Further insights into subspace methods with applications in face recognition

Zhu, Yan

2009

Zhu, Y. (2009). Further insights into subspace methods with applications in face recognition. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/15161>

<https://doi.org/10.32657/10356/15161>

FURTHER INSIGHTS INTO SUBSPACE METHODS WITH APPLICATIONS IN FACE RECOGNITION

ZHU YAN

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2009

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

Date
.....

Zhu Yan
.....

*To Ying De, Bing Xin, Chen Yu,
Mom and Dad
for their love and support.*

Acknowledgment

I would like to express my sincere gratitude to my supervisor, Prof. Eric Sung, for his guidance, encouragement and support in every stage of my Ph.D study. I could count on him to provide guidance and insight whenever I was stuck. Moreover, he also allowed me considerable freedom to explore various topics and to develop my abilities in research and problem solving. His knowledge, kindness, and vision as a professional researcher have provided me lifetime benefits.

I am grateful to Prof. Chan Kap Luk, Prof. Chua Chin Seng and Prof. Mao Kezhi for providing the enlightening discussions and comments with their abundant knowledge to improve my works. I would also like to thank my colleagues, Wang Lei, Li Xuchun and Zhang Wenbo, for the joyful discussions and study groups as well as their supports and encouragements on my research.

Special thanks to Yu Dan, Cai Zhi and Liu Bibo for their friendships and countless help during the long period of my graduate studies (in both full-time stage and part-time stage). I would also like to express my appreciation to Li Jiang, Wang Yingjie and Ren Ying for their kind help and enjoyable discussions in a wide range. I would like to thank all my fellow graduate students and technicians in the Machine Intelligence Research Lab. My memorable days of life in Nanyang Technological University can never be complete without them.

Last but not least, I am so grateful to my families for their love and continuous support over these years of my full-time and part-time studies. Thanks to my husband, Ran Yingde, for his encouragement and help that are always there whenever I need them the most. Thanks to my Dad and Mom, not only for their care and nurturing for myself, but also for their great help in nurturing my young boys. Without their help, I could never have time to accomplish this thesis. Lastly, I must thank two special little persons – my first son, Ran Bingxin, and my second son, Ran Chenyu, both arrived during my part-time Ph.D study. Thank you for being such nice and cute babies. In fact, you are the true motivation for mommy to complete the last part of this research, despite of the busy full-time work schedule and the inconvenience caused by my second pregnancy. I dedicate this thesis to my families.

Contents

Acknowledgements	i
Contents	ii
Summary	v
List of Figures	vii
List of Tables	x
List of Abbreviations	xi
1 Introduction	1
1.1 Motivation	3
1.2 Objectives	4
1.3 Thesis Contributions	5
1.4 Thesis Organization	5
2 Background	7
2.1 Machine-Based Face Recognition	7
2.1.1 Challenges Still Remaining in Existing Face Recognition Re- search	8
2.1.2 Review of Machine-Based Face Recognition Techniques	9
2.2 Subspace Methods for Statistical Pattern Classification	12
2.2.1 Principal Component Analysis (PCA)	13
2.2.2 Linear Discriminant Analysis (LDA)	14
2.2.3 Feature Selection	15
2.2.4 Classifier Design and Support Vector Machines	16
3 Margin-Maximization Discriminant Analysis	19
3.1 Introduction	19
3.2 The MMDA Formulation	21
3.3 Direct Dimension Reduction	26
3.3.1 The Computation Trick for MMDA	28
3.4 Margin Improvement via Subspace Approach	29

3.5	Non-Linear MMDA	30
3.6	Experiments	33
3.6.1	A Toy Data Example	33
3.6.2	Experiments on ORL, UMIST and YALE Face Databases	33
3.7	Conclusions	40
4	Relevance-Weighted Discriminant Analysis	42
4.1	Introduction	42
4.2	The Estimation of Non-Overlapping Class Regions	44
4.2.1	Introduction	44
4.2.2	The Geometric Meaning of λ	45
4.2.3	Modeling the Non-Overlapping Class Regions	48
4.3	Feature Relevance	51
4.4	Relevance-Weighted Discriminant Analysis	53
4.4.1	The Framework of RWDA	53
4.4.2	wLDA - a Preliminary Version of RWDA	54
4.5	Justification of Relevance Weightage	55
4.6	Experiments	59
4.6.1	LDA versus wLDA	59
4.6.2	RWDA Performance on the UMIST Face Database	61
4.6.3	RWDA Performance on the ORL Face Database	64
4.6.4	RWDA Performance on the YALE Face Database	64
4.6.5	Discussions	70
4.7	Conclusion	72
5	Investigations on 2D Subspace Methods	73
5.1	Introduction and Motivation	73
5.2	Objectives and Experiment Structure	75
5.2.1	Experiment Structure	76
5.3	Derivations and Experiments	76
5.3.1	The Covariance Face Method	76
5.3.2	The Error Face Method	81
5.3.3	The Effect of Image Size	84
5.3.4	The Discriminant Covariance Face Method	87
5.3.5	Feature Space Expansion with Image Stacking	89
5.3.6	SVD Basis for Facial Pose Estimation and Face Recognition	91
5.4	Conclusions	96
6	Conclusions and Recommendation	100
6.1	Conclusions	100
6.2	Future Works	102
	Author's Publications	104

Bibliography

105

Summary

Subspace methods were shown to be very efficient in reducing data dimension and extracting statistical features. Over the years, it is widely applied to multi-class pattern classification problems, such as face recognition, which often involves high dimensional data and large data set. With the advances in machine learning theory and the renewed interests in large-scale pattern classification problems, we re-visit the subspace methods to apply new concepts from recent classifier design and resolve some prolonged performance issues. The work in this thesis provides further insights into the subspace methods, with special interests in face recognition applications.

Firstly, we introduce a geometric margin measure into the subspace methods, and propose the Margin-Maximization Discriminant Analysis (MMDA). It is based on an additive-form discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$, and we prove that it can approximately maximize the average margin between the classes with the help of a class-spread regulator β . This new formulation is computationally more stable than the conventional LDA which involves a quotient-form discriminant function. It naturally avoids LDA's singularity problem of S_W , and hence, can achieve direct dimension reduction of the input data in the discriminant domain. A computational trick for MMDA is proposed to make it more effective when data dimension is much larger than the number of the samples. The nonlinear formulation of MMDA is also derived using the "kernel trick". Experiments with three face databases show that MMDA can find feature sets with larger average margin between the classes as compared to LDA, and it often achieves better classification results.

Secondly, we look into the "peaking phenomenon" and the overfitting problem that is persistent with LDA and its variants. We proposed a simple but novel feature relevance weighting scheme, called Relevance-Weighted Discriminant Analysis (RWDA), to resolve these issues. Our definition of feature relevance is based on the amount of non-overlap between the classes, which can be analytically computed based on Fisher discriminant values and a constant spread regulator M . The effectiveness of the proposed relevance weightage is theoretically justified. Our intensive experiments with three popular face databases show that RWDA can completely eliminate the peaking phenomenon of LDA, and also demonstrates better generalization for classification. This analysis also suggests a new insight into the root cause of overfitting for classifiers using distance metric.

Finally, we look into the design of 2D subspace methods that uses 2D image representation to overcome the computation intractability for problems of high dimension and very large data set. A series of investigations have been carried out and

various 2D subspace methods have been proposed based on our framework called the Spectral-Face Analysis. Our general approach is to use SVD to compute 2D projection basis and therefore compute projection vectors of each image. Although these methods are giving slightly inferior performances as compared to the traditional subspace methods, they predate some of the recently successful 2D subspace methods and show some interesting findings.

List of Figures

3.1	The Projected Margin between Two Classes on a Feature Direction $\mathbf{w}$	22
3.2	Function Values of $ a - b $.	23
3.3	Function Values of $ a ^2 - b ^2$.	24
3.4	PCA Dimension Reduction May Lose Discriminant Information.	27
3.5	Margin obtained by MMDA, SVM and LDA - toy data	33
3.6	Sample Images from ORL, UMIST and YALE face databases.	34
3.7	Recognition performance on ORL Database	35
3.8	Recognition performance on UMIST Database	36
3.9	Recognition performance on YALE Database	36
3.10	Performance of MMDA and LDA at varied feature size - ORL Database	39
3.11	Performance of MMDA and LDA at varied feature size - UMIST Database	40
3.12	Performance of MMDA and LDA at varied feature size - YALE Database	41
4.1	Projected Class Mean and Standard Deviation on LDA Feature Vector	46
4.2	Typical values of λ_k , Δ_k and Ω_k for UMIST face database (8 images/class trained).	47
4.3	Typical values of λ_k , Δ_k and Ω_k for ORL face database (5 images/class trained).	48
4.4	Non-Overlapping Regions on a Feature $\mathbf{w}_k$ for a simple two-class problem. (Assume class distribution is i.i.d. and the spread is $2\sigma_C$.)	50
4.5	Non-Overlapping Regions for non-i.i.d. two-class problem (The overlapped case).	51
4.6	Regions of High Probability of Classification Error for Overlap Data.	55
4.7	Loci of Equal Choice for Unweighted and Weighted Features.	56
4.8	NNC in the Narrow Region B with and without Relevance Weightings.	57
4.9	Sample u in B is Just at the A-B Boundary.	58
4.10	Sample u at the A-D Boundary	59
4.11	Classification Performance of wLDA v.s. LDA and PCA on UMIST Database	60
4.12	Classification Performance of wLDA v.s. LDA and PCA on ORL Database	60
4.13	Classification Performance of wLDA v.s. LDA and PCA on YALE Database	61

4.14	Classification Results of RWDA Compared to Other Methods - UMIST Test A (8 samples/class trained)	62
4.15	Classification Results of RWDA Compared to Other Methods - UMIST Test B (5 samples/class trained)	63
4.16	Classification Results of RWDA Compared to Other Methods - ORL Test A (5 samples/class trained)	66
4.17	Classification Results of RWDA Compared to Other Methods - ORL Test B (4 samples/class trained)	67
4.18	Classification Results of RWDA Compared to Other Methods - ORL Test C (3 samples/class trained)	68
4.19	Classification Results of RWDA Compared to Other Methods - YALE Test A (8 samples/class trained)	69
4.20	Classification Results of RWDA Compared to Other Methods - YALE Test B (5 samples/class trained)	70
5.1	Example of First 18 2D Projection Bases for the Covariance Face Method.	78
5.2	Reconstruction with the 2D Projection Bases for the Covariance Face Method (a) original image, M (64×64 pixels); (b) covariance face, $(M - \bar{M})(M - \bar{M})^T$, (c)(d)(e) reconstructed covariance face using the first 5, 30, 64 features respectively.	78
5.3	Performance of PCA at Different Feature Size.	79
5.4	Performance Deviation of PCA at Different Feature Size.	80
5.5	Performance of Covariance Face Method at Different Feature Size. . .	80
5.6	Performance Deviation of Covariance Face Method at Different Fea- ture Size.	81
5.7	Error Face, $C = E_i\{M_i - \bar{M}\}$, from a computer.	83
5.8	Example of First 18 2D Projection Basis for the Error Face Method. .	84
5.9	Performance of Error Face Method at Different Feature Size.	85
5.10	Performance Deviation of Error Face Method at Different Feature Size.	85
5.11	Performance of Various Methods over Different Image Size (NN Clas- sifier).	86
5.12	Performance of Various Methods over Different Image Size (NFL Clas- sifier).	86
5.13	Circulant Stacking of a Face Image. (a) Original image ($m \times n$ pixels); (b) $2m \times 2n$ stacking by circulant expansion; (c) schematic description of the transformation from (a) to (b).	90
5.14	Performance of the Covariance Face Method with $4m \times 4n$ Circulant Stacking.	91
5.15	Singular Value Decomposition (SVD) on a Face Image.	92
5.16	Average Face and its Singular Values s_k	93
5.17	$\mathbf{u}_k$ of Average Face (First 17 and Last 3).	93
5.18	$\mathbf{v}_k$ of Average Face (First 17 and Last 3).	94
5.19	Face Images and the Corresponding $\mathbf{u}_1$ and $\mathbf{v}_1$	95

5.20	Face Images and the Corresponding $\mathbf{u}_4$ and $\mathbf{v}_4$	95
5.21	Face Images and the Corresponding Vertical Face Profiles ($\mathbf{u}_1$) - Sample Group A.	97
5.22	Face Images and the Corresponding Vertical Face Profiles ($\mathbf{u}_1$) - Sample Group B.	98

List of Tables

3.1	Classification Accuracy (%) of MMDA and LDA at the Highlighted Points of Fig. 3.7 to Fig. 3.9	38
3.2	t Test to compare MMDA($\beta = 9$) and LDA(K_{pca}^*) at the optimal feature size	38
4.1	Best Performance of RWDA- L_2 and LDA- L_2	65
4.2	Performance Deviation at the Peak Points of RWDA- L_2 and LDA- L_2	65
4.3	t Test to compare the peak performance of RWDA- L_2 and LDA- L_2	65
4.4	The position of critical point T for RWDA	71
5.1	Enhancement with Class-Discriminant Analysis.	89

List of Abbreviations

ABBREVIATIONS	FULL EXPRESSIONS
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
FDA	Fisher Discriminant Analysis
ICA	Independent Component Analysis
KPCA	Kernel-Principal Component Analysis
KDA	Kernel-Discriminant Analysis
DCT	Discrete Cosine Transformation
WT	Wavelet Transform
MDS	Multidimensional Scaling
LLE	Locally Linear Embedding
LPP	Locality Preserving Projections
MMC	Margin-Maximization Criterion
MMDA	Margin-Maximization Discriminant Analysis
RWDA	Relevance-Weighted Discriminant Analysis
wLDA	weighted Linear Discriminant Analysis
SRM	Structural Risk Minimization
SVM	Support Vector Machine
SMO	Sequential Minimal Optimization
SVD	Singular Value Decomposition
SV	Singular Value
<i>std</i>	Standard Deviation
RP	Random Projection

Chapter 1

Introduction

A large emphasis of science and engineering research is in the analysis of large amount of experimental data. The great challenge is to find a simple rule, law or formula that can explain the massive data collected. In the field of machine learning, the prevalent methodology is to learn from training examples for classification of objects, regression analysis or to discover new knowledge. In some applications, the training data is of high dimensions and is a very large set.

In this thesis, only classification is considered. The key issue here is to find compact representations (features) from the high dimensional data that are deemed the best for the classification tasks. These chosen features must perform well on unseen data (generalization) while avoiding at the same time the pitfalls of overfitting and underfitting. Moreover, when the number of training samples is very large, a physical (memory/storage) and time constraint arise.

How can a classifier learn from so much data with a conventional computer? An upper bound is reached for time and storage for conventional learning algorithms. Support Vector Machines (SVM) [114], requiring quadratic programming techniques, are computationally expensive. An SVM extension, called the SMO, is devised by [108] that enables the quadratic programming approach to be tractable by iteratively solving for two points at a time. But even SMO cannot solve for the optimal hyperplane solution if the dimension is too large or if the data set is very large. Thus, in recent years, there is concentrated effort to find methods that can handle both large datasets and high feature dimensions. Therefore, two main issues are to reduce the dimension of the input space and to find the best features in this reduced dimension space for the task at hand. This thesis will address these two major problems. Furthermore, the insights and methodologies that we will propose will primarily be focused on one very important application area which involves high dimension and very large data set. This is Face Recognition.

Face recognition has become an active research area in the past three decades owing to the high demand for robust, non-intrusive biometric inspection systems [44]. It is generally considered as a very difficult multi-class pattern classification problem that often involves high dimensional data and probably large data set.

To save the computation power and the memory requirements of the computing

system, and also to avoid the curse of dimensionality that could affect the accuracy of the parameter estimation due to insufficient samples [6], a small number of the most useful features have to be extracted from the high dimensional data before feeding the samples to matching or classification stage. This process is usually called dimension reduction/feature extraction, and there are numerous methods to achieve this. Some take the supervised approach that utilize the class identity information of the data, while some do not (the non-supervised approach). Prior information about input data can also be applied at various stage of the dimension reduction. Particularly, we are more interested in the group of techniques that are motivated by the information theory such as the follows:

- subspace methods: such as principal component analysis (PCA) [131], Fisher/linear discriminant analysis (FDA or LDA) [120] [4] [92], independent component analysis (ICA) [98] [29], and their nonlinear (kernel transformed) versions, such as Kernel-PCA (KPCA) and kernel-LDA (KLDA/KDA) [141] [89]
- frequency or spatial-frequency domain transformations: such as discrete cosine transformation (DCT) [101] [71] [127] and wavelet transformation (WT) [7] [139] [84]
- manifold learning-based methods: such as multidimensional scaling (MDS) [22], locally linear embedding (LLE) [112] [53] and locality preserving projections (LPP) [27]

Among them, the subspace methods is perhaps the most popular and pioneering one, which reduce data dimension by finding the important projection directions (features) that optimizes certain statistical properties. For example, LDA can find features that maximizes between-class scatter and minimizes within-class scatter of the training samples, which has found many applications for classification problems. The features found this way are thus data dependent. Being different from the subspace methods, the (spatial-)frequency domain transformations extract lower dimensional image representations using data independent basis. They are widely used for image compression and as such can also be used in dimensionality reduction. Transformations like DCT and WT has been optimized for fast computation and minimal human perception error after reconstruction. However, they can potentially discard high frequency information that might be useful for classification.

The manifold learning-based methods, such as LLE and LPP, are more recently developed methods. They can discover intrinsic, non-linear low-dimensional manifold embedded in the high-dimensional Euclidean space, and hence are good for compact representations of high-dimensional data, especially for visualization and clustering purposes. However, compared to the subspace methods, some of the manifold methods are more dependent on the parameter settings to discover the underlying data structure. According to Ridder and Duin [27], LLE will always need at least one parameter controlling its nonlinearity. It is interesting to note that in one of our proposed methods (MMDA), a parameter is also introduced to enable it to search for the maximum margin or non-overlap. However, in our case, only one

parameter is needed. Moreover, the manifold methods may tend to overfit the data, leaving classification a more difficult task. One important consideration of manifold learning methods is the choice of weights for the features. Beikin and Niyogi [5] suggest a choice of either gaussian weights or equal weights for their Laplacian Eigenmap. Of course, there are other weightage schemes that have not yet been explored. How important is it to find a good if not the optimal set of weights? Can it be found? In our thesis, we answered this question for LDA in the affirmative.

