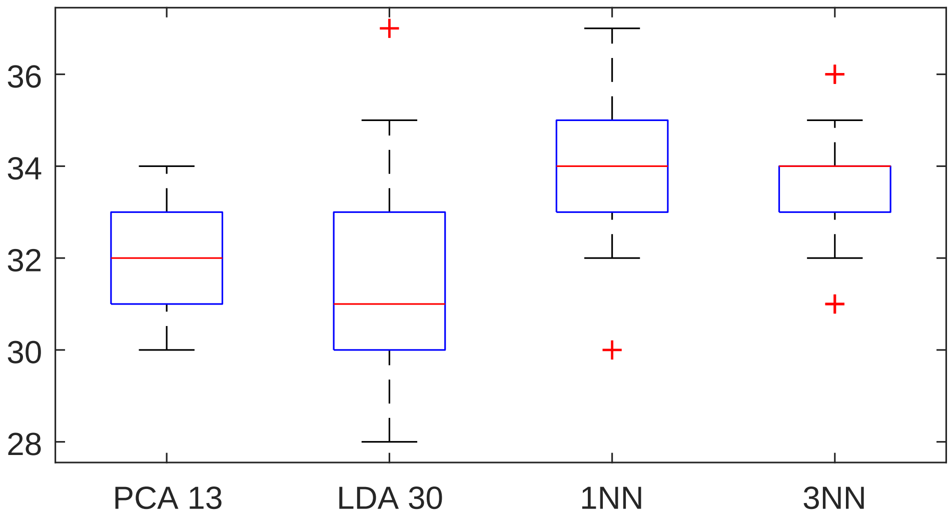


(c)
Select 8 images of each subject as test set

4.2 Recognition Performance with Fixed Training Size

In this section, we ran each method 50 times on the ORL dataset with a fixed test set of size 40 (every subject selects the 10th image as test). In each run, the subjects may have different numbers of images selected in the training set, but the total training size is fixed at 200.

Figure 4.3 Boxplot of Correctly Recognized Total out of 40 tests



The 3-NN method has a median of 34 correctly recognized subjects and is the least variable. The 1-NN method performs similarly as the 3-NN method but has several runs of higher accuracy than the 3-NN. LDA has the lowest median (31), and it also has the greatest variability.

For a certain test subject with low accuracy in recognition by one method, the other methods also don't perform well. 1-NN, 3-NN and PCA have similar patterns, but PCA's (reduced to 13 dimensional space) recognition performance is not as good as KNN with training size of 200. LDA can correctly identify when other methods misidentify in several runs, but still makes more misclassified decisions than other methods.

For the 10th image of Subject 10, 1-NN and 3-NN classify it as Subject 3, 4, 8, and 38.