There are some questions still not answered. So far, the proposed algorithms report finding manifold structures for the first two dimensions. But it is not known if such methods are capable of finding the geometrical structures that are embedded in dimensions higher than a few. Manifold learning nevertheless has interesting and important potentials in many fields of science and engineering. Roweis et. al. [112] remarked that “LLE is likely to be even more useful in combination with other methods in data analysis and statistical learning. For example, a parametric mapping between the observation and embedding spaces could be learned by supervised neural networks whose target values are generated by LLE”. Our thesis is focused on subspace learning methodology but it can be seen that the two approaches can complement one another.

As such, the traditional subspace methods are still widely applied in various classification problems, especially for multi-class problems where it has advantage over other paradigms like SVM and ADABOOST. Recently, it also gained renewed interests due to the development of nonlinear subspace methods using the “kernel tricks”. Driven by the advances in machine learning theory and the ever-demanding classification problem scales, it is still worth-while to re-visit the design of the subspace methods in an attempt to discover further insights and then to find better solutions to resolve some persistent problems. In this thesis, we provide further insight into the subspace methods, with special interest to face recognition applications.

1.1 Motivation

Given a set of data, what can one learn from it? This is one of the main objectives of machine learning. This scenario occurs in countless scientific investigations and engineering applications. More and more challenging applications are ventured into especially those with very high dimensional input data and large number of data, such as face and expression recognition, text classification, data mining, gene expressions and a host of others. This poses a problem to conventional classification design due to the intractability of the dimension reduction process. Some approaches have been offered such as sequential methods to converge to optimal solutions [69] [128] [130] [119]. For subspace methods, this has also been explored. In 2000, we set out to investigate the possibility of not vectorizing the image in image classification problems. The idea is to ensure that the method is independent of the number of data as the input feature dimension is kept constant within limits of either the row space and/or the column space of the image. This was our earliest foray into gaining

new insights into subspace classification methods.

In LDA and other classifiers, it has been reported that there exists a peaking phenomenon in which the classification performance is maximum when a certain number of features are used but it deteriorates when more features are included [31] [109] [59]. The common consensus is that it is caused by overfitting. Too many features implies that the classifier may have become over-expressive and begins to learn noise as well. We wonder why this has to be so, especially for the case of subspace methods. This motivates us to consider what makes a feature useful/relevant and whether we should treat each feature with equal importance. Our investigations led us to the hypothesis that the relevance of a feature should be measured by the amount of non-overlapping area of class distributions.

This is quite a general claim for discriminant analysis and it also leads us to consider how the relevance should be formulated. The conclusion of our research verified by experiments is a weightage scheme we call the Relevance Weightage.

Another crucial consideration in classifier design is that of the margin. SVM based on the structural risk minimization theory of Vapnik gives the optimal margin of separation between two linearly separated classes [134]. Many other classifiers are also designed with margin maximization as the criterion. These include variants of SVM [62] [113] [16] [35], ADABOOST [110] [124] [67], etc. It is known that present subspace methods (PCA and LDA) are not specifically designed to maximize the margin and deemed not able to do so since they are linear methods. We delved into this issue and have devised a novel nonlinear approach to LDA design that can approach the solution of optimal margin of separation. Moreover, we showed that margin is referred to only for separable classes. When the classes are non-separable, we should really be trying to maximize the non-overlapping regions. These two considerations are unified by our novel reformulation of the LDA framework. Our proposed method is called MMDA. We also discover many advantages of MMDA over the conventional LDA.

1.2 Objectives

Motivated by the above discussion, the works dedicated in my thesis focus on improving the existing subspace methods, particularly PCA-based and LDA-based ones, toward the following three objectives:

1. to incorporate margin maximization into conventional linear subspace methods for better extraction of class-relevant information.
2. to find an understanding of what makes a feature relevant in classification problems and to explain the “peaking phenomenon” and overcome it if possible.
3. to explore 2D subspace methods that can help overcome the computation intractability for classification problems with large datasets and high dimensional data.

The performances and characteristics of the proposed methods are evaluated mainly in the domain of face recognition problems.

1.3 Thesis Contributions

This thesis contains three major contributions:

- (i) The additive formulation of LDA which we call Margin-Maximization Discriminant Analysis (MMDA). It adds an extra dimension to allow for the search of optimal margins for better performance over the conventional quotient-form LDA, and hence, intrinsically avoids the singularity problem. In addition, MMDA possess the following good attributes: a) It can achieve direct dimension reduction by the complex domain trick. b) It generates orthogonal directions/features and yields more features than the conventional LDA. c) It allows for the kernel trick.
- (ii) The Relevance-Weighted Discriminant Analysis (RWDA) to avoid the peaking phenomenon. We hypothesize that as long as a feature is relevant, it will improve the recognition rate if correctly weighted. Also, we have proposed the relevance weighting that has been verified in face recognition experiments. Consequently, we show that the reported peaking phenomenon which attributed the deterioration in recognition rates beyond the peak to overfitting was not so.
- (iii) Explorations into 2D Subspace Method. We pioneered the idea of 2D PCA instead of applying PCA to the vectorized image. The key motivation is to solve the high dimension and very large data size problem that plagues dimensional reduction methodologies such as PCA and LDA. Some novel ideas were explored and observations made. It is noted that since then there have been progress by others who have made 2D PCA and 2D LDA successful and efficient method.

1.4 Thesis Organization

The rest of the thesis is organized as follows:

Chapter 2 introduces some background knowledge on face recognition and the use of subspace methods for classification tasks. This is followed by the proposal and evaluation of the Margin-Maximization Discriminant Analysis (MMDA) in chapter 3. Chapter 4 discusses our definition of feature relevance and proposed a weighing scheme for LDA, called Relevance-Weighted Discriminant Analysis (RWDA), which uses an analytical feature relevance measure to overcome the peaking phenomenon and achieve better generalization than LDA. In Chapter 5, we explore several approaches of 2D subspace methods which uses 2D image representation to overcome the computation limit for problems of high data dimension and very large data set.

Finally, Chapter 6 lists the conclusions of the whole thesis and the recommendation for further research work.

Chapter 2

Background

In this chapter, we provide background knowledge on machine-based face recognition and subspace methods. Concepts of feature selection and support vector machines are also introduced in the context of statistical pattern classification, which serve as fundamental knowledge for the rest of the chapters.

2.1 Machine-Based Face Recognition

As part of biometric identification technology, face recognition by computer vision has become an active research area in the past three decades. It finds wide applications in authentication/identification systems, some examples of which include law enforcement, security access control, public surveillance, and the new generation of human-machine interface (such as in teleconferencing and video phone), where the machine needs to be aware of people in its immediate presence.

Typically, machine-based face recognition systems are based on the images or video sequences that contain human faces. The system needs to extract faces out of the background (face detection), generate characteristic features (feature extraction), and then match the features to those of the faces in a pre-stored database (matching or classification). Since the images and videos can be taken without people's awareness, face recognition often boasts its non-intrusive nature compared to other biometric technologies such as retina recognition and fingerprint recognition.

In the context of pattern recognition research, we often attribute face recognition purely to the task of extracting features from face images and matching them to a known person. Thus, the task mainly concerns differentiating objects within the face class, which all have similar structures but experience large appearance changes. Compared to the face detection, which refers to the task of locating faces in the input media, it is much more complex, as the face detection only needs to distinguish face objects from non-face objects.

Depending on the purpose of application, face recognition can also be referred to as face verification or face identification tasks. There are some differences between the two: face verification is to compare an input face with a claimed identity to the images of the same person stored in the database to see whether the claim is true.

Face identification, on the other hand, is to compare an unknown face with all the faces in the database to find out whether there exists a true match.

In the following sub-sections, we first summarize the major challenges in current research of face recognition systems. This is followed by a brief review of the machine-based face recognition techniques.

2.1.1 Challenges Still Remaining in Existing Face Recognition Research

Face recognition appears to be an intuitive task for human vision system. However, it is very challenging for machine vision systems due to the difficulty in extracting invariant features for individuals under various appearance changes (large within-class variations). Extrinsically, such variations may come from different viewpoint (pose and scale), and illumination. Intrinsically, expression, wearing of spectacles, make-up, beard, aging, e.t.c., also makes the problem hard to handle. Currently, most of the research focuses on resolving pose, illumination and expression variations. The appearance changes due to intrinsic factors are mainly handled by continuously updating new “looks” of a face into the database. But even if we can generate invariant features for each individual, it is still questionable whether such features are differentiable among different individuals in a large database.

Some interesting results on difficulties in recognizing different groups of people have been reported in Face Recognition Vendor Test (FRVT) 2002 [106], which provides independent evaluations of commercially available and prototype face recognition technologies. Based on their results, the current systems tend to recognize males more easily than females, and older people more easily than younger people. This suggests that the representative face space may need to be adapted for different groups of people to ensure robust recognition performance in general.

The latest results of FRVT in 2006 [107] reported the recent accomplishments for state-of-the-art face recognition technologies. Their major findings are summarized as follows:

- Since 2002, the error rate of face recognition has dropped by at least an order of magnitude. This might be achieved by the use of high-resolution still face images and the advances in algorithm design.
- For the ability to recognize faces across illumination changes, some of the best systems exhibits comparable or even better performances as compared to human-beings in recognizing unfamiliar faces. This improvement may not be due to the use of high resolution images, but rather may have come from better understanding of illumination correction across images.

These results show that over the last 5 years, tremendous progress has been made in the design of machine-based face recognition systems. The available systems are now more robust and less vulnerable to illumination changes. However, even the best system evaluated still cannot compete with human capability in recognizing

familiar faces, which is far more robust than recognizing unfamiliar faces [51]. Thus, the challenge for current face recognition research is to deal with pose, expression and all sorts of intrinsic changes, and ultimately equip the machine with capability similar to that of humans in recognizing familiar faces. Since the machines can potentially handle much larger face database as compared to human-beings, this would mean a lot to practical large-scale applications, and eventually contribute a lot to pattern classification and machine learning research.

2.1.2 Review of Machine-Based Face Recognition Techniques

Parallel to “re-engineering” of human face recognition systems, researchers from computer science have devised a large number of models and algorithms for machine-based face recognition. Some of them are specific to face objects, but more of them are general to pattern classification and object recognition problems. Two comprehensive surveys for machine-based face recognition techniques have been conducted in 1995 [15] and 2003 [147] by Chellappa and Zhao et. al.. In this section, we briefly review the techniques used for feature extraction and matching/classification of face recognition, and the review of feature extraction is further grouped into facial feature-based methods and template-based methods.

1) Facial feature extraction

One way to automate face recognition is to extract facial features such as eyebrows, eyes, nose, mouth, chins and even face profiles (side views), and match faces based on their geometric measurements, such as shape and relative positions. Most of the early systems adopted this approach [43] [68] [65] [66] [52]. This kind of methods can achieve compact face representation by retaining only the 2D structural information. However, they have two major problems. Firstly, the geometric measures of facial features alone do not appear to be distinguishable enough across different persons, and they often give low recognition rates [13] [21]. This might be partially due to the non-rigidity of faces, for example, the expression changes. Secondly, the automatic extraction of facial features has always been a complex issue, of which simple image processing (such as line/edge detection) or template matching techniques [144] may not be working robustly owing to vast appearance changes and illumination variations. Thus, one has to consider configural structure of facial features and use deformable models to locate the important facial features. The most prominent models of this kind include Cootes et. al.’s Active Shape Model (ASM), Flexible Appearance Model (FAM) and Active Appearance Model (AAM) [18] [77] [19].

More generally, facial feature can also refer to some local abstract characteristics of face regions such as lines, curves, edges, fiducial points or areas. In [100], Nakamura et. al. used isodensity lines to identify faces. Later, Gao and Leung [41] proposed to represent face images using Line Edge Map (LEM), and they achieve good classification results as compared to the popular eigenface method especially

under varying lighting conditions. Such abstract facial features can be more easily generated using image processing techniques. However, some prefiltering techniques are often required to remove irrelevant edges or points, in order to speed up the matching process.

Facial features can also be extended to 3D shape features that are generated from 3D face model [80] [45] [121] [50]. A recent survey of 3D and Multi-Modal 3D+2D face recognition techniques have been conducted by Bowyer et. al. [11]. The 3D face models can be acquired from laser range finders, stereo imaging, or structured light approaches. Some researchers also tried to recover 3D shape of faces by morphing from multiple 2D images. The use of 3D shape features can potentially resolve the problem of pose variation. However, its enhancement in handling non-rigid face deformation, such as expression change, is not so straightforward.

2) Template feature extraction

Another group of face recognition algorithms treats the whole face image as a template and directly extracts features from its raw intensity values. In some literature, they are called template-based or appearance-based face matching, which tackle all sorts of face appearance variations by including enough sample templates in the database.

The most popular methods for template feature extraction is perhaps the group of subspace methods, examples of which include Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA). Turk and Pentland applied PCA to face recognition and extracted a set of “face-like” feature vectors which they call eigenfaces [131]. It demonstrates fast feature extraction and good classification rates, which then led to numerous following works [105] [104] [38] [97] [96]. PCA minimizes information loss in subspace representation, but it could retain much unwanted variations, such as intra-class appearance variations, and may not be optimal for classification. LDA (sometimes called Fisher Discriminant Analysis) was then applied to extract more class-discriminant features [120] [4] [146], and many of its enhancement versions were also proposed in searching of optimal features for classification [86] [87] [85]. PCA and LDA work on second-order statistics and have become performance benchmarks in face recognition. In later researches, ICA was also applied to this field to explore feature vectors that minimize dependencies in higher order statistics [98] [29]. Recently, nonlinear subspace methods that apply kernel transformations to PCA, LDA and ICA e.t.c. have also become popular. This led to numerous studies on Kernel-PCA, Kernel-LDA and Kernel-ICA for face recognition applications [141] [89] [93], and many have reported superior recognition performance. The details of these subspace methods will be further discussed in next section.

Another group of methods for template feature extraction are based on a set of predefined basis functions that are independent of the training data. Examples of them include the use of Discrete Cosine Transformation (DCT) [101] [71] [127] and Wavelet Transformation (WT) [7] [139] [84] in face recognition. DCT and WT

achieve good image compression and aim to minimize human perception error after reconstruction. They basically extract spatial-frequency information of images as the template features and use them for face matching.

The template features discussed above capture global characteristic of face images and constitute a holistic approach. As motivated by studies of both human perception and machine-based face recognition, facial features are also crucial for the recognition of faces. A trend of the current face recognition system is to use a hybrid approach, for example, by applying template feature extraction around fiducial points of faces. Typical examples are Wiskott et. al.'s Elastic Bunch Graph Matching [139] and its successors. This approach utilizes both holistic and local feature information of faces, and has become more and more popular in new design of recognition systems.

3) Matching and classification

After feature extraction, the matching/classification of faces can be performed with any classification methods. The popular ones used for face recognition include Nearest Neighbor, Neural Networks, Hidden Markov Models, and more recently, Support Vector Machines. Moreover, methods such as Elastic Graph Matching and Object Trajectory Matching have also been customized for face objects.

Nearest Neighbor or k-Nearest Neighbor is perhaps the simplest classifier, which assigns an input query a label based on the identity of the nearest samples. The distance between the samples is usually measured by Euclidean distance, or L_p norms (e.g. $p = 1, 2, \text{ or } 3$). It requires no training and relies on a sufficient number of samples to achieve good classification. Although being naive, Nearest Neighbor has been popular in classification systems and often served as a benchmark for algorithm comparison. For face recognition, it has been implemented with many subspace methods such as PCA and LDA [131] [4].

Neural Networks take analogy of biological neural systems, and have been widely used for face classification [37] [58] [79]. Many network configurations have been designed which possess different learning characteristics. The special trait of neural network is that it can combine feature extraction and classification within one network structure. Moreover, the classifier can also be nonlinear by choosing certain configuration.

As a probabilistic method, Hidden Markov Model (HMM) that is originally applied to 1-D data also finds successful applications as face recognition classifiers [101] [71]. Nefian and Hayes [101] coded faces with their 2-D HMM (embedded HMM). In [7], it was claimed that the combination of wavelet encoding together with HMM can outperform other methods.

More recently, Support Vector Machine (SVM) has become popular and is widely applied to numerous classification tasks including face recognition [46] [54] [60]. It finds separating hyper-planes with optimal margin and demonstrates better generalization for classification. However, the method is essentially a binary classifier. Special formulation or framework has to be adopted to make it suitable for face

recognition problems, which may sacrifice its optimality. Moreover, during classifier training, the computation increases tremendously with number of samples, and cross-validation is often necessary to select proper parameters. In Section 2.2.4, more details of SVM are provided.

Other types of classifiers that are successful for face recognition include Elastic Graph Matching, which was applied by Wiskott et. al. to wavelet coded sub-templates of faces [139]. They report superior recognition performance across poses. In [8], Object Trajectory Matching has also been proposed, which builds trajectories of each face sequence (profile to profile) in a low dimensional eigenspace and uses them for matching.

To this date, face recognition from video sequences have become more and more popular [75] [49] [123]. In addition to compare spatial information as those from static images, classifiers now need to explore temporal information from consecutive frames as well. In [88], HMM has been used to learn the statistics and temporal dynamics of the face sequences, which was then used to provide likelihood scores during classification. As it is believed that the temporal information is the key to provide dynamic cues for pose and expression changes, classifiers that can effectively capture such correlation would become more and more of research interests.

2.2 Subspace Methods for Statistical Pattern Classification

When modeling the datasets (e.g. to build a classifier), the number of samples required for accurate parameter estimation grows exponentially with the dimensionality of the feature space [6] [30]. This observation is often referred to as Curse of Dimensionality, and it makes it a necessary process to perform dimension reduction or feature extraction before the actual classification occurs.

Subspace methods are widely used for dimension reduction and feature extraction in pattern classification, where the original data dimension is often very high and consists of a lot of information redundancies. They find a subset of important feature directions that optimizes certain statistical property of the samples. The subspace representation of the input data is then obtained by projecting the data onto these feature vectors through linear transformation.

The history of the subspace methods in data analysis dates back to the 1930s [57]. The methods were first applied in data compression and optimal reproduction, such as in [74], and it was only until 1967 that their applications in pattern classification emerged [137]. One way to apply subspace methods for pattern recognition is as described by Oja [102], whereby each class of patterns has its own set of basis vectors derived. Since the 1990s, holistic subspace that uses common basis for all the classes has become popular, which suggests a simple, yet effective feature extraction scheme for pattern classification.

Most of the subspace methods are now based on the holistic subspace approach. Popular examples of them include Principal Component Analysis (PCA), Linear

Discriminant Analysis (LDA) and Independent Component Analysis (ICA). PCA chooses feature directions that maximize projected data variation. This produces a set of uncorrelated features that can represent data with minimum loss of information. However, they are not necessarily good for discrimination in classification. LDA seeks to find features that can better discriminate different classes. This is done by extract feature directions that maximize the between-class scatter and minimize the within-class scatter. Theoretically, it is generally believed that LDA outperforms PCA for classification. However, some researches show that it may not always be the case [92]. This might be due to poor data sampling or bad parameter tuning in the LDA process. Both PCA and LDA work on second-order statistics, ICA, on the other hand, works on second-order and higher-order statistics to minimize input dependencies. The normal procedure of ICA is to firstly decorrelate the input data by PCA, and then reduce the remaining higher-order dependencies by ICA. ICA is originally popular in Blind Source Separation (e.g. separating two audio signals using the fact that they are independent), but its application to classification tasks such as face recognition has also been explored [29].

The above mentioned subspace methods are all linear. Recently, many of their nonlinear forms have been proposed using kernel transformation, which makes them more flexible in capturing nonlinear structure of the datasets. The principle is to use an appropriate nonlinear kernel mapping to transform data to a sufficiently high dimensional space, and then obtain subspace features as projection directions in this nonlinearly mapped space. Due to the high dimensionality of the nonlinear space, this approach would not be feasible if we implement them directly. Fortunately, “kernel trick” can be applied, which evaluates the similarity between the nonlinearly mapped samples in terms of a kernel function [114]. This avoids explicit computation of the nonlinear mapping of the samples, and leads to many implementations of the nonlinear subspace methods such as Kernel-PCA, Kernel-LDA and Kernel-ICA [141] [89] [3]. They make subspace methods more useful for practical applications at additional costs of kernel selection and fine-tuning.

The remainder of this section gives background information on some important subspace methods and statistical pattern classification issues that are relevant to this thesis. Firstly, PCA and LDA algorithms are described in details. This is followed by an introduction to feature selection methods. Finally, classifier design principles and support vector machines are introduced.

2.2.1 Principal Component Analysis (PCA)

PCA is an orthogonal subspace transformation, which extracts features as the projection directions that capture maximal sample variances. These features are called Principal Components, or, in the context of face recognition, Eigenfaces [131]. For practical datasets that have high correlation in the input space, usually only the first few principal components contain significant data variation, and are sufficient for data representation. This makes PCA very effective for dimension reduction, which retains as much as possible of the sample variation in the preserved PCA features.

The formulation of PCA is summarized as follows: Consider N samples, $\mathbf{x}_1$ to $\mathbf{x}_N$, represented in vector form, the scattering of the samples can be measured by the covariance matrix

$$\mathbb{C} = E_i\{(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T\}, \quad (2.1)$$

where $\mathbf{m}$ is the sample mean, and $E_i\{\cdot\}$ denotes the expectation of the variables for all i . The first K (assume K is much smaller than the original data dimension) principal components $\{\mathbf{w}_k\}$ can be solved as the eigenvectors of $\mathbb{C}$ corresponding to the first K largest eigenvalues. This results in orthogonal PCA feature sets $W_K = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K]$, of which the corresponding eigenvalue λ_k gives the projected sample variance on that feature. Finally, a sample can be represented by its projection on the principle components as

$$\mathbf{y}_i = W_K^T(\mathbf{x}_i - \mathbf{m}), \quad (2.2)$$

where the sample mean is subtracted to centralize the transformed data representation. Classification can then be carried out with the transformed data representation in the PCA subspace.

In the context of machine learning, PCA is often referred to as a unsupervised method, which retains data structure in low dimension without considering the class labels. It is mainly used as a dimension reduction tool for classification, and is not designed to extract class-discriminant information. However, many researches show that it is still quite effective for classification tasks [131] [92].

2.2.2 Linear Discriminant Analysis (LDA)

LDA is an extension of PCA that takes account of the class labels (a supervised approach). It finds feature directions that minimize within class-scatter, and at the same time, maximize between-class scatter. In this way, the features obtained preserve more class-discriminant information. The formulation of LDA allows it to be directly applied to multiclass problems. It is said that LDA can produce optimal feature sets for classification if the class distributions are independent and identically distributed Gaussians [30], although for practical datasets this is rarely the case.

The formulation of LDA goes as follows: consider N samples $\{\mathbf{x}_k\}$ belonging to C different classes, and assume that there are N_i samples for class i that is defined by subset $\mathcal{C}_i$. The between-class and within-class scatter matrices of the samples can be computed as:

$$S_B = E_i\{(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T\}, \quad \text{and} \quad (2.3)$$

$$S_W = E_i\{E_k\{(\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^T\}\} \quad (\forall \mathbf{x}_k \in \mathcal{C}_i), \quad (2.4)$$

where $\mathbf{m}$ is the global mean, $\mathbf{m}_i$ is the class mean, and $E_i\{\cdot\}$ denotes the expectation of the variables for all i . The Fisher's discriminant function is defined as

$$\mathcal{J}(\mathbf{w}_k) = \frac{\mathbf{w}_k^T S_B \mathbf{w}_k}{\mathbf{w}_k^T S_W \mathbf{w}_k}, \quad (2.5)$$

in which LDA features are found as a set of unitary vectors $\mathbf{w}_k$ that maximizes $\mathcal{J}(\mathbf{w}_k)$.

The vector $\mathbf{w}_k$ can be obtained by solving a generalized eigen-decomposition problem $S_B \mathbf{w}_k = \lambda S_W \mathbf{w}_k$, or equivalently, $S_W^{-1} S_B \mathbf{w}_k = \lambda_k \mathbf{w}_k$, where λ_k is the corresponding generalized eigenvalue of $\mathbf{w}_k$. The value of λ_k indicates the ratio between the projected inter-class variance over intra-class variance along feature $\mathbf{w}_k$ [30]. Similar to PCA, data can be represented by the first few LDA features with the largest λ_k values, and the projection of the input data to the LDA subspace can be similarly computed through Eq. 2.2.

Unlike PCA, LDA produces feature sets that are non-orthogonal, meaning that they are correlated and in the context of information theory, can be said to contain mutual information. Thus, LDA is not as effective as PCA for data compression, but the former usually results in better classification results. In practical implementation, LDA often encounters “small sample size (SSS) problem” and “peaking phenomenon”, which greatly affect its performance stability. In Chapter 3 and 4, we will discuss these issues in more details.

2.2.3 Feature Selection

The subspace methods discussed above often belong to the feature transformation or feature extraction stage, which transform the original inputs into a small, or compact set of features, and achieve significant reduction of data dimensionality. However, not all the features obtained this way are relevant to the classification task. Some of them may even consist of noise. To save the resources in data representation and processing, more importantly, to further reduce the feature dimension, and hence, avoid curse of dimensionality, feature selection process shall be engaged before classification occurs.

Differentiating itself from feature transformation, feature selection can be defined as a process to select a subset of the original features that optimizes one or more criteria, rather than producing an entirely new set of dimensions for the data [23]. Feature selection methods can be grouped into combinatorial and weighting approaches, of which, the combinatorial approaches can be further divided into filters and wrappers. Below we briefly describe the concepts of various feature selection approaches. For more details, one may refer to survey papers in [9] and [48].

The filter model chooses features by the intrinsic property of individual feature. The features are either used or discarded. The name “filter” comes from the fact that it filters out irrelevant attributes before induction or classification occurs [63]. For example, in [94], information gain was used to rank and select features. In some literature, subspace methods were also referred to as filters as they select a subset of feature directions purely based on certain statistical properties of the input data. Thus, filters are independent of the induction or classification algorithm, and can be combined with any such method. However, in filters, the features are considered in isolation, which may not be optimal as we know sometimes only certain feature combinations give good results [9].

The wrapper model, on the other hand, evaluates all possible subsets of the features according to the results of the selected classifier, and chooses the set that gives best classification result. They consider the classifier as a subroutine, rather than as a postprocessor, and hence, are named as wrappers [63]. Since the number of possible feature combinations are exponential to the number of features, practically wrappers are very time-consuming and cannot work with naive algorithms such as exhaustive search. Popular wrapper methods go with stochastic search (e.g. genetic algorithm and simulated annealing) and sequential search based on greedy heuristics (e.g. forward selection or backward elimination) [9]. Moreover, in contrast to filters, features are evaluated in combination rather than in isolation. Research showed that wrappers perform better than filters especially when generalizing to unseen data [63]. However, they are also less computationally effective than filters as they need to call for classifier evaluation for each feature set considered. In general, wrappers are not practical for very large number of features (e.g. > 50). For such case, it is often recommended to use filters as a preprocessing stage to confine the features to a much reduced and promising subset, and then use wrappers to optimize the feature selection with respect to a particular classifier.

Both filter and wrapper can be considered as a sort of “hard” feature selection as the features are either selected or not. Another type of “soft” feature selection has also been proposed in the form of feature weighting, which put relevance weights to the available features to control their usage in classification. Feature weighting is most useful for classifiers such as k-NN that measure similarity using distance metric [103]. Most feature weighting methods are based on heuristics, i.e. the weights are updated according to classifier evaluation with the training samples. Other ways of computing weights may come from conditional probability distributions [118] [126] and information-theoretic metric [24]. More details of the feature weighting issues will be discussed in Chapter 4, where we propose our own relevance and feature weighting schemes.

2.2.4 Classifier Design and Support Vector Machines

Most of the traditional classifiers are designed purely based on empirical risk minimization, which minimize training errors. If the classifier is complex enough, the training error can always be reduced to zero. However, this often make the classifier overfit to the training data and fail to capture the true underlying structure of the data distribution. As a result, the generalization ability of the classifier to unseen data is severely impacted.

Vapnik et. al. proposed structural risk minimization (SRM) as a guideline for model selection when learning from finite training data sets [133] [20] [132]. They show that the upper bound of the actual risk (generalization error) can be defined as the sum of the empirical risk and a complexity penalty term of the model. Therefore, an upper bound on generalization error can be minimized by keeping the classifier complexity low, and at the same time, minimizing the empirical error. Compared to empirical risk minimization, SRM directly minimizes the actual error bound and

leads to classifiers with better generalization capabilities.

The complexity of a classifier can be measured in terms of Vapnik-Chervonenkis (VC) dimension, which is the largest number of points that a classifier can shatter¹ [114]. It has been proved by Vapnik that maximizing the margin of separation between the classes is equivalent to minimizing the VC dimension [134]. This leads to the development of a group of classifiers: the Support Vector Machines (SVM) and its variants.

SVM finds the optimal hyperplane that gives the largest margin of separation between the training samples of two classes [114]. Since such hyperplane represents classification functions of lowest possible complexity, it actually performs structural risk minimization and can achieve good generalization regardless of the dimensionality of the input data. SVM has recently become very popular because its nonlinear form can be feasibly obtained using the kernel trick. To handle real datasets where the classes are most likely non-separable, soft margin SVMs, such as C -SVM and ν -SVM [16], have also been proposed, which allows to ignore some outliers of the class samples by adding certain constant slack variables. These constants control the trade-off between the margin and the misclassification errors.

The training of SVM is by solving a quadratic programming (QP) problem with linear constraints. The resultant optimal hyperplane is described by linear combination of support vectors (i.e. the samples sitting on the class margin). Thus, the complexity of SVM depends on the number of support vectors, not on the dimensionality of the feature space. The parameters of SVM, such as kernel type and kernel parameters, often have large impact on its classification performance, and are usually determined by cross validation with the training data. The QP problem of SVM scales exponentially with number of samples, to overcome the computation and memory constraints, practical SVM implementations usually only approximate the optimal solution by iterative strategies [62] [108] [129] [114].

Another limitation of SVM is that it is essentially a binary classifier, which is designed to find optimal solution for two class problems. Many researches have been done to extend SVM to multiclass problems, most of which decompose the problem into a series of binary ones, for example, in pairwise structure or in one-versus-rest structure [28] [2]. This kind of approaches breaks one multiclass problem into a large number of binary problems, which is not effective when the number of classes is large. Moreover, they require special voting scheme to merge the results from binary classifiers and do not necessarily imply the best rule for the original k -category classification problem [81]. In [132] [138] [12] and [81], true extensions of SVM to the multiclass formulation have also been proposed by constructing the decision function considering all classes at once. Like binary SVMs, the performance of these multiclass SVMs heavily depends on parameter tuning, which is often through time-consuming cross-validation processes. Moreover, they generally only approximate the optimal classifier compared to their binary counterparts.

The discussions above only serve as a brief introduction to the concepts of SRM

¹“Shatter” means to generate all possible labeling of the points.

and SVM. For details, please refer to [94] [16] [114].

Chapter 3

Margin-Maximization Discriminant Analysis

3.1 Introduction

LDA and its linear/non-linear variants have been shown to be very successful for multi-class problems with high dimensional data, such as face recognition applications [4] [143] [141] [90]. Generally, they find features as projection directions that maximize the Fisher discriminant $\mathcal{J}(\mathbf{w}) = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}}$, where S_B and S_W are between-class and within-class scatter matrices of the training samples. This conventional quotient form of the discriminant function actually poses some limitation to the performance of LDA. Firstly, it can only measure a type of normalized distance between the class center, that does not necessarily reflect the maximum separating margin between the classes. Secondly, to find the LDA features, we often need to solve the generalized eigen-decomposition problem $S_B \mathbf{w} = \lambda S_W \mathbf{w}$, or equivalently, $S_W^{-1} S_B \mathbf{w} = \lambda \mathbf{w}$. The involvement of the term S_W^{-1} not only causes computational inefficiency, but also makes the LDA solution inherently unstable. This is especially prohibitive when there exists the Small Sample Size (SSS) Problem, where the number of samples available is much less than the dimensionality of input data. In such a situation, S_W becomes singular or near-singular. It would be impossible for LDA to directly extract features from the input data space, and subsequently to achieve direct dimension reduction [40] [61]. In practical applications such as face recognition, this is often the case as the input data may contain all the pixel values of a image which is of very high dimension.

We recognize that these limitations of LDA come about mainly because of the quotient structure of the Fisher Discriminant, and we ask whether the S_W^{-1} term can be avoided in the formulation of a class-discriminant analysis. For classification tasks, we actually prefer features with least overlap among the classes. This can be measured as the average margin between the class boundaries. Zero overlap will coincide with the case of separable classes, a desirable situation though seldom met in real applications. Almost all classifier designs have given attention to the maximization of the margin between classes. When the classes are nonlinearly-

separable, kernel methods are applied to convert the input feature space into (often higher dimensional) kernel space that hopefully leads to linear separability, where the SVM classifier exhibits superior generalization performance. But when the classes are non-separable, kernel transformations do not help. In this case, the soft-margin SVM is required to allow tolerance of errors. The soft-margin SVM minimizes the square of the misclassification training errors. We propose an alternative approach. Instead of focusing on the margin, we could think of maximizing the non-overlap between classes. In this way, our method will take care both the separable and non-separable cases simultaneously.

In this chapter, we propose an alternative form of discriminant analysis, which is called Margin-Maximization Discriminant Analysis (MMDA), based on an additive form discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$. This additive form turns out to possess a number of important advantages over the quotient form of the LDA. We will show that such a discriminant function can approximately maximize the average margin between the classes with the help of a class-spread regulator β . Compared to the quotient form, our MMDA naturally avoids the singularity problem of S_W , and can, as will be proved, allow for direct extraction of discriminant features from the input data space. This avoids the possible loss of discriminant information encountered by methods of others in handling the singularity problem of LDA [117] [143] [61], and is also computationally more effective than them.

Our MMDA formulation is derived independently; however, we have found in the literature a similar formulation by Li et. al. [82], which is called Margin Maximization Criterion (MMC). Both the methods introduced the additive structure but the principles underlying each of the two methods are different. Li et. al.'s method is based on maximizing the margin; the authors give a derivation assuming that the class distributions are Gaussian. This is constraining and for arbitrary distributions, we show that their derivations cannot lead to true margin maximization. On the other hand, our MMDA makes no assumption regarding class distribution and furthermore, takes into account both separable and non-separable cases, as mentioned before. In addition, there are significant differences in the underlying derivations and the framework of the methods. In our approach, we introduce a class-spread regulator β in the additive form discriminant function. This is necessary, otherwise as in their case, converging to the maximum margin would not be possible in general. Also, we give much further insights into the properties of the additive structure.

Looking at the formulation, MMDA becomes PCA when $\beta = -1$, and the above mentioned MMC when $\beta = 1$. Therefore, MMDA can be considered as the generalization of PCA and MMC. The use of β in effect provides an extra degree of freedom over LDA, which enables us to search for features with better margin between the classes. As propounded by Vapnik et. al. from the structural risk minimization point of view [134], classifiers with large margin tend to have strong underlying generalization properties. For non-separable cases, β can be considered as the counterpart of the C and ν parameters used in soft margin SVMs such as C -SVM and ν -SVM [16], which allows to ignore some outliers of the class samples. The value of β can be optimized via cross validation with the training samples. As will be shown later

in the experiments, the performance of MMDA is actually quite stable over large β values (e.g. $\beta \geq 9$).

Two issues are discussed. The first is that in many classification problems, such as in text and face recognition applications, the feature dimension d is very large. So it is essential to reduce this dimension significantly before classifier design and for fast on-line classifications. PCA is the usual tool for this. In PCA, the eigenvectors of the inherent covariance matrix $\mathbb{C}$ are extracted. But $\mathbb{C}$ has the form $\mathbb{C} = AA^T$ which is $d \times d$. The usual trick is to solve for the eigenvectors indirectly by $A^T A$, a well-known computational trick in PCA. This works if the number N of training samples is much less than d . And so, a $N \times N$ matrix is tractable. But when N is large and even greater than d , this trick is no longer of value. The second issue is that for the conventional quotient LDA feature extraction, the computation trick cannot be used. Consequently, more researchers resort to first reducing the feature space via PCA. But this will lead to possible loss of discriminative information. In this chapter, we will show how MMDA solves both issues simultaneously.

We also propose a computation trick for MMDA which makes the computation less prohibitive when handling very high dimensional input data. It turns out that MMDA also achieves dimension reduction directly in the discriminant domain. This is significant since conventional approaches implement PCA for dimension reduction before feature extraction by LDA. It will be shown that this can lead to the possibility of loss of discriminative information, which deteriorates the classification accuracy. To cater for non-linearly separable situations, non-linear MMDA will also be derived using the well-known “kernel trick”, which enables one to compute the nonlinear transformation of data via some kernel functions.