Figure 4.4 Recognition Result of 40 Subjects of ORL

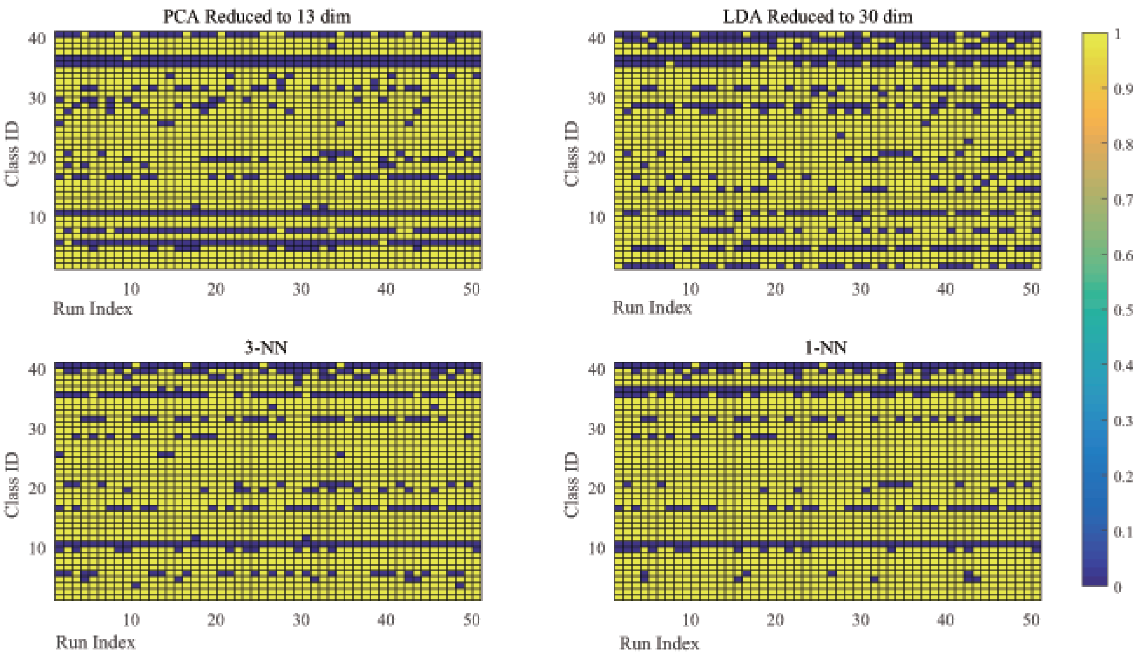
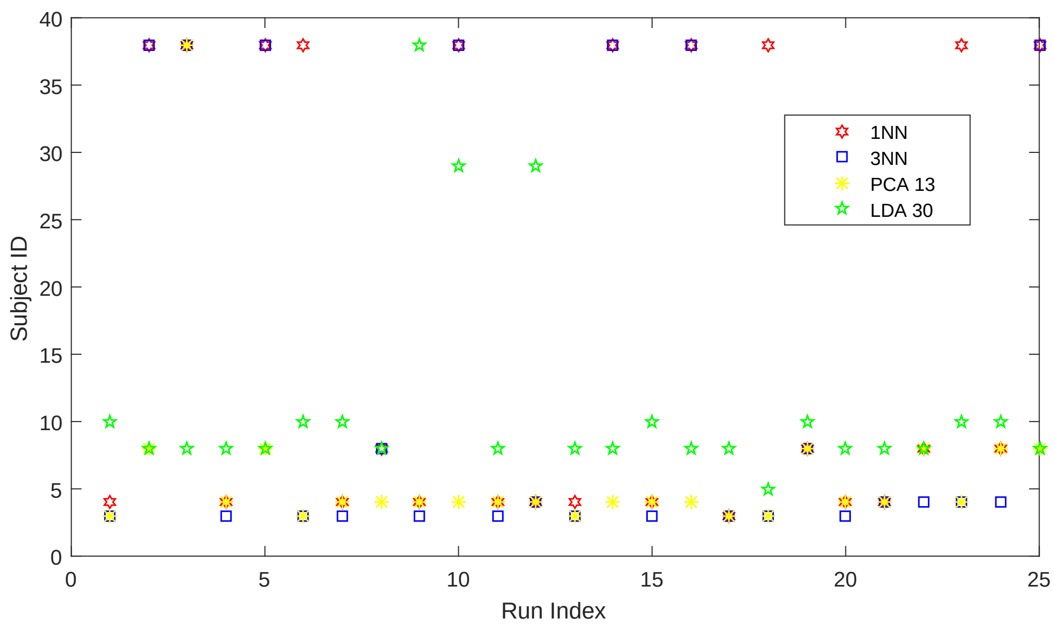


Figure 4.5 Recognition Result for the 10th Image of Subject 10 as Test in 25 Runs



PCA classifies it as 3, 4, 8, 36, 38. The methods give wrong classifications for all 50 runs. LDA classifies it as 5, 8, 10, 19, 21, 29, 38, which shows LDA is different than the other methods, in that it is more sensitive to the selected training images.

4.3 Further Comparison

In this section, we ran 50 times with fixed training size of 30 on every subjects' images (10 subjects $\times$ 5 images) of Self-made data. Between each iteration, the selected training images are different.

Taking the first, second or fifth image of Subject 6 as a test sample, the KNN method is not able to recognize the subject, but has high accuracy (above 85%) on the third and fourth image. However, LDA can correctly recognize the images for which KNN fails. Overall, while LDA may have the lowest accuracy, it is more stable with respect to different partitions of the dataset into training and test sets, compared to the other methods. That is, it has the smallest standard error of the accuracy rate for each subject.

Figure 4.6 Self-made Data Images

For Subject 7 and 8, all methods perform very well, but not for Subject 2.