The organization of this chapter is as follows: First, the formulation of MMDA and its additive form discriminant function is derived in Section 3.2. This is followed by the discussion of its advantages in direct dimension reduction in Section 3.3, and a computation trick for efficient feature extraction of MMDA is also presented in Section 3.3.1. In Section 3.4, the margin improvement achieved by the new discriminant function is discussed. The nonlinear MMDA formulation is derived in Section 3.5. The experiments are presented in Section 3.6, which compare the performance of MMDA and LDA over a toy example and for some face recognition problems. The experiments are focused on linear MMDA to verify some fundamental properties. Finally, conclusions are presented in the last section.

3.2 The MMDA Formulation

In this Section, we derive the formulation of MMDA. The original motivation is to find a way to maximize the non-overlapping regions between classes and also to maximize the average geometric margin between classes in feature directions. We will show that such an optimization criterion leads to an additive form of discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$, where S_B and S_W takes the same definition as those used in the traditional LDA. In MMDA, we introduce a constant β which

Figure 3.1: The Projected Margin between Two Classes on a Feature Direction $\mathbf{w}$

regulates the class spread based on the actual sample distribution. Its meaning and principle of value selection has been clearly shown along with the derivation of the new discriminant function. Here, the method will be formulated in linear space. A non-linear version of MMDA will be given later in Section 3.5 using the kernel trick.

Consider N d -dimensional sample data $\{\mathbf{x}_k\}$ belonging to C different classes, and assume that there are N_i samples for class i that is defined by subset $\mathcal{C}_i$. The between-class and within-class scatter matrices can be given in the expectation form:

$$S_B = E_i\{(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T\}, \quad \text{and} \quad (3.1)$$

$$S_W = E_i\{E_k\{(\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^T\}\} \quad (\forall \mathbf{x}_k \in \mathcal{C}_i), \quad (3.2)$$

where $\mathbf{m}$ is the global mean, $\mathbf{m}_i$ is the class mean, and $E_i\{\cdot\}$ denotes the expectation of the variables for all i . Now, we will look into the projected class distributions on a feature direction $\mathbf{w}$, and see how to link the projected class margins to Eq. (3.1) and (3.2).

Let μ_i and σ_i be the projected mean and standard deviation (*std*) of class i samples on $\mathbf{w}$, and assume the distribution of class samples are i.i.d.. The margin between two classes as projected into the direction of $\mathbf{w}$ can be measured as

$$\mathcal{M}_{i,j} = |\mu_i - \mu_j| - b(\sigma_i + \sigma_j), \quad (3.3)$$

where b is a constant that quantifies one-side within-class spread. We call $\mathcal{M}_{i,j}$ the projected margin, and Fig. 3.1 provides a geometric illustration of it. The value of b depends on the actual distribution type of the class samples. For example, if normal distribution is assumed, we can choose $b = 3$, as statistically very few data will fall outside 3 *std*.

Eq. (3.3) reflects the true projected margin on $\mathbf{w}$, but when solving for optimal $\mathbf{w}$ with the largest margin, it will lead to nonlinear formulation which is analytically

Figure 3.2: Function Values of $|a| - |b|$.

intractable. To facilitate the feature derivation, we can approximate Eq. (3.3) using its squared terms as follows:

$$\hat{\mathcal{M}}_{i,j} = |\mu_i - \mu_j|^2 - \beta(\sigma_i^2 + \sigma_j^2), (\beta = b^2). \quad (3.4)$$

For Eq. (3.4), it is easier to provide a linear solution of the optimal features. The difference between $\hat{\mathcal{M}}_{i,j}$ and $\mathcal{M}_{i,j}$ will be illustrated later in Section 3.4. For now, we just need to note that $\hat{\mathcal{M}}_{i,j}$ is an approximation of $\mathcal{M}_{i,j}$ which grossly follows the latter's trend. Fig. 3.2 and 3.3 gives a simple illustration of this point using abstract mathematic symbols $|a|$ and $|b|$ to replace the non-negative terms of Eq. (3.3) and (3.4). We can see that the difference of the squared terms still captures the trend of $\mathcal{M}_{i,j}$, although it is not exactly measuring the geometric margin. Thus, we can use Eq. (3.4) to approximately formulate a margin-based optimization criteria.

Now, the task is to associate Eq. (3.4) to the scatter matrices and feature vectors to form a discriminant function. Taking account of multi-class problems, we aim to maximize the average margin between any two classes. Although this is not an optimal approach, it allows direct formulation of multi-class problems without decomposing into a series of binary classifiers. In LDA-based method, we often adopt this kind of averaging approach to get a simple formulation for multi-class problems. With this thought, the new discriminant function can be formulated as follows:

$$\begin{aligned} J &= E_{i,j}\{\hat{\mathcal{M}}_{i,j}\} \\ &= E_{i,j}\{|\mu_i - \mu_j|^2\} - \beta E_{i,j}\{\sigma_i^2 + \sigma_j^2\}. \end{aligned} \quad (3.5)$$

Figure 3.3: Function Values of $|a|^2 - |b|^2$.

The first part of Eq. (3.5) can be rewritten as

$$\begin{aligned} E_{i,j}\{|\mu_i - \mu_j|^2\} &= \mathbf{w}^T E_{i,j}\{(\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^T\} \mathbf{w} \\ &= \mathbf{w}^T E_{i,j}\{[(\mathbf{m}_i - \mathbf{m}) - (\mathbf{m}_j - \mathbf{m})] \cdot [(\mathbf{m}_i - \mathbf{m}) - (\mathbf{m}_j - \mathbf{m})]^T\} \mathbf{w} \end{aligned}$$

Expanding the terms, we have

$$\begin{aligned} E_{i,j}\{|\mu_i - \mu_j|^2\} &= 2\mathbf{w}^T E_i\{(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T\} \mathbf{w} \\ &= 2\mathbf{w}^T S_B \mathbf{w}, \end{aligned} \quad (3.6)$$

using the fact that $E_{i,j}\{(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T\} = E_j\{(\mathbf{m}_j - \mathbf{m})\} E_i\{(\mathbf{m}_i - \mathbf{m})^T\} = 0$. Similarly, the second part of Eq. (3.5) can be reduced to

$$\begin{aligned} \beta E_{i,j}\{\sigma_i^2 + \sigma_j^2\} &= 2\beta E_i\{\sigma_i^2\} \\ &= 2\beta E_i\{E_k\{\mathbf{w}^T (\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^T \mathbf{w}\}\} \\ &= 2\beta \mathbf{w}^T S_W \mathbf{w}. \quad (\mathbf{x}_k \in \mathcal{C}_i) \end{aligned} \quad (3.7)$$

Thus, Eq. (3.5) is equivalent to

$$J(\mathbf{w}) = \mathbf{w}^T (S_B - \beta S_W) \mathbf{w}, \quad (3.8)$$

with the constant 2 removed. This results in the additive form of LDA.

Eq. (3.8) is a margin-based discriminant function which approximately maximizes the average margin between classes as projected on feature direction $\mathbf{w}$. The best set of $\{\mathbf{w}\}$, i.e. the unit vectors which satisfy

$$\mathbf{w} = \arg \max_{\mathbf{w}} J(\mathbf{w}), \quad (3.9)$$

can be obtained by solving an eigen-decomposition problem of $S_B - \beta S_W$. This solution is easily derived hence. Note that J is homogeneous in $\mathbf{w}$, it should be constrained to be a non-zero vector. Let $\mathbf{w}$ be a unit vector. By the Lagrange Multiplier method for equality constraints, we minimize L instead of J where

$$L(\mathbf{w}) = J(\mathbf{w}) - \lambda(\mathbf{w}^T \mathbf{w} - 1) = \mathbf{w}^T (S_B - \beta S_W) \mathbf{w} - \lambda(\mathbf{w}^T \mathbf{w} - 1) \quad (3.10)$$

$$\frac{dL(\mathbf{w})}{d\mathbf{w}} = 2(S_B - \beta S_W) \mathbf{w} - 2\lambda \mathbf{w} = 0 \quad (3.11)$$

This leads to the standard eigenvector problem

$$(S_B - \beta S_W) \mathbf{w} = \lambda \mathbf{w} \quad (3.12)$$

where to maximize J , select $\mathbf{w}^*$ which is associated with the maximum eigenvalue λ of $S_B - \beta S_W$. Here, the term S_W^{-1} is not encountered and so the small sample size problem is avoided.

Note that $S_B - \beta S_W$ is symmetric but may not be positive semidefinite. Thus, there could be negative eigenvalue, λ_i , for certain $\mathbf{w}_i$, which approximately indicates the extent of overlap between the classes. It shall also be noted that all the MMDA features $\{\mathbf{w}\}$ are orthogonal to each other and the total number of features available depends on the rank of $S_B - \beta S_W$, which is bounded by $\min(d, N-1)$. Usually this is far more than the total number of conventional LDA features available. Conventional LDA yields at most $C-1$ eigen-directions, where C is the number of classes. Since in general $d \gg C$ and $N \gg C$, this is another advantage of MMDA.

As mentioned, $\beta = b^2$ regulates the class spread based on the actual sample distribution. Although ideally we may need different β for different classes and feature directions, in practice one common β is enough to quantify the spread of all classes in any feature direction. A loose bound for such spread regulation is governed by Chebyshev's Inequality for any probability distribution. It states that for a random variable X with expected value μ and finite variance σ^2 ,

$$\Pr(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2}, \quad (3.13)$$

where k is any real number $k > 0$. In other words, no more than $\frac{1}{k^2}$ of the values are more than k standard deviations away from the mean. In practice, the values are usually much closer to the mean than this loose bound. If we assume gaussian distribution, $b = 3$ (or $\beta = 9$) would allow MMDA to take account of 99.7% of the class samples, which is sufficient to generate features with large margin. However, the accurate estimation of β can only be obtained by cross-validation with the testing samples, which would be quite feasible as it is only a one dimensional search.

From structural risk minimization point of view, β provides an extra degree of freedom, which allows to find a better margin of separation than by the conventional LDA. For the case of non-separable classes, β can help to find the maximally non-overlap. Large β used in MMDA helps to control the capacity of the feature sets, which regulates the search of the features towards the ones producing large margin.

This in effect help to boost up the generalization ability of the resultant classifier. Moreover, β also allows to ignore some outliers of the class samples by its nature. For non-separable cases, this is a very useful trait. It is similar to soft margin SVMs such as C -SVM and ν -SVM [16], which uses constant C or ν to control the tradeoff between margin and misclassification errors. In other words, it helps to prevent the outliers from affecting the optimal hyperplane.

3.3 Direct Dimension Reduction

In this section, we discuss the details of one important property of MMDA, which is Direct Dimension Reduction. As explained early, for high-dimensional input data, LDA and its variants often cannot achieve direct dimension reduction due to the singular or near singular scatter matrix, S_W , which is used in the denominator of the Fisher discriminant function. To deal with this singularity problem of LDA, many techniques and LDA variants have been proposed, which can be roughly grouped into three approaches.

The first approach, which is also the most pioneering one, uses pseudo-inverse $S_W^\dagger$ to replace S_W^{-1} [125] [40] [117] [116] [111]. Ye and Janardan et. al. [61] pointed out that the pseudo-inverse approach is equivalent to approximating the solution using a least-squares solution method, and provided a theoretical justification of it. However, this kind of approach is computationally intensive when dealing with high dimensional data, and recently many approximations have been proposed [61] [14].

The second approach adds a small perturbation to the scatter matrix [39] [55] [148] [25] [90] to make it nonsingular, which functions like a regularization term. The algorithm is commonly known as the regularized LDA, or RLDA for short. A limitation of RLDA is that the amount of perturbation to be used is difficult to determine [25]. If the value is large, we lose the information on the scatter matrix, whereas very small values may not be sufficiently effective to resolve the singularity problem [117]. Usually, the proper values of the regularization terms have to be found via cross-validation from the training samples. However, this might be computationally prohibitive. Some of the methods even introduce two regularization terms which require two-dimensional search of the parameters.

The third approach is often called two-stage LDA, which applies a pre-dimension reduction step to the input data and then perform LDA on the reduced space. The most popular one of this type uses PCA for the dimension reduction step [4]. Similar dimension reduction methods, such as LSI and Partial Least Squares (PLS), have also been tried [76] [120].

In [143], the null space of S_B was discarded so that S_W can be maximized in the remaining space. Recently, Discriminant Common Vectors (DCV) [14] and Linear Discriminant Analysis via QR decomposition (LDA/QR) [142] have also been proposed, where the training samples were projected to the null space of S_W or the range space of S_B prior to the LDA-based process. Experiments in [76] showed that some two-stage LDA methods achieved comparable performance as the pseudo-

Figure 3.4: PCA Dimension Reduction May Lose Discriminant Information.

inverse approach and RLDA. The problem with the two-stage approach, however, is the risk of losing discriminant information during the dimension reduction step. This can be illustrated by a simplified graph in Fig. 3.4, using the PCA-based dimension reduction [4] for example. As shown in Fig. 3.4, if we need to discard one direction, PCA will discard the one with better class discrimination (the second principle direction), since in this direction the total data has small variance. But when we project the data to the first principle direction to perform LDA, we will find that an originally separable problem has become non-separable due to this dimension reduction process.

In our proposed MMDA, we avoided the singularity problem of LDA by employing an additive form of discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$ (c.f. Eq. (3.8)). Therefore, all the features can be directly obtained from the input space without being affected by the Small Sample Size (SSS) Problem. Our method is a true direct LDA dimension reduction method not requiring the standard PCA pre-processing step. We have shown that PCA dimension reduction can sometimes lose discriminative information. In this respect, our MMDA method preserves the discriminative information. Moreover, our proposed MMDA is inherently more stable than LDA which is based on a quotient form discriminant function. LDA performances are shown to be very sensitive over certain parameters, such as the amount of perturbation added [25] and the reduced space dimension of PCA (c.f. Section 3.6.2). Whereas as will be shown later in Section 3.6, the performance of MMDA is quite insensitive to β if it is large enough.

However, MMDA may still be computationally prohibitive if the input space dimension, d , is very high. This is because to solve for the MMDA features, we need to do eigen-decomposition on a matrix of size $d \times d$. In the subsection below, a computation trick is proposed to resolve this problem, which will make MMDA

more effective.

3.3.1 The Computation Trick for MMDA

In this section, we propose a computation trick of MMDA for applications with high dimensional input data, where the data dimension, d , is much larger than the number of the samples, N , ($d \gg N$). It solves MMDA by eigen-decomposing a matrix of the size $(N + C) \times (N + C)$ instead of $d \times d$ (C is the total number of classes), which in effect, scales down the computation complexity from $O(d^2)$ to $O((N + C)^2)$.

The idea is very similar to how we find high-dimensional PCA features [143]. Since both S_B and S_W are symmetric matrices and β is a constant, $S = S_B - \beta S_W$ is a symmetric matrix. Under the above assumption, $\text{rank}(S) \ll d$, we can find a matrix Z such that $S = ZZ^T$, and compute the eigenvectors of S from a much smaller matrix $Z^T Z$. However, since S may not be positive semidefinite, the eigenvalues can be complex. This fact indicates that our formulation must be complex also when we develop this computation trick.

Let $\Phi = \{\phi_1, \phi_2, \dots, \phi_C\}$ and $\Psi = \{\psi_1, \psi_2, \dots, \psi_N\}$, where $\phi_i = \sqrt{\frac{N_i}{N}}(\mathbf{m}_i - \mathbf{m})$ and $\psi_k = \sqrt{\frac{1}{N}}(\mathbf{x}_k - \mathbf{m}_i)$ ($\mathbf{x}_k \in \mathcal{C}_i$). It can be verified that S_B and S_W can be put in the form of $S_B = \Phi\Phi^T$ and $S_W = \Psi\Psi^T$. Hence, S can be rewritten as $S = XX^T - YY^T$, where $X = \Phi$ and $Y = \sqrt{\beta}\Psi$. Let $Z = [X|jY]$, where j denotes the imaginary unit, we have $S = ZZ^t$. Note that ‘ t ’ denotes the non-conjugate transpose, for example, $(jY)^t = jY^T$ where Y is real.

Z is a $d \times (N + C)$ complex matrix. But as S is real and symmetric, the eigenvectors and eigenvalues of S , $\{\mathbf{w}_i\}$ and $\{\lambda_i\}$, are all real. Here, we show how to compute $\{\mathbf{w}_i\}$ and $\{\lambda_i\}$ from the complex matrix $Z^t Z$, whose size is only $(N + C) \times (N + C)$. First, we do eigen-decomposition on $Z^t Z$:

$$Z^t Z \hat{\mathbf{w}}_i = \hat{\lambda}_i \hat{\mathbf{w}}_i, \quad (3.14)$$

which is of much lower computation than directly solving eigen-decomposition of S . Multiplying Z to both sides of Eq. (3.14), we have

$$ZZ^t(Z\hat{\mathbf{w}}_i) = \hat{\lambda}_i(Z\hat{\mathbf{w}}_i). \quad (3.15)$$

Thus, we can see that $\hat{\lambda}_i = \lambda_i$, and $Z\hat{\mathbf{w}}_i = \alpha\mathbf{w}_i$, where α might be a real or complex constant.

Because Z and $Z^t Z$ are complex, $\hat{\mathbf{w}}_i$ and hence $Z\hat{\mathbf{w}}_i$ could be complex. Without loss of generality, we assume $Z\hat{\mathbf{w}}_i = \mathbf{a} + j\mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ are real vectors. Eq. (3.15) now becomes $ZZ^t\mathbf{a} + jZZ^t\mathbf{b} = \hat{\lambda}_i\mathbf{a} + j\hat{\lambda}_i\mathbf{b}$, which suggests that

$$ZZ^t\mathbf{a} = \hat{\lambda}_i\mathbf{a} \quad \text{and} \quad ZZ^t\mathbf{b} = \hat{\lambda}_i\mathbf{b}. \quad (3.16)$$

Therefore, $\mathbf{a}$ and $\mathbf{b}$ can either be a zero vector, or a real multiple of the eigenvector $\mathbf{w}_i$. Either case, we can compute $\mathbf{w}_i$ from $Z\hat{\mathbf{w}}_i$ as

$$\mathbf{w}_i = \mathbf{v} / \text{norm}(\mathbf{v}) \quad (\mathbf{v} = \text{real}(Z\hat{\mathbf{w}}_i) + \text{imag}(Z\hat{\mathbf{w}}_i).) \quad (3.17)$$

Equations Eq. (3.14) and (3.17) together give us an efficient way to extract the linear MMDA features directly from a high-dimensional input space when the number of training samples is not as large. It is very useful for image-based pattern classification problems such as face recognition, whose input data are arrays of pixel values with the dimension going up to hundreds of thousands. With this trick, we can directly extract MMDA features from high dimensional input space without loss of class discriminant information. Whereas, other techniques as previously mentioned cannot achieve direct dimension reduction of LDA on the original input space and often have to introduce new parameters.

3.4 Margin Improvement via Subspace Approach

In this section, we analyze how our proposed discriminant function can in principle be used to maximize the geometric margin for the use in the subspace methods. In practice though, due to the finite sample size available, we can only approach the solution of maximum margin for the linearly separable case. With the non-separable case, it can find the directions with least overlap.

Vapnik et. al. [133] [20] [132] proposed the Principle of Structural Risk Minimization as the means for best generalization of a learning algorithm. In SVM, for pattern classification problems, this means finding the optimal hyperplane that maximally separates two classes. The SVM computation is very expensive and many have offered a myriad of solutions [62] [108] [36] to ease the computational load. They are basically non-linear iterative methods. We wonder if it is possible to endow subspace methods such as PCA and LDA with margin maximization ability so that we can take the advantage of their simplification. Also, with the subspace methods, we can have a direct formulation for multi-class problems (though not so optimal) without decomposing into a series of binary classifiers as what SVM usually does. Obviously, direct PCA and LDA cannot achieve margin maximization since they are linear methods, while the formulation of the margin is non-linear.

Recall Eq. (3.3), the margin between two classes, along a feature direction $\mathbf{w}$, can be calculated as $\mathcal{M}_{i,j} = |\mu_i - \mu_j| - b(\sigma_i + \sigma_j)$, where μ denotes the class mean, σ denotes the standard deviation of the class samples, and b is a spread regulator that is to depict the class boundaries. If all classes follow the same type of symmetric, uniform distribution, the formula above gives the exact margin. In real applications, such distributions are ideal. If we use the discriminant quantities such as the between-class scatter matrix, S_B , and the within-class scatter matrix, S_W , derived from the input data, an additive discriminant function, $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$ ($\beta = b^2$), has been constructed in Section 3.2, which tries to approximate the margin. In terms of μ and σ , the approximated margin by this criterion was shown to be $\hat{\mathcal{M}}_{i,j} = |\mu_i - \mu_j|^2 - \beta(\sigma_i^2 + \sigma_j^2)$ (c.f. Eq. (3.4)).

For MMDA to yield the true margin $\mathcal{M}_{i,j}$, we need to have

$$\mathcal{M}_{i,j} = \hat{\mathcal{M}}_{i,j} \quad (3.18)$$

i.e. $|\mu_i - \mu_j| - b(\sigma_i + \sigma_j) = |\mu_i - \mu_j|^2 - \beta(\sigma_i^2 + \sigma_j^2)$. or

$$\beta = \frac{|\mu_i - \mu_j|^2 - |\mu_i - \mu_j| + b(\sigma_i + \sigma_j)}{(\sigma_i^2 + \sigma_j^2)} \quad (3.19)$$

Hence, for each specific case, we can find a value of β that satisfies Eq. 3.18.

But it should be noted that β is a function of the class distributions $(\mu_i, \mu_j, \sigma_i, \sigma_j)$ and b , all of them depend on $\mathbf{w}$. Therefore, to find the optimal $\mathbf{w}$ that yields the maximum margin, the value of b is needed. But b depends on $\mathbf{w}$. This makes the solution intractable and b must be searched for, resulting in a one dimensional search. The obvious way as adopted is to use validation to determine b for the optimal direction for maximum margin of separation. Equivalently, by Eq. 3.19, we can search for β instead.

In practice, the validation process may not lead to the optimal value of β because of the finite sample size. Moreover, cross validation does not guarantee optimal solutions though it can lead to margin improvements. With denser input data, it is expected that β will converge to the optimal. Thus in principle, our MMDA can enable one to find the optimal $\mathbf{w}$ for maximum margin but in practice this is not guaranteed depending on the data available.

3.5 Non-Linear MMDA

An important requirement in all classifier designs is that they should be able to handle non-linearly separable cases. SVM will not be useful if it cannot be “kernelised”. In consequence, a kernel transform is not useful unless it can use the Kernel Trick, meaning that the classifier training only should need the inner product terms of the input data. And so, we should also show that MMDA can be kernelised and that the kernel trick is valid for MMDA as well.

Since the MMDA formulation only involves dot products of data, we can extend MMDA into non-linear feature space using the famous kernel trick [10] [114] [115]. Let us consider a non-linear mapping Φ of input data into a feature space $\mathcal{H}$

$$\Phi : \mathcal{X} \rightarrow \mathcal{H}, x \mapsto \Phi(x), \quad (3.20)$$

which could end up in extremely high dimensionality. The kernel trick makes it possible to avoid the intensive computation of $\Phi(x)$ by computing the dot product of $\Phi(x_i)$ and $\Phi(x_j)$ as a kernel function

$$K_{i,j} = K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle. \quad (3.21)$$

K can be any Mercer Kernel such as polynomial, sigmoidal or Gaussian kernels, which implicitly defines the non-linear mapping Φ . In this section, we show our original derivation of how to formulate MMDA in non-linear space using this kernel trick.

We re-write the margin-based discriminant function in feature space $\mathcal{H}$ as

$$J(\mathbf{w}) = \mathbf{w}^T (S_B^\Phi - \beta S_W^\Phi) \mathbf{w}, (\mathbf{w} \in \mathcal{H}). \quad (3.22)$$

Now, the between-class and within-class scatter matrices in $\mathcal{H}$ become

$$S_B^\Phi = E_i\{(\mathbf{m}_i^\Phi - \mathbf{m}^\Phi)(\mathbf{m}_i^\Phi - \mathbf{m}^\Phi)^T\}, \quad \text{and} \quad (3.23)$$

$$S_W^\Phi = E_i\{E_k\{(\Phi(\mathbf{x}_k) - \mathbf{m}_i^\Phi)(\Phi(\mathbf{x}_k) - \mathbf{m}_i^\Phi)^T\}\} \quad (\forall \mathbf{x}_k \in \mathcal{C}_i). \quad (3.24)$$

The solution of MMDA features in $\mathcal{H}$ can be obtained by finding the eigenvectors of $(S_B^\Phi - \beta S_W^\Phi)$. But to avoid computing $\Phi(\cdot)$, we need to transform Eq. (3.22) into a dual problem so that the kernel functions can be used instead.

Since all $\mathbf{w}$ with nonzero eigenvalues lie in the span of $\Phi(\mathbf{x}_1), \dots, \Phi(\mathbf{x}_N)$, there exist coefficients $\alpha_l (l = 1, \dots, N)$ such that

$$\mathbf{w} = \sum_{l=1}^N \alpha_l \Phi(\mathbf{x}_l). \quad (3.25)$$

Substitute Eq. (3.25) to compute the projection of $\mathbf{m}_i^\Phi$, we have

$$\begin{aligned} \mathbf{w}^T \mathbf{m}_i^\Phi &= \sum_{l=1}^N \alpha_l \Phi(\mathbf{x}_l)^T E_k\{\Phi(\mathbf{x}_k)\} \quad (\forall \mathbf{x}_k \in \mathcal{C}_i) \\ &= \sum_{l=1}^N \alpha_l E_k\{\langle \Phi(\mathbf{x}_l), \Phi(\mathbf{x}_k) \rangle\} \\ &= \sum_{l=1}^N \alpha_l E_k\{K_{l,k}\} \\ &= \boldsymbol{\alpha}^T \tilde{\mathbf{m}}_i, \end{aligned} \quad (3.26)$$

where $\boldsymbol{\alpha}$ and $\tilde{\mathbf{m}}_i$ are l -dimensional vectors with $(\boldsymbol{\alpha})_l = \alpha_l$ and $(\tilde{\mathbf{m}}_i)_l = E_k\{K_{l,k}\}$. Similarly,

$$\begin{aligned} \mathbf{w}^T \mathbf{m}^\Phi &= \sum_{l=1}^N \alpha_l \Phi(\mathbf{x}_l)^T E_j\{\Phi(\mathbf{x}_j)\} \quad (j = 1, \dots, N) \\ &= \sum_{l=1}^N \alpha_l E_j\{\langle \Phi(\mathbf{x}_l), \Phi(\mathbf{x}_j) \rangle\} \\ &= \sum_{l=1}^N \alpha_l E_j\{K_{l,j}\} \\ &= \boldsymbol{\alpha}^T \tilde{\mathbf{m}}, \end{aligned} \quad (3.27)$$

where $\tilde{\mathbf{m}}$ is also an l -dimensional vector with $(\tilde{\mathbf{m}})_l = E_j\{K_{l,j}\}$. Also, the projection of $\Phi(\mathbf{x}_k)$ onto $\mathbf{w} \in \mathcal{H}$ is

$$\begin{aligned} \mathbf{w}^T \Phi(\mathbf{x}_k) &= \sum_{l=1}^N \alpha_l \Phi(\mathbf{x}_l)^T \Phi(\mathbf{x}_k) \\ &= \sum_{l=1}^N \alpha_l \langle \Phi(\mathbf{x}_l), \Phi(\mathbf{x}_k) \rangle \\ &= \sum_{l=1}^N \alpha_l K_{l,k} \\ &= \boldsymbol{\alpha}^T \tilde{\mathbf{x}}_k, \end{aligned} \quad (3.28)$$

where $(\tilde{\mathbf{x}}_k)_l = K_{l,k}$.

Substitute Eq. (3.26), ((3.27)) and ((3.28)) into Eq. (3.23) and (3.24), the dual of $\mathbf{w}^T S_B^\Phi \mathbf{w}$ and $\mathbf{w}^T S_W^\Phi \mathbf{w}$ can be obtained as

$$\begin{aligned} \mathbf{w}^T S_B^\Phi \mathbf{w} &= \mathbf{w}^T E_i\{(\mathbf{m}_i^\Phi - \mathbf{m}^\Phi)(\mathbf{m}_i^\Phi - \mathbf{m}^\Phi)^T\} \mathbf{w} \\ &= \boldsymbol{\alpha}^T E_i\{(\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})(\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})^T\} \boldsymbol{\alpha} \\ &= \boldsymbol{\alpha}^T \tilde{S}_B \boldsymbol{\alpha}, \quad \text{and} \end{aligned} \quad (3.29)$$

$$\begin{aligned} \mathbf{w}^T S_W^\Phi \mathbf{w} &= \mathbf{w}^T E_i\{E_k\{(\Phi(\mathbf{x}_k) - \mathbf{m}_i^\Phi)((\Phi(\mathbf{x}_k) - \mathbf{m}_i^\Phi)^T)\} \mathbf{w} \quad (\forall \mathbf{x}_k \in \mathcal{C}_i) \\ &= \boldsymbol{\alpha}^T E_i\{E_k\{(\tilde{\mathbf{x}}_k - \tilde{\mathbf{m}}_i)(\tilde{\mathbf{x}}_k - \tilde{\mathbf{m}}_i)^T\} \} \boldsymbol{\alpha} \\ &= \boldsymbol{\alpha}^T \tilde{S}_W \boldsymbol{\alpha}. \end{aligned} \quad (3.30)$$

Now, $\tilde{S}_B$ and $\tilde{S}_W$ only involve kernel terms, therefore, the direct computation of $\Phi(\cdot)$ is completely avoided.

Substitute Eq. (3.29) and (3.30) into Eq. (3.22), the final dual of the margin-based discriminant function in $\mathcal{H}$ become

$$J = \boldsymbol{\alpha}^T (\tilde{S}_B - \beta \tilde{S}_W) \boldsymbol{\alpha}. \quad (3.31)$$

Similar to the linear formulation, $(\tilde{S}_B - \beta \tilde{S}_W)$ remains symmetric but may not be positive. The non-linear MMDA features can thus be solved in terms of $\boldsymbol{\alpha}$ as the eigenvectors of $(\tilde{S}_B - \beta \tilde{S}_W)$, which approximately scales in $O(N^2)$. For very large datasets, sparse solutions can be exploited to make the computation viable.

Based on the discussions above, the non-linear projections of input data to the feature space, $\mathbf{y}_k = \mathbf{w}^T \Phi(\mathbf{x}_k)$, can be computed using Eq. (3.28) given a choice of a kernel function. From this point onwards, the classification can be performed using standard classifiers such as k-Nearest Neighbor Rule and SVM. This extension of MMDA into the non-linear space allows us to use the method with more flexibility. But in the experiment section below, we will concentrate on the performance of linear MMDA to reveal its fundamental properties.

Figure 3.5: Margin obtained by MMDA, SVM and LDA - toy data

3.6 Experiments

3.6.1 A Toy Data Example

First, we evaluate the performance of linear MMDA on a separable two-class toy problem, and compare it to the linear SVM and the conventional LDA. Fig. 3.5 shows the separation hyperplane obtained by the three methods respectively, with the corresponding margin values indicated. The samples are of two dimensional data and the two classes are labeled by crosses and circles respectively. For MMDA and LDA, the hyperplane plotted is perpendicular to their feature directions. The margins are computed as the minimal distance between the opposite class samples as perpendicular to the hyperplane direction. For information, the nearest two samples from opposite classes are indicated by green triangles.

From Fig. 3.5, it is observed that the maximal margin obtained by MMDA is 0.095 at $\beta = 7.8$. SVM achieves a similar margin of 0.094, while LDA can only find a feature with $margin = 0.057$. This simple example evidently suggests that MMDA adds the margin maximization possibility into subspace methods. The proper selection of β value can generate optimal hyperplane with the largest margin. Whereas for LDA, such properties have never been taken into consideration.

3.6.2 Experiments on ORL, UMIST and YALE Face Databases

We now evaluate the performance of MMDA for real face recognition problems using the ORL [32], the UMIST [33] and the YALE [34] face databases. The ORL and the UMIST face images are 112×92 pixels, while the YALE face images are 231×195 pixels. We use the proposed computation trick in Section 3.3.1 to extract the MMDA features. The performance of MMDA is compared to the conventional two-stage LDA as proposed in [4], where PCA was used to reduce the input space dimension so that S_W can become nonsingular. To make the results less dependant of the training samples, all the results reported are averaged over 100 random runs, where for each run, the training samples are randomly selected from the database and the rest of the images are used for classification. Here, we focus on the first $C - 1$ features of MMDA, where C is the total number of classes, since this is the size limit of the LDA features that we compare to. Classification is done by the

(a) ORL Database

(b) UMIST Database

(c) YALE Database

Figure 3.6: Sample Images from ORL, UMIST and YALE face databases.

simple nearest neighbor method with the L_2 norm-based distance.

Fig. 3.6 shows some sample images from the three face databases. The ORL face database [32] consists of 40 persons (classes), each with 10 images, which are taken at different times, varying the lighting, facial expressions and some artifacts, such as with or without glasses. The subjects are usually in an upright, frontal position, but with small angle pose changes. The UMIST face database [33] consists of 575 gray-scale images of 20 individuals(classes), where the number of images per person varies from 19 to 48. The database covers a wide range of sex/race/appearance, and it also

Figure 3.7: Recognition performance on ORL Database

accounts for pose variations from profile to frontal views. All the images are cropped to 112×92 pixels. The YALE face database [34] contains 165 grayscale images of 15 individuals. There are 11 frontal-view images per subject covering different facial expressions (such as happy, sad, sleepy) and lighting conditions (such as center-light, left-light). Comparing to the ORL database, UMIST and YALE are of more drastic intra-class variations.

The overall performance

First, we evaluate the performance of MMDA under different β and compare it to that of LDA. Fig. 3.7 to Fig. 3.9 summarize the results of the two methods for the three databases at the optimal feature size (i.e. the number of features that achieves the highest classification accuracy on average). In each subplot, the upper curve shows the accuracy rate with the 1 *std* performance deviation over the 100 random runs, while the lower bar graph gives the corresponding optimal feature size. For MMDA plots, we highlight the three points where $\beta = -1$ (PCA equivalent), $\beta = 1$ (MMC) and $\beta = 9$ (A general setting suggested for MMDA). Whereas, for LDA plots, the dimensionality after the PCA dimension reduction, K_{pca} , has been varied incrementally, and the best possible performance achieved by this method is highlighted.

From Fig. 3.7 to Fig. 3.9, we can see that the performance of MMDA becomes stable for large β and achieves best performance at $\beta = 10$ for ORL database and $\beta = 30$ for both UMIST and YALE databases. In all the three cases, the best performance does not deviate much from the results for $\beta = 9$.

Figure 3.8: Recognition performance on UMIST Database

Figure 3.9: Recognition performance on YALE Database

For LDA, the best possible performance degrades with increasing K_{pca} , and the differences are especially significant for ORL and UMIST databases. The optimal setting of K_{pca} is 40 for ORL database, 30 for UMIST database and 45 for YALE database. This is an interesting finding as it is contradictory to the common sense that with higher dimension preserved from the input space (larger K_{pca}), less discriminant information is lost and LDA should be able to produce better feature sets

for classification. One possible explanation is that when K_{pca} is large but can still make S_W non-singular, the solution of LDA is unstable as S_W could still be close to singular. The experiment results suggest that the two-stage LDA can only give reasonable results when S_W^{-1} is well defined, i.e. with low enough data dimension comparing to the number of samples for the LDA step. However, such dimension is hard to estimate and will affect the performance significantly. From results in Fig. 3.7 to Fig. 3.9, we can also see that the optimal K_{pca} value deviates a lot from Belhumeur et. al.'s suggested setting of $N - C$ [4] (where N is the total number of samples, and C is the number of classes.). The optimal K_{pca} has to be found from cross-validation with the training data in order to optimize LDA's performance.

Comparing the recognition rates, we can see that MMDA (just at a general setting of $\beta = 9$) consistently performs better than LDA's best possible results. Table 4.1 summarizes the accuracy of the two methods at the highlighted points of Fig. 3.7 to Fig. 3.9. We can see that MMDA at $\beta = 9$ consistently performs better than LDA at its optimal setting across all the three databases. To view the statistical significance of this difference, we conducted a one-tailed t Test over the experimental data. The degree of freedom is 198 as each method has been tested over 100 random runs. Table 3.2 summarizes the t Test results, and we can see that the confidence level of this difference is quite high, especially for ORL and YALE databases. For example, For ORL database, there are 99.99% chances that MMDA($\beta = 9$) is better than LDA(K_{pca}^*). Although for UMIST database, the improvement of MMDA over LDA is not so significant both in terms of the difference value and the confidence level, there is still some room to improve MMDA by choose the optimal β as shown in Fig. 3.8. So ultimately, it is possible for the UMIST database to demonstrate better statistical significance on the improvement of MMDA over LDA.