Table 4.4 PCA (Dim 10) Correctly Recognized Total out of 50 Runs

		Subject ID									
		1	2	3	4	5	6	7	8	9	10
ith image as test	1	50	0	49	39	13	0	50	50	44	45
	2	50	14	41	37	0	41	49	39	38	50
	3	33	0	24	0	17	0	50	50	0	50
	4	12	0	0	48	8	3	38	50	39	50
	5	50	0	50	2	0	39	50	50	0	0
total		195	14	164	126	38	83	237	239	121	195
standard deviation		16.79	6.26	21.09	22.49	7.64	21.41	5.27	4.92	22.21	21.91

Table 4.5 LDA (Dim 6) Correctly Recognized Total out of 50 Runs

		Subject ID									
		1	2	3	4	5	6	7	8	9	10
ith image as test	1	42	0	19	23	43	21	49	36	10	2
	2	44	0	25	35	38	2	27	45	2	45
	3	20	0	0	11	40	9	49	39	42	15
	4	16	0	0	30	22	31	50	47	28	48
	5	18	0	40	44	4	26	50	50	12	1
total		140	0	84	143	147	89	225	217	94	111
standard deviation		13.78	0.00	17.14	12.46	16.37	12.03	10.08	5.77	16.04	22.88

Table 4.6 3-NN Correctly Recognized Total out of 50 Runs

		Subject ID									
		1	2	3	4	5	6	7	8	9	10
ith image as test	1	50	0	50	0	50	0	50	50	43	40
	2	50	0	50	12	0	0	50	50	28	50
	3	50	0	0	5	38	43	50	50	13	50
	4	50	0	0	50	33	42	50	50	50	50
	5	50	0	40	0	0	0	50	50	0	0
total		250	0	140	67	121	85	250	250	134	190
standard deviation		0.00	0.00	25.88	21.04	22.94	23.28	0.00	0.00	20.68	21.68

Table 4.7 1-NN Correctly Recognized Total out of 50 Runs

		Subject ID									
		1	2	3	4	5	6	7	8	9	10
ith image as test	1	50	0	47	39	50	0	50	50	50	13
	2	50	0	50	37	35	0	50	50	35	50
	3	12	0	0	9	38	43	50	50	12	16
	4	12	0	0	50	33	50	50	50	50	50
	5	50	0	50	0	0	0	50	50	0	0
total		174	0	147	135	156	93	250	250	147	129
standard deviation		20.81	0.00	26.87	21.37	18.65	25.59	0.00	0.00	22.62	22.90

Chapter 5

Conclusion

The statistical facial recognition methods PCA, LDA and KNN work very differently from how a human would perform a recognition task. In this paper, the algorithms all work well on some images which are distinguishable from others, but each lacks the ability to capture facial features and small details. When similar images are together, females may even be classified as males. More complex algorithms and increased precision are needed to improve recognition accuracy. Another approach is geometric algorithms investigating the relationship of mouth, eyes, brows etc., which imitate a human's identifying features.

This thesis applies these methods to the ORL database, which yields 10,304 variables. Our results show that the KNN method has the best overall accuracy. When the training set's size rises above 300, PCA, which reduces the dimension from 10,304 to 13 can almost be as good as KNN, which keeps all the information.

From the perspective of operation, KNN is always shorter than PCA and LDA. The computational complexity of computing distances of a large matrix is smaller than computing eigenvalues of a relatively small matrix. But in practice, when the image data has hundreds of thousands of variables, PCA is good for space saving and efficiency.

The 3-NN method does not outperform the 1-NN method, sometimes 3-NN misclassifies while 1-NN correctly classifies.

Among PCA, KNN, LDA, LDA is the most different. It's more sensitive to the changing of training images, and can perform better when KNN and PCA all fail. It's less sensitive to the image changes of test subjects, compared to the result that KNN can do very well on certain test images, and will perform poorly on other test images from the same subject. Due to the limitation of the LDA algorithm for high dimensional matrices, we first reduce to a relatively small space, and then perform LDA. This reduces the accuracy of the LDA procedure. A better algorithm of computing eigenvalues and eigenvectors of large matrices may be helpful to improve the performance of LDA. In a related paper, Navarrete and Ruiz-del-Solar [Navarrete and Ruiz-del Solar, 2002] claim that LDA outperforms PCA on all tasks in their tests.

References

[Delac et al., 2005] Delac, K., Grgic, M., and Grgic, S. (2005). Independent comparative study of pca, ica, and lda on the feret data set. *International Journal of Imaging Systems and Technology*, 15(5):252–260.

[Fukunaga, 2013] Fukunaga, K. (2013). *Introduction to statistical pattern recognition*, pages 445–450. Academic Press.

[Gross, 2011] Gross, R. (2011). Face databases. In *Handbook of Face Recognition*, page 301. Springer.

[Kim, 1996] Kim, K. (1996). Face recognition using principle component analysis. In *International Conference on Computer Vision and Pattern Recognition*, pages 586–591.

[Navarrete and Ruiz-del Solar, 2002] Navarrete, P. and Ruiz-del Solar, J. (2002). Analysis and comparison of eigenspace-based face recognition approaches. *International Journal of Pattern Recognition and Artificial Intelligence*, 16(07):817–830.