The β setting effects

Looking into the sensitivity of the MMDA performance over β , we can see that larger β , e.g. $\beta > 4$ achieves stable performance on all the three databases. However, searching for the optimal β by cross validation can still slightly boost up the MMDA performance. But if we want to avoid this, a general setting of $\beta = 9$ can be recommended as it gives close to optimal results.

Comparing to other subspace methods such as PCA and MMC in [82] that can be included in the MMDA framework in the view of different β settings, we can see that MMDA with large β consistently performs better. This shows the importance of β in regulating the searching of the feature space towards the large margin directions. For PCA (equivalent to MMDA with $\beta = -1$), the discriminant function actually measures the total data scatter. The feature sets are not purposely fed with any class-discriminant information, and thus gives bad classification results. For MMDA (equivalent to MMDA with $\beta = 1$) the classification accuracy is much better than PCA, but as $\beta = 1$ allows too many samples to be outliers, the resultant feature sets are still quite different from the large margin directions. This is similar to a bad setting of C in the soft margin SVM, which could seriously affect the searching of

Table 3.1: Classification Accuracy (%) of MMDA and LDA at the Highlighted Points of Fig. 3.7 to Fig. 3.9

	MMDA			LDA
	$\beta = -1$	$\beta = 1$	$\beta = 9$	K_{pca}^*
ORL	93.99/1.70	96.00/1.61	96.81 /1.33	96.07/1.50
UMIST	91.33/2.28	96.54/1.79	97.46 /1.50	97.33/1.42
YALE	80.10/5.22	93.81/3.88	98.18 /1.49	97.63/1.86

The value after / is the performance standard deviation over 100 runs.

Table 3.2: t Test to compare MMDA($\beta = 9$) and LDA(K_{pca}^*) at the optimal feature size

t Test	ORL	UMIST	YALE
t	4.38	0.76	2.99
$ConfidenceLevel > (\%)$	99.99	78	99.8

the optimal hyperplane. As shown in Fig. 3.7, the best feature size is 18 for $\beta = 1$ and 9 for $\beta = 9$. Thus, $\beta = 1$ and $\beta = 9$ may generate very different sets of features, though their performance difference is not very large.

For large β , the performance of MMDA is stable, but small variation still exists. For ORL, the performance slightly drops when $\beta > 10$, because a larger β may over-punish the feature space searching, and hence miss some useful features. But for UMIST and YALE, where the class distributions are known to be more sparse than ORL, $\beta = 30$ to 100 gives the best results, as larger class spread is now expected.

Varying the feature size

We now check the performance of MMDA and LDA at various feature sizes. We compare MMDA at its general setting $\beta = 9$ to LDA at its optimal settings for the three databases. All the features are ranked in the order of decreased eigenvalue.

Figure 3.10: Performance of MMDA and LDA at varied feature size - ORL Database

As shown in Fig. 3.10 to Fig. 3.12, the performance of MMDA is slightly better than that of LDA on all the three databases. Although LDA can generate totally $C - 1$ features, the optimal feature size is usually far less than that, and using more features actually deteriorate the classification results (e.g. Fig. 3.10 to Fig. 3.11). This phenomenon is often explained as overfitting of the feature sets to the training samples. In Fig. 3.12, the overfitting of LDA is not observed, simply because the optimal feature size is 12, which is too close to the total feature size of LDA.

For MMDA, it is clear that there is no obvious trend of overfitting when features with smaller margins are added for classification. This gives an indication that the MMDA features, being orthogonal to each other, contain no mutual information and are more constructive in forming discriminant feature space. However, the LDA features, being non-orthogonal sets, contains much mutual information along the feature directions, and cannot achieve the best performance by simply adding up all the features.

Interestingly, in Fig. 3.10 and Fig. 3.12, the performance of MMDA on ORL and YALE database is much better than that of LDA for the first few features. This evidently shows that the MMDA features which try to maximize margins are more effective for classification comparing to the LDA features that only maximizes the normalized distance between class centers. However, for the UMIST database (Fig. 3.11), this advantage is not so obvious, which might be due to the sub-optimal β setting that produces features with poorer margins.

For databases with large within-class variation, such as the UMIST database, we also suggest to form sub-clusters of the classes for classification. In this way, both LDA and MMDA can better preserve the discriminant information and it is also the

Figure 3.11: Performance of MMDA and LDA at varied feature size - UMIST Database

current trend for practical face recognition applications.

3.7 Conclusions

In this chapter, we proposed an alternative form of discriminant analysis for general multi-class problems, which is called Margin-Maximization Discriminant Analysis (MMDA). It is based on an additive form discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$, which is shown to be able to approximately maximize the average margin between the classes. Comparing to the conventional quotient form discriminant of LDA, the new discriminant function naturally avoids the singularity problem of S_W that is prohibitive in LDA. Moreover, it can directly extract class discriminant features from high dimensional input space, which avoids the possible loss of discriminant information, and is also computationally more effective.

The proposed MMDA is found to be a good replacement of LDA when dealing with Small Sample Size (SSS) problems. The class-spread regulator β appeared in the discriminant function provides an extra degree of freedom over LDA, which enables us to search for features with better margin between the classes. When $\beta = 1$, our MMDA becomes equivalent to a similar method MMC proposed by Li et. al. [82]. We show that large β (e.g. $\beta \geq 9$) is enough to regulate feature search towards the ones with better margins. The value of β can be optimized through cross validation. For non-separable cases, β functions like the C or ν parameters in soft margin SVMs, which help prevent the outliers from affecting the optimal

Figure 3.12: Performance of MMDA and LDA at varied feature size - YALE Database

feature sets.

In this chapter, we also enclose a computation trick for MMDA which makes it more effective for high dimensional input data. To cater for non-linearly separable situations, non-linear MMDA is derived using the well-known “kernel trick”. The experiments with linear MMDA show that it can produce features with larger margins compared to LDA, which may even approach the optimal margin at a proper β setting. Overall, MMDA can achieve better performance than LDA. Moreover, its performance becomes quite stable for large enough β , whereas for LDA, the performance is very sensitive to certain parameter that is used to combat the singularity problem.

It has been shown that MMDA is amenable to Kernel Transformations. In the future, we may look into the performance of nonlinear MMDA and evaluate the performance of the MMDA-based methods on more benchmark data sets for classification problems. For databases with large within-class variation, for example, those follow multi-modal distribution, sub-clustering of the class samples can also be tried to see whether this can further boost up the MMDA performance.

Chapter 4

Relevance-Weighted Discriminant Analysis

4.1 Introduction

In Chapter 3, the performance of Linear Discriminant Analysis (LDA) was compared based on its best feature size. However, as revealed in Section 3.6.2, the performance of LDA suffers from the “Peaking Phenomenon”, where the classification rates of LDA starts to drop after the “peak point” when subsequent features are included for classification. This peaking phenomenon has been reported in many research papers such as [31] [109] [59], and is often attributed to overfitting. The common solution to search for the optimal feature size is by cross-validation.

In fact, the peaking phenomenon is not limited to LDA, but are common to all classifiers, since overfitting is a salient issue for classifier design. In [59], massively parallel computation is employed to find the optimal number of features as a function of sample size. They tested on various classifiers including 3-Nearest Neighbor (3-NN), linear/nonlinear SVM, perceptron and LDA, and concluded that SVM exhibits strong robustness with respect to large feature sets and is superior to LDA which is not robust. However, they pointed out that their investigations did not find the optimal feature size but only close to it. We discovered in early experimentations with LDA that the peaking phenomenon did not occur when the LDA features were weighted in a certain manner. Motivated, we set out to make rigorous confirmation of this. Further experiments confirmed our observations and led us to wonder why such weightages worked. In this chapter, we give a theoretical explanation and also applied the resultant weightage scheme to face recognition using three different well known databases. The key finding lies with the meaning of feature relevance.

We examine the overfitting problem of LDA through the insights of its feature (LDA directions) properties. We can see that certain LDA features invariably possess very large within-class variations. Depending also on the value of the between-class scatter, this large variation often contributes to the large overlap between classes. This is often the case for LDA features with smaller Fisher discriminant value λ_k . When $\lambda_k < 1$, within-class variation is larger than the between-class vari-

ation. Adding features with $\lambda_k \ll 1$ for distance metric-based classification would seemingly deteriorate classification results. Also, from information point of view, LDA features are non-orthogonal sets that capture the most class-discriminant directions. They may contain mutual information of projected data and using them with equal weightage in a classification metric has to be re-examined.

Based on these considerations, we suggest a novel view of classifier overfitting. We will show that as long as a feature is “relevant”, the performance can be further enhanced by its inclusion. The term “relevant feature” will be explained later. We are effectively stating that the peak observed by others need not happen as long as there are more relevant features available. Our claim is that by assigning weights according to the degree of relevance of the features, it is possible for the classifier to monotonically improve the classification performance.

We propose to use a novel “relevance” measure to scale all the LDA features and bring them into a comparable basis commensurate with their individual relevance. This Relevance-Weighted Discriminant Analysis (RWDA) will be shown to be effective in alleviating the peaking phenomenon, and enable better generalization than traditional LDA. By this approach, we show that it is possible to overcome the overfitting problem even for classifiers trained with small sample size. A theoretical justification of such feature weighting scheme is also given based on the probability of classification error on weighted and unweighted features.

Feature relevance has been an important concept in pattern recognition and machine learning, which often refers to how relevant a feature is to a target concept. It is often used for the purpose of feature selection, with either “hard” or “soft” values (c.f. [78]). A hard relevance indicates if the feature is selected or not, and is used to select an optimal subset of features. Such relevance measure is often done through heuristic search of all feature subset combinations through some statistical analysis. A soft relevance measure, however, refers to a range of values (often in percentage) that indicates the significance of the features for classification ([9] [78]). They are usually used in feature weighing of various classifiers, and the relevance weights may come from conditional probability distributions ([118] [126]), information-theoretic metric ([24]), or even an exhaustive search of discretized weight space ([70]). Recent researches have extended the feature weighting into unsupervised learning paradigm, where EM algorithms are popularly used to estimate feature weights ([47] [78]). In [95], a score related to the Fisher discriminant has also been used to weigh features for k-means clustering. We propose a different way of looking at the “soft” feature relevance that is appropriate in the context of classification performance.

Our Definition of Feature Relevance: *A feature is relevant if there exists at least one non-overlapping region of projected data between two or more classes along the feature. The greater the non-overlap, the more relevant is the feature.*

Based on this definition, we propose our relevance measure using a novel model of non-overlapping regions of projected class distribution. For LDA features, it can

be analytically obtained from Fisher discriminants without much additional computation. This relevance measure is purely based on the actual sample distribution and requires no heuristic search or probabilistic inference. As for the kernel extension of discriminant analysis, our relevance measure does not pose any limitation and can be readily extended into kernel space.

It shall be noted that our proposed RWDA is completely different from the recently proposed Relevance Weighted LDA of Tang and Suganthan et. al. [122]. In [122], the relevance based weights are applied to each class when computing within-class scatter (S_W) to reduce the influence of outlier classes in the final LDA features. Such weights is inversely related to how well a class is separated from the rest in the original space, which in effect concentrate the LDA features to those classes that are more difficult to classify (the non-outliers). Therefore, their approach focuses on finding LDA features of weighted scatter matrices, while our approach puts relevance weights on the features derived from the unweighted LDA. Hence, the motivations and underlying principles are different. Another point to note is that their method is iterative while ours, based on the by-product of the solution to the maximization of the Fisher discriminant, is analytic.

This chapter is organized as follows. First, a model of non-overlap class regions is proposed based on LDA's geometric property in Section 4.2. This leads to the derivation of a simple feature relevance measure in Section 4.3. In Section 4.4, the RWDA method is proposed together with a first version called wLDA, which provides an insight into the evolution to our final RWDA algorithm. In Section 4.5, a theoretical justification for using the relevance weightage is presented by analyzing the probability of classification error on weighted and unweighted features. Several experiments are investigated to demonstrate the advantages of the RWDA over LDA and PCA based on UMIST, ORL and YALE Face Databases. Conclusions are presented in the last section.

4.2 The Estimation of Non-Overlapping Class Regions

4.2.1 Introduction

Subspace methods like LDA extract features as projection directions and project samples onto them for representation. The usefulness of a feature to classification is thus highly related to the non-overlapping regions between two or more projected class distributions. In this section, we look into the geometric meaning of Fisher discriminant, λ , of LDA, and propose a general way to estimate the non-overlapping class regions on any feature (not limited to LDA) based on statistical class distribution parameters. We will then show that such an estimation model can be derived from Fisher discriminant values, which involves no additional computation for LDA. This is an advantage over other weighted LDA methods mentioned before. The key idea of the relevance weightage is its direct relation to the eigenvalues of the LDA

solutions. To see why, the insight into the geometrical meaning of these eigenvalues has to be explored.

4.2.2 The Geometric Meaning of λ

Recall the Fisher Discriminant Function

$$\mathcal{J}(\mathbf{w}_k) = \frac{\mathbf{w}_k^T S_B \mathbf{w}_k}{\mathbf{w}_k^T S_W \mathbf{w}_k} = \lambda_k, \quad (4.1)$$

where S_B and S_W are between- and within-class scattering matrices. λ_k represents the Fisher discriminant value of feature $\mathbf{w}_k$, which can be solved together with $\mathbf{w}_k$ from a generalized eigenvalue problem $S_B \mathbf{w}_k = \lambda_k S_W \mathbf{w}_k$. For a C -class problem, we can obtain at most $C - 1$ LDA features $\mathbf{w}_k$ with non-zero λ_k , where the eigenvalues are ranked from highest to lowest with increasing index k . In this section, we will show how to associate the Fisher discriminant value λ to the average projected class mean (μ_C) and standard deviation (σ_C) on a LDA feature vector. Note that μ_C is originated from the global mean of all the samples.

The projection of a sample $\mathbf{x}_j$ onto a feature $\mathbf{w}_k$ is simply computed by linear transformation $\mathbf{y}_j^{(k)} = \mathbf{w}_k^T \mathbf{x}_j$. Recall the formula of Between-Class and Within-Class Scatters for sample patterns $\{\mathbf{x}_j\}$ (cf. Eq. (3.1) and (3.2))

$$S_B = E_i\{(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T\}, \quad \text{and} \quad (4.2)$$

$$S_W = E_i\{E_j\{(\mathbf{x}_j - \mathbf{m}_i)(\mathbf{x}_j - \mathbf{m}_i)^T\}\} \quad (\mathbf{x}_j \in \mathcal{C}_i), \quad (4.3)$$

where $\mathbf{m}$ is the global mean, $\mathbf{m}_i$ is the mean of class i , and $E_i\{\cdot\}$ denotes the expectation of the variables for all i . The projected between-class and within-class scatters on feature $\mathbf{w}_k$ can be defined as

$$\Delta_k = \mathbf{w}_k^T S_B \mathbf{w}_k, \quad \text{and} \quad (4.4)$$

$$\Omega_k = \mathbf{w}_k^T S_W \mathbf{w}_k, \quad (4.5)$$

Substituting Eq. (4.2) and (4.3) into them, we can have

$$\Delta_k = E\{[\mathbf{w}_k^T(\mathbf{m}_i - \mathbf{m})]^2\} = E\{\mu_i^2(k)\} = \mu_C^2, \quad \text{and} \quad (4.6)$$

$$\Omega_k = E\{[\mathbf{w}_k^T(\mathbf{x}_j - \mathbf{m}_i)]^2\} = E\{\sigma_i^2(k)\} = \sigma_C^2, \quad (4.7)$$

where $\mu_i(k)$ and $\sigma_i(k)$ are the projected mean and standard deviation of Class i on $\mathbf{w}_k$, and μ_C and σ_C are approximately the average projected class mean and standard deviation on $\mathbf{w}_k$ respectively. Note that due to the subtraction of $\mathbf{m}$ in Eq. (4.6), both $\mu_i(k)$ and μ_C originate from the projection of the global mean of all the samples on the feature direction in question. Fig. 4.1 illustrates the geometric meaning of projected class mean and standard deviation using a simple two class example. They give good indication of class distributions along each feature vector.

4.2 The Estimation of Non-Overlapping Class Regions

46

Figure 4.1: Projected Class Mean and Standard Deviation on LDA Feature Vector

With the equation above, we can now associate Fisher discriminant λ_k to the average projected class mean and standard deviation on $\mathbf{w}_k$:

$$\lambda_k = \frac{\Delta_k}{\Omega_k} = \frac{\mu_C^2}{\sigma_C^2}. \quad (4.8)$$

As compared to λ_k values, the knowledge of the values of Δ_k and Ω_k provides us with more insights into the class-distribution along the LDA features. The following are findings from some investigations that we did. Fig. 4.2 and 4.3 give some typical values of λ_k , Δ_k and Ω_k for the full span of the LDA features as obtained from the UMIST and the ORL face databases. $\{\lambda_k\}$ is obtained along with solving the Fisher Discriminant Function for $\{\mathbf{w}_k\}$, while $\{\Delta_k\}$ and $\{\Omega_k\}$ can be easily computed once $\{\mathbf{w}_k\}$ are determined.

From Fig. 4.2 and 4.3, as λ_k monotonically decreases from the first to the last feature, Δ_k gradually drops from a relatively large value to zero, while Ω_k slowly grows which may eventually become much larger than the corresponding Δ_k values. Though the changes of Δ_k and Ω_k are not strictly monotonic, their overall trends are. Typically and especially for multi-class problems, the trend of Δ_k tends to be monotonically decreasing with respect to λ_k while the corresponding Ω_k tends to be increasing.

The monotonic trends of Δ_k and Ω_k suggest some general properties of LDA features on class distributions. For the first few features where $\Delta_k \gg \Omega_k$, i.e. $\mu_C \gg \sigma_C$, the distance between the class centers is generally large, and the within-class data variability is small on average. For example, one can refer to $\mathbf{w}_1$ in

4.2 The Estimation of Non-Overlapping Class Regions

47

(a) Values of λ_k (b) Values of Δ_k and Ω_k

Figure 4.2: Typical values of λ_k , Δ_k and Ω_k for UMIST face database (8 images/class trained).

Fig. 4.1. Such a feature means there is good class separation, and the non-overlap regions between two or more classes are large; and maximum when separable. But as $\mathbf{w}_k$ moves towards the end of the feature sequence, the value of Δ_k drops while Ω_k increases. Eventually, we can have $\mu_C \approx 0$ whereas $\sigma_C \gg \mu_C$. For these features, the projected class centers become closer and closer to each other, while the within-class data variability increases gradually. $\mathbf{w}_2$ in Fig. 4.1 gives an example for such features. The overlap among the classes becomes severe on them, but still there are some non-overlapping regions which are far from the projected class centers. They can be helpful in identifying a subset of the class samples.

(a) Values of λ_k (b) Values of Δ_k and Ω_k

Figure 4.3: Typical values of λ_k , Δ_k and Ω_k for ORL face database (5 images/class trained).

The interesting point to draw from the above discussion is the measure of the level of non-overlapping regions for overlap classes and the amount of separation margin for separable classes from the values of Δ_k and Ω_k . We now probe deeper into this.

4.2.3 Modeling the Non-Overlapping Class Regions

First, we look into the non-overlapping regions on a feature $\mathbf{w}_k$ for a simple two-class problem. In a two class non-separable problem, there are regions in each class

4.2 The Estimation of Non-Overlapping Class Regions

49

which do not contain any samples from the other class. These regions are termed the “non-overlapping regions”. We would like to find feature directions onto which the projected class samples give maximum non-overlapping regions. If the distributions are separable, the non-overlapping regions have maximum values. Here, we assume two classes each with unimodal distributions. For multi-modal cases, we must resort to kernel transformations (as with all other classifiers) and hope that in the kernel space, the distributions are unimodal.

Assume sample distributions are i.i.d., the class center is μ_C apart from the global mean, and the class spread is $2\sigma_C$ for both classes. Fig. 4.4 models all the possible cases of class distribution from a) separated classes ($\mu_C \gg 2\sigma_C$), b) just touching classes ($\mu_C \approx 2\sigma_C$), to c) overlapped classes ($\mu_C \ll 2\sigma_C$). In terms of λ , this is equivalent ¹ to a) $\lambda_k \gg 4$, b) $\lambda_k \approx 4$, and c) $\lambda_k \ll 4$.

From Fig. 4.4, it is clear that for cases a) and b), the non-overlapping regions per class can be measured as $4\sigma_C$ for each of the class. For case c), the measure of non-overlapping regions becomes a bit tricky, but consider each class is moved away from global mean by μ_C in opposite directions, this effectively “creates” a non-overlapping region of $2\mu_C$ for each of the class. Therefore, in our simple model, the non-overlapping class regions can be measured in terms of σ_C and μ_C if the class boundary is a known multiple of σ_C from its class center.

For general two-class distributions with the class centers being μ_{Ci} and the class spread being $M\sigma_{Ci}$ around the class center (M is a constant), we can see that the non-overlapping regions become $2M\sigma_{Ci}$ for each class if no overlap occurs. When the two class projections overlap with each other, as what is shown in Fig. 4.5, the non-overlapping regions $\mathfrak{R}_1$ and $\mathfrak{R}_2$ for class 1 and 2 respectively can be exactly computed as

$$\mathfrak{R}_1 = (\mu_{C1} + \mu_{C2}) + M(\sigma_{C1} - \sigma_{C2}), \text{ and} \quad (4.9)$$

$$\mathfrak{R}_2 = (\mu_{C1} + \mu_{C2}) + M(\sigma_{C2} - \sigma_{C1}). \quad (4.10)$$

Thus, on average, the non-overlapping regions per class can still be measured as $2M\sigma_C$ for non-overlapped or just touching classes, and $2\mu_C$ for overlapped classes. Note that μ_C and σ_C represent the average class mean and standard deviation as projected to $\mathbf{w}_k$ respectively.

To generalize to multi-class problems, it is not feasible to look at non-overlapping regions of individual classes, as they may vary much from one class to another. One way to get by this problem is to treat them as a set of one-against-the-rest problems. But this leads to a more complex structure. To solve the multi-class problem efficiently, we can statistically measure the non-overlapping regions in terms of average class distribution, i.e. based on μ_C and σ_C . This, again, leads us back to the measurement model above. But for multi-class case, this model is now measuring

¹Here, it is assumed that the estimated boundary is twice the standard deviation from the center. By Chebychev’s inequality, 12.5% (one-sided) of the samples will overlap with the other class distribution. The boundary could be set at three standard deviations but in practice twice is fine. If the distributions are gaussian, the number of samples outside the boundary will be much less. However, for the present discussion, this is not an issue.

4.2 The Estimation of Non-Overlapping Class Regions

50

Figure 4.4: Non-Overlapping Regions on a Feature w_k for a simple two-class problem. (Assume class distribution is i.i.d. and the spread is $2\sigma_C$.)

Figure 4.5: Non-Overlapping Regions for non-i.i.d. two-class problem (The overlapped case).

the average non-overlapping regions of class distribution as projected on that feature. According to discussion above, this average estimation shall remain accurate for any type of class distributions. The only assumption is that the class boundary can be described as a known or estimated multiple, M , of within-class standard deviation. For certain distribution type, e.g. Gaussian Distribution, $M = 2$ can guarantee that 95% of the samples fall within such a spread range. This equals 2.5% of the samples being treated as class outliers.

4.3 Feature Relevance

Non-overlapping regions are in effect the ‘working part’ of a feature that contributes to classification tasks and should be directly related to feature relevance. They can help distinguish a subset of the classes from the rest. Within this subset, we can then possibly rely on some other features to determine the actual class label. In previous discussion, a feature is relevant if the distributions of the two or more classes contain some non-overlapping regions. The greater the non-overlap, the greater its relevance. Having modeled the non-overlapping region estimation, we now propose a novel analytical measure of feature relevance based on that.

Recall the non-overlapping region estimation of projected class distribution on a feature as proposed in Section 4.2.3,

$$\gamma = \begin{cases} 2M\sigma_C & \mu_C \geq M\sigma_C, \\ 2\mu_C & \mu_C < M\sigma_C, \end{cases} \quad (4.11)$$

where μ_C and σ_C are the average projected class mean and standard deviation respectively, and M is a constant that regularizes the average within-class spread. The normalized non-overlapping region measure that can be compared across features

can be obtained by dividing the right side of Eq. (4.11) by $2\sigma_C$:

$$\bar{\gamma} = \begin{cases} M & \frac{\mu_C}{\sigma_C} \geq M, \\ \frac{\mu_C}{\sigma_C} & \frac{\mu_C}{\sigma_C} < M. \end{cases} \quad (4.12)$$

When $\bar{\gamma}$ equals to M , this means that the distributions do not overlap (i.e. separated). Then the non-overlapping regions are at the maximum.

For feature relevance measure of $\mathbf{w}_k$, we replace $\frac{\mu_C}{\sigma_C}$ by $\sqrt{\lambda_k}$ according to Eq. (4.8), and define a critical point T as the point in the feature sequence, after which the features start to have significant overlapping among the classes. Since the features are ranked descendingly with λ_k , T can be chosen as the last feature whose λ_k is greater than or equal to M^2 , i.e.

$$T = \arg \max_k \{\lambda_k \geq M^2\}. \quad (4.13)$$

Thus $\sqrt{\lambda_T}$ is the eigenvalue that borders between separated classes and overlapping classes and is used as the reference. However, as the weights for features are measured relative to one another, we can take the ratio of $\sqrt{\lambda}$ s instead. In our approach, $\sqrt{\lambda_T}$ is chosen to be the reference to which all other weights are compared with. So features whose $\sqrt{\lambda}$ is greater than $\sqrt{\lambda_T}$ should be treated with equal weightage since they have equal (maximum) non-overlapping regions.

With the considerations above and normalizing the right side of Eq. (4.12) by $\sqrt{\lambda_T}$, the final relevance measure of a feature $\mathbf{w}_k$ is thus defined as

$$\Gamma_k = \begin{cases} 1 & \lambda_k \geq M^2, \\ \sqrt{\frac{\lambda_k}{\lambda_T}} & \lambda_k < M^2, \end{cases} \quad (4.14)$$

which is purely related to the Fisher discriminant value of that feature and a spread regulator M . This equation suggests that based on the non-overlapping region analysis, all the features before and at the critical point T , i.e. from $\mathbf{w}_1$ to $\mathbf{w}_T$, has the same relevance value (since all have maximum non-overlaps), while the features beyond T have different relevance values related to their λ_k , whose value will tend to vanish. Happily, by taking ratios, M only functions as a threshold regulator and does not directly affect the value of Γ_k .

From Section 4.2.2, we can also see that $\sqrt{\lambda_k}$ actually represents the normalized class separation on a feature w.r.t. the average class standard deviation (*std*), which gives a good description of the class distribution on the LDA features. In other words, it is the average projected distance between class center and global mean normalized over the average class *std*. From Eq. (4.14), it is interesting to find out that the relevance measure of features remains to be the same until the class distribution on a feature vector reaches a critical point at $\sqrt{\lambda_T}$, where the classes start to overlap. After that, the feature relevance will be proportional to normalized class separation. This simple relevance measure uses existing LDA parameters to provide a statistical indication of the non-overlapping regions even for multi-class problems with general class distribution. In the next section, we will use it to build up a weightage scheme to combine the LDA features for classification.

4.4 Relevance-Weighted Discriminant Analysis

The weightage scheme we proposed here is called Relevance-Weighted Discriminant Analysis (RWDA). The underlying rational is based on our basic definition of feature relevance measured by Γ . A preliminary version of it called wLDA is also given for comparison, which applies a weightage not based on our feature relevance measure, but based on the normalized class separation, $\sqrt{\lambda_k}$. In the experiments below, wLDA will help to verify the effectiveness of our proposed weightage scheme.

We should add here that our weightage scheme is not the same as that for the Mahalanobis distance that has been prevalent in statistical applications. It is to weight each feature or random variable with the inverse of its associated standard deviation. For classification problems, we argue that the element of inter-class distance separation (Δ_k) must necessarily be part of the weightage scheme, which our weightage scheme has done. Moreover, RWDA is based on feature relevance which is based on the areas of non-overlap.

4.4.1 The Framework of RWDA

Based on the discussion in Section 4.3, the framework of RWDA is proposed as follows:

1. *Compute LDA features $\{\mathbf{w}_k\}$ and the corresponding Fisher discriminant $\{\lambda_k\}$ from the training samples.*
2. *Set the value of the spread regulator M .*
3. *Determine the Critical Point T of the feature set, such that $T = \arg \max_k \{\lambda_k \geq M^2\}$ (cf. Eq. (4.13)).*
4. *Scale $\{\mathbf{w}_k\}$ to $\{\Gamma_k \mathbf{w}_k\}$ and use the latter for classification. Note that $\Gamma_k = \begin{cases} 1 & \lambda_k \geq M^2 \\ \sqrt{\frac{\lambda_k}{\lambda_T}} & \lambda_k < M^2 \end{cases}$ (cf. Eq. (4.14)).*

The setting of M can be based on statistical learning of the training samples or more precisely, data distribution analysis. Since M is only used loosely in finding the critical feature and its value does not need to be accurately set, usually, it is enough to use statistical results to choose M as a constant between 2 to 3. A larger M means less percentage of samples would be considered as outliers when evaluating the non-overlapping regions. For example, for multi-class problems, a rule-of-thumb is to set $M \approx 2.45$, such that $M^2 = 6$. If the class distribution is Gaussian, this is equivalent to the class spread of 99.5% of samples. In the experiments, we will stick to this setting for face recognition. For other application domains, training data can

reveal roughly where the critical feature is.

Alternatively, data distribution analysis can also be used to estimate a proper M value. This can be done by finding an average M for all classes that encloses a pre-defined percentage of the class samples. The penalty is the added computational effort in the training stage.

By examining the practical values of λ_k , it is seen that the relevance weights of RWDA drop from the constant 1 after the critical point, then slowly vanish to 0. As in Section 4.2.2, LDA features with smaller λ_k values usually have larger within-class scatter Ω_k than previous features, which may exceed their own between-class scatter Δ_k . If such features are included at equal weightage with good features, their large overlaps may adversely affect the classification effort. Thus there exists a point beyond which adding more features may degrade the classification performance and it explains why LDA exhibits peaking phenomenon. We show that this peaking phenomenon can be eliminated in RWDA by scaling the LDA features in accordance to their relative relevance. Our hypothesis is that as long as a feature is relevant, it can help to enhance classification accuracy commensurating with its degree of relevancy. This is verified later by our extensive investigations on face databases. In Section 4.5, we will provide a more rigorous justification to our relevance weightage.

4.4.2 wLDA - a Preliminary Version of RWDA

wLDA is a preliminary version of RWDA, which apply $\sqrt{\lambda_k}$ as weights to the LDA features. In fact, it is designed prior to RWDA, and the correct (latter) model of non-overlapping regions only flashes to mind after observing the problem with wLDA. In the experiments below, we report a finding discovered from early investigations based on wLDA, which shows that features yielding separated classes in their projections should be weighted equally. (The reason, which only became obvious to us later, is that they all have the same amount of non-overlap regions and thus should exhibit same level of relevance.) The consequence of not doing so was found to result in poorer performance of our relevance weightage scheme prior to the peaking phenomenon.

Formally, wLDA is proposed as follows:

For an LDA feature $\mathbf{w}_k$ with generalized eigenvalue λ_k , modify $\mathbf{w}_k$ such that

$$\mathbf{w}_k \longrightarrow \sqrt{\lambda_k} \mathbf{w}_k$$

This is a simple scheme which uses normalized class separation as weights. Compared to RWDA, the weights are in effect only different for the first few feature vectors before the critical point. In Section 4.6, we will evaluate the performance of wLDA, RWDA and LDA with extensive experiments. This in turn strengthen our conviction that the amount of non-overlapping region is the correct measure of feature relevance as against class separation information (margin) considered by others.

Figure 4.6: Regions of High Probability of Classification Error for Overlap Data.

4.5 Justification of Relevance Weightage

In this section, we provide a theoretical justification of the relevance weightage by analyzing the probability of classification error on weighted and unweighted features. We provide an understanding to why feature weightages based on their relevance are effective. The nearest neighbour classifier (NNC) will be used in our argument.

In order to explain the principle, a two class setting is assumed without loss of generality. First, consider the two overlapped classes Class 1 and Class 2 with two features, $\mathbf{w}_1$ and $\mathbf{w}_2$. Assume for simplicity their distributions are uniform and as shown in Fig. 4.6.

Theoretically for continuous distributions, the total probability of error expected is represented by region A. Region A is the lower bound and no classifier can do better than this. If only feature $\mathbf{w}_1$ is used, we get at least the error shown in the crossed region of the projected data, which is $A + B + C$. For feature $\mathbf{w}_2$, the minimum error cannot be less than $A + D + E$ irrespective of the classifier. Obviously, feature $\mathbf{w}_1$ gives better performance than $\mathbf{w}_2$ since the latter has much greater overlap. If we choose only $\mathbf{w}_1$, we incur greater errors than the lower bound, A. Consequently, it stands to reason that we should not discard $\mathbf{w}_2$. But from reports by others, there seem to be a peak of selected features beyond which the recognition performances drop. This drop has been alluded to overfitting [31] [59]. The observed overfitting happens because along $\mathbf{w}_2$ the overlap is great and thus including feature $\mathbf{w}_2$ degrades the classifier. We point out here that by proper weighting of the features, we could improve the performance beyond the observed peak. We will show that as long as a feature is “relevant” it can be useful and not cause any overfitting problem. Overfitting is due to a classifier being over-expressive so that they learn the noise as well. But noise is due to overlapping of class distribution. Our premise for relevance weighting is not on the overlap but

Figure 4.7: Loci of Equal Choice for Unweighted and Weighted Features.

on the non-overlapping regions. So adding another feature which has some non-overlapping regions should be useful. It should not lead to overfitting if properly weighed by its relevance.

By our previous definition, feature $\mathbf{w}_2$ should be relevant since there exists along it non-overlapping regions (Fig. 4.6). It is clear that $\mathbf{w}_2$ contains discriminative information. Thus the key question is how we can tap this useful information without compromising the classification accuracy. Here, it will be shown that the answer is linked to Fisher discriminant values, λ_i , which is related to the relevance of $\mathbf{w}_i$.

Consider now a set of discrete samples distributed i.i.d. w.r.t. the uniform distributions of Fig. 4.6 and let Δx be the average spacing, or sample density, between data. For example, if on a line there are N samples spread over an interval of L , then the average spacing between two neighbouring samples is L/N . The overlap of the data along feature $\mathbf{w}_1$ is less than along $\mathbf{w}_2$, which as shown previously, is dependent on the Fisher discriminant value $\sqrt{\lambda_i}$ along the associated feature direction. Larger $\sqrt{\lambda_i}$ implies smaller overlap or larger non-overlaps. Take the ratio $\sqrt{\lambda_1}/\sqrt{\lambda_2} = k$, that is the overlap in the data is in the order of k times more in the $\mathbf{w}_2$ than in the $\mathbf{w}_1$ direction.

As explained, it is preferred to place more confidence in data along $\mathbf{w}_1$ than along $\mathbf{w}_2$ since the expected errors are greater along the $\mathbf{w}_2$ direction. Since it has some relevance, $\mathbf{w}_2$ is included to achieve better performance by our hypothesis. Consider the situation in Fig. 4.7 where an unknown sample data $\mathbf{u}$ is presented. For the unweighted ($k = 1$) case, the nearest neighbour rule will pick $\mathbf{p}$ and $\mathbf{q}_1$ equally since their Euclidean distances to $\mathbf{u}$ are the same. For a weight of k greater than unity, the locus of equal choice is now an ellipse as shown. The greater k is, the more elongated is the ellipse in the direction of $\mathbf{w}_1$. The limit of $k = \infty$ ignores $\mathbf{w}_2$.

We are focusing on the problem of peaking which occurs when λ_i is much smaller

Figure 4.8: NNC in the Narrow Region B with and without Relevance Weightings.

than λ_{max} . The amount of overlap in the former direction will be great and there is expected to be little non-overlapping regions. Can these relevant regions be salvaged and put to some good use? Our hypothesis says that it can. When $\sqrt{\lambda_{max}}/\sqrt{\lambda_i} = k$ is very large², say in the range of 10 to 100, the width of the non-overlapping regions in $\mathbf{w}_2$ such as B and C of Fig. 4.6 will be small in comparison to the center-to-center distance. Assume k is chosen appropriately. Then for regions B and C, where the width of the non-overlapping region(s) along $\mathbf{w}_2$ is so narrow, the benefit of relevance weightages for the inclusion of feature $\mathbf{w}_2$ comes to light.

Consider the situation in Fig. 4.8 that examines the NNC classification errors in the narrow non-overlapping region B. Note first that the expected error of the NNC is the Bayes error that is determined by the probability densities of the two classes in that local region. It is to be shown that there will be with high probability less classification errors with relevance weighting than without. Let $\mathbf{u}$ be a point in B that is δ distance from the boundary of A and B. If Δx is the average sample spacing, then it will be very unlikely the ellipse not to contain any samples inside it. The larger k is, the less likely. Consequently if the circle is fully in B, NNC makes no error. Error will occur only if $\delta < \Delta x$. The probability of error will increase as $\mathbf{u}$ gets nearer to the boundary.

When $\mathbf{u}$ is very close to the boundary ($\delta \approx 0$) as shown in Fig. 4.9, the error is half of what it will make if only $\mathbf{w}_1$ was used. This is because in the lower half of the ellipse, all the points are in B and so will be correctly classified by NNC. In the upper half of the ellipse, the probability of error will depend on the probability densities of the two classes in A. For uniform distributions, the error is 50%. Hence for the situation in Fig. 4.9, the error is 25%.

In all, if only $\mathbf{w}_1$ was used, the probability of error from A is 50% by NNC. Whereas if both features were used, the error in A will be close to zero, the contri-

²Assume λ_{max} corresponds to $\mathbf{w}_1$ and λ_i corresponds to $\mathbf{w}_2$

Figure 4.9: Sample u in B is Just at the A-B Boundary.

bution of errors occurs when $\delta < \Delta x$ which will be small. Then, when both features are used, in comparing weighted and non-weighted features, observe that the errors start to occur when $\delta < k\Delta x$. With k large, the errors by the unweighted features will be much greater than with relevantly weighted features. A similar argument applies to the boundary of A and C.

What remains to be shown is to examine the classification of points close to the boundaries between A and D and between A and E. Recall that the assumption is that D and E are much larger than A in area (as $\lambda_1 \gg \lambda_2$). We show that the classification will not be degraded.

In Fig. 4.10, let $\mathbf{u}$ in D be a distance of δ from the boundary between A and D. If $\delta > k\Delta x$, then NNC will correctly classify $\mathbf{u}$. In the case when $\delta < k\Delta x$, the circle and ellipse will transgress into A. The probability of error will be in the ratio of the area of that part of the circle (unweighted) on the left of the boundary to that on the right. But this is the same ratio for the ellipse (relevance weighted). Hence the probability of error for the weighted case and for the unweighted case is the same. Furthermore, the error probability if only $\mathbf{w}_1$ was used is the same. This becomes apparent if we collapse the ellipse to a line (along $\mathbf{w}_1$). Thus for the A-D and also the A-E boundaries, the error probabilities remain intact with weighted features.

Finally, from all the possible cases, we conclude that weighting features properly can improve the classification performance by recovering information from non-overlapping regions existing in the added feature. Proper weightage is crucial and shall come from the feature relevance that is determined by the amount of overlap. We cannot give an absolute weight to each of the features but instead its relevance should be compared with other features' relevance. Thus, the motivation is to set k as a ratio between a reference feature with the feature under consideration. This leads us back to the relevance weighting scheme as proposed previously.

Figure 4.10: Sample u at the A-D Boundary

4.6 Experiments

In this section, the performance of RWDA is evaluated using the UMIST [33], the ORL [32] and the YALE [34] face databases. The UMIST database consists of 575 gray-scale images of 20 individuals(classes). The ORL database consists of 40 persons (classes) each with 10 images, while the YALE database consists of 15 persons each with 11 images. Thus, the maximum number of LDA features is 19 for UMIST, 39 for ORL, and 14 for YALE. The classification is done by Nearest Neighbor using either L_1 or L_2 norms as the distance (similarity) measure. To make the test results statistically meaningful, all the testings reported below are averaged over 100 random runs, where for each run, the training samples are randomly selected from the database and the rest of the images are used for classification testing.

4.6.1 LDA versus wLDA

First, we compare the classification performance of wLDA and LDA, the results of PCA are also presented here for reference. In our context here, LDA will refer to the conventional LDA without any weighting.

Fig. 4.11 to Fig. 4.13 show the classification results on UMIST, ORL and YALE face databases for a range of feature sizes. For UMIST and YALE, the training pool contains 8 random samples per person, while for ORL, 5 random samples per person are trained. First, we can see that the LDA based methods generally perform much better than PCA. Although the feature sizes shown in Fig. 4.11 to Fig. 4.13 are far less than the full range of PCA features, we have nevertheless tested up to the maximal number of PCA features and found no further improvement on the recognition rates. As also being confirmed with the results shown in the next three sections, PCA features are much less effective for classification tasks compared to

Figure 4.11: Classification Performance of wLDA v.s. LDA and PCA on UMIST Database

Figure 4.12: Classification Performance of wLDA v.s. LDA and PCA on ORL Database

the LDA-based methods.

Fig. 4.11 to Fig. 4.13 also show that there is severe peaking phenomenon demonstrated by the conventional unweighted LDA. After the optimal feature size has been reached, the performance of LDA deteriorates progressively thereafter. In practical applications, this is not preferred as it will raise one more parameter to adjust, which is the optimal feature size.

Figure 4.13: Classification Performance of wLDA v.s. LDA and PCA on YALE Database

The results of wLDA showed no sign of peaking phenomenon in Fig. 4.11 and Fig. 4.12, and a much less severe peaking in Fig. 4.13 compared to that of LDA. However, for the ORL database, the best performance of wLDA can never reach that of LDA. Something is not right with the weightages. It is noted that at the start, the first few features when weighted already showed poorer performance than LDA. The cause is due to the fact that for these most relevant features, the projected classes do not overlap. So their non-overlapping regions are equal and hence should be weighted equally. This finding led us to the ultimate RWDA scheme.

Another observation from Fig. 4.11 and Fig. 4.12 is that LDA performs better with L_2 norm than with L_1 norm, while wLDA works better with L_1 norm. Although in Fig. 4.13 this is not consistently maintained, the difference between the use of the two norms is very small. For the subsequent tests, we save work by only comparing LDA with L_2 norm and wLDA with L_1 norm to the performance of RWDA.

4.6.2 RWDA Performance on the UMIST Face Database

The classification performance of RWDA on the UMIST database are evaluated at two different training sample sizes: 8 samples/class (Test A) and 5 samples/class (Test B). The results are shown in Fig. 4.14 and 4.15 at various feature sizes, and are compared to other methods such as LDA, wLDA, and PCA. Both L_2 and L_1 norms are used with RWDA as distance measures for classification. For RWDA, we set $M \approx 2.45$, or equivalently $M^2 = 6$, following the discussion in Section 4.4.1. Fig. 4.14(a) and 4.15(a) show the average classification accuracy over 100 random runs, while Fig. 4.14(b) and 4.15(b) show the corresponding performance standard deviation obtained from the statistics.

From Fig. 4.14 and 4.15, we can see that RWDA- L_2 statistically performs the best

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.14: Classification Results of RWDA Compared to Other Methods - UMIST Test A (8 samples/class trained)

among the five test combinations. It is not only the most accurate, but is also the one among the lowest performance standard deviations over 100 randomly selected training/testing combinations. The accuracy of RWDA- L_2 appears higher than that of LDA even at the latter's optimal feature size. Moreover, the performance

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.15: Classification Results of RWDA Compared to Other Methods - UMIST Test B (5 samples/class trained)

of RWDA- L_2 increases monotonically as more features are added on, completely eliminating the peaking phenomenon of LDA.

We also notice that RWDA works better with L_2 norm, in contrast to the wLDA case, where L_1 norm works better. The performance of RWDA- L_2 is 0.35% higher

than that of RWDA- L_1 in Test A, and is 0.8% higher in Test B. As discussed in Section 4.2, the relevance measure we used is based on the geometric property of the LDA features. So it is not a surprise to find that our method better suits the Euclidean distance (i.e. the L_2 norm). Comparing the size of training pool in Tests A and B, we also see that the improvement of the RWDA over LDA is more significant when the samples are more sparse. This is an indication that RWDA has stronger generalization ability.

4.6.3 RWDA Performance on the ORL Face Database

The tests on ORL database are conducted and presented in a similar way to those of UMIST. This time, three tests are conducted at varied training pool size:

- Test A: 5 samples/class,
- Test B: 4 samples/class, and
- Test C: 3 samples/class.

As shown in Fig. 4.16, 4.17 and 4.18, the RWDA- L_2 combination still performs the best and is among the most stable methods. The minor differences between the results over ORL and UMIST database are

1. the peaking phenomenon of LDA is less severe on ORL, and
2. the performance improvement of RWDA- L_2 over LDA is slightly less on ORL (cf. Table 4.1).

A possible explanation for such behavior is that the UMIST database contains larger pose variations, and its samples are more sparsely distributed over the feature space.

4.6.4 RWDA Performance on the YALE Face Database

For YALE Database, we tested the performance of RWDA on two kinds of training sample sizes as follows:

- Test A: 8 samples/class,
- Test B: 5 samples/class, and

The rest of the test set-up is the same as those with the UMIST and the ORL databases.

As shown in Fig. 4.19 and 4.20, the performance comparison between RWDA and LDA is very similar to those with UMIST and ORL databases. The advantages of RWDA is clearly demonstrated with the experimental data.

In Table 4.1 and 4.2 we study the classification accuracy and the performance deviation at the peak points of RWDA- L_2 and LDA- L_2 for the various tests. We found that RWDA consistently performs better than LDA especially for UMIST and

Table 4.1: Best Performance of RWDA- L_2 and LDA- L_2

Accuracy (%)	UMIST		ORL			YALE	
	Test A	Test B	Test A	Test B	Test C	Test A	Test B
RWDA- L_2	97.6	93.3	96.3	93.6	89.1	94.7	92.1
LDA- L_2	96.9	92.5	96.2	93.3	88.6	93.7	91
Difference*	0.7	0.8	0.1	0.3	0.5	1.0	1.1

* $\%_{RWDA} - \%_{LDA}$ Table 4.2: Performance Deviation at the Peak Points of RWDA- L_2 and LDA- L_2

Performance Deviation (%)	UMIST		ORL			YALE	
	Test A	Test B	Test A	Test B	Test C	Test A	Test B
RWDA- L_2	1.28	2.37	1.50	1.77	2.49	3.01	2.92
LDA- L_2	1.28	2.38	1.49	1.97	2.47	3.55	2.66

Table 4.3: t Test to compare the peak performance of RWDA- L_2 and LDA- L_2

t Test	UMIST		ORL			YALE	
	Test A	Test B	Test A	Test B	Test C	Test A	Test B
t	3.85	2.37	0.47	1.13	1.42	2.14	2.77
$\alpha < (\%)$	0.01	1.0	32.0	13.0	8.0	1.7	0.3
ConfidenceLevel(%)	99.99	99	68	87	92	98.3	99.7

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.16: Classification Results of RWDA Compared to Other Methods - ORL Test A (5 samples/class trained)

YALE databases. In terms of stability, the two methods are similarly good at their peak points. But the peak point of RWDA- L_2 occurs when all the relevant features are used, while the actual peak point for LDA has to be searched for in practice. Since RWDA requires few additional computation for the weightage, RWDA guarantees a better performance than LDA without much further effort in searching the

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.17: Classification Results of RWDA Compared to Other Methods - ORL Test B (4 samples/class trained)

optimal feature size.

To verify the statistical significance of the experimental results above, we conducted t Test to compare the performance differences between RWDA and LDA for all the test cases above. The degree of freedom is 198 for all the cases as each

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.18: Classification Results of RWDA Compared to Other Methods - ORL Test C (3 samples/class trained)

method has been evaluated with 100 random tests. Table 4.3 summarizes the t Test results, and we can see that the confidence level for RWDA to perform better than LDA is more than 98% for UMIST and YALE databases. For ORL database, the confidence level is relatively low, but it consistently improves to 92% as less samples

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.19: Classification Results of RWDA Compared to Other Methods - YALE Test A (8 samples/class trained)

are involved for training.

(a) Accuracy Averaged over 100 Runs

(b) Corresponding Performance Deviation

Figure 4.20: Classification Results of RWDA Compared to Other Methods - YALE Test B (5 samples/class trained)

4.6.5 Discussions

In this section, the important findings from the experiments are discussed.

First, it has been verified experimentally that the proposed relevant weights in RWDA do avoid the peaking phenomenon and continue the monotonic rise in classifi-

Table 4.4: The position of critical point T for RWDA

	UMIST		ORL			YALE	
	Test A	Test B	Test A	Test B	Test C	Test A	Test B
T	5	5	10	8	8	7	5

cation performance as more relevant features are added. We find in Fig. 4.14 to 4.18, the inclusion of more LDA features at proper relevance weightage can consistently enhance the recognition performance until a saturation point is reached. Thus, with RWDA, we can directly use the whole set of LDA features without worries of having to find the optimal feature size. Similar to LDA, RWDA works better with L_2 norm for distance measure instead of L_1 norm. We have also verified experimentally why the early features before the critical point should be equally weighted among each other.

Also, RWDA is shown to have better generalization ability than LDA especially for more sparsely distributed samples, i.e. smaller training pool or samples with bigger within-class variations. From results in Section 4.6.2 to 4.6.4, it is observed that RWDA consistently performs better than LDA's peak recognition rates, and the features after LDA's peak points can still add up to the generalization performance. This shows that the relevance weightage can better utilize the feature information to help classification.

We now comment on the position of critical point T of RWDA, which is resultant from the M setting. Note that the weights of the first T features for RWDA are all scaled to 1, which is actually equivalent to LDA and the differences only come for the rest of the features. The critical point T in RWDA can be found near to the LDA peak point, where the performance curves of RWDA- L_2 and LDA- L_2 start to deviate. From the experiments in Section 4.6.2 to 4.6.4, the values of T are found and summarized in Table 4.4. We can see that the value of T tends to decrease with more sparsely distributed samples. This could possibly imply larger overlapping class regions for LDA features obtained from such samples.

A final observation is about the PCA's performance as compared to the LDA-based approaches. Our experiments consistently show that the LDA-based methods outperform PCA by at least a few percentage. However, at certain feature size, PCA may still overtake LDA such as in the case of Fig. 4.18. Therefore, it is important to set proper feature size for LDA. The RWDA method proposed here well serve this purpose.

4.7 Conclusion

In this chapter, we propose a simple LDA variation, RWDA, which applies a novel relevance weightage to LDA features to combat the peaking phenomenon that is persistent with LDA and its variants. The relevance measure we propose is derived from an non-overlapping region model, which can be generalized to all projection-based features and cater for arbitrary data distributions. For LDA, this relevance weightage can be readily computed from Fisher discriminant values with a constant spread regulator M . The effectiveness of such relevance weightage is also justified by analyzing the probability of classification error on weighted and unweighted features. Our intensive experiments with the UMIST, the ORL and the YALE Face Databases show that RWDA can completely eliminate the peaking phenomenon of LDA, which alleviates the requirements of selecting optimal feature size. Moreover, RWDA demonstrates better generalization ability than the direct use of the same set of LDA features, which can actually lead to better classification performance.

The investigations in this chapter indicates that as long as a feature is relevant, it can be used in proper weightage to enhance the generalization ability of the classifier. This is a novel finding as compared to the traditional view of overfitting that features after the optimal feature size will only deteriorate the classification results. The RWDA we proposed here extend this new finding and is proved to work well for difficult multi-classification problems, under high input space dimension, and with sparsely distributed samples.

The formulation of RWDA allows it to be readily extended to the non-linear version of LDA using the kernel tricks. In the future, we may look into how our relevance measure works with Kernel Discriminant Analysis (KDA). Also, it is worthwhile to test RWDA on other benchmark multi-class data sets to verify its efficiency on applications other than face recognition. As for database with sub-clusters of classes, RWDA may not give good enhancement over LDA. For such situation, we recommend to pre-cluster the samples before applying RWDA. It is also interesting to see how RWDA works for such kind of database.

Chapter 5

Investigations on 2D Subspace Methods

5.1 Introduction and Motivation

Chapter 3 and Chapter 4 focus on subspace methods with conventional vector-based (1D) data representation, where all the samples are treated as points in a high dimensional vector space. For image-based applications such as pattern recognition and image retrieval, the 2D image data has to be reshaped into a long vector containing all its pixel values, which often results in very high dimensional data space. The majority of the existing pattern recognition and analysis algorithms are based on this 1D representation and many have achieved great success. Unfortunately, such a scheme often makes it computationally prohibitive to solve for high dimensional problem with very large datasets, which often incur large size matrices during the training stage.

Under the 1D data representation, the ever possible concise way to represent the sample statistics for subspace methods is by $N \times N$ matrices, where N is the total number of the samples. For linear subspace methods, the original matrix size involved in feature extraction is often $d \times d$, where d is the data space dimension, but we can reduce the eigen-decomposition problem by working on a $N \times N$ matrix $A^T A$ instead of a $d \times d$ matrix AA^T if $d \gg N$ (c.f. Section 3.3.1). For nonlinear subspace methods based on the kernel trick, the eigen-decomposition is already based on $N \times N$ matrices as we only evaluate kernel matrix of the sample size, although the most computation intensive part is to evaluate this $N \times N$ kernel matrix for all d -dimensional data. Thus, either case, the problem may exceed the computation and/or memory limit on a PC if both d and N are very large. In this thesis, we address this problem of high dimensionality and large data set. Our approach is simply not to vectorise the image. This means the proposed method will be kept within the confines of the image dimension. Thus, for all practical image classification problems, the above-mentioned problem is overcome by our approach.

In this chapter, we investigate on how to directly apply the matrix-based (2D) representation of image data for subspace methods to overcome this problem. This

2D Subspace Approach has a few advantages compared to the conventional subspace methods using 1D representation. Firstly, it is possible to restrict the matrix involved for feature extraction to the image size irrespective of the data size, hence elevating the computation load when both data dimension and sample size are large. Secondly, without vectorizing the pixels, 2D spatial information of the image data can be better preserved and possess 2D structures that we will show can be usefully extracted. Thirdly, the data statistics can now be evaluated in a lower dimensional space, which does not require as much samples as those required by the conventional 1D representation for proper estimation of the data distribution. Our approach thus avoids the Curse of Dimensionality and also reduces the probability of encountering the Small Sample Size Problem. In this chapter, we explore a few ways to apply 2D data representation to subspace methods, and compare their performances to conventional vector-based subspace methods using the ORL face database.

Our work on 2D Subspace Methods were first reported in Year 2000 and Year 2002, and are among the earliest to pioneer the study of 2D subspace approaches of PCA and LDA for classification problems, especially in face recognition [149] [150]. Prior to us, Hong [56] and Cheng et. al. [17] proposed some 2D Subspace Methods based on Singular Value Decomposition (SVD) in early 1990s without concerning class-discriminant information. Hong used SVD to derive SV feature from individual face images for classification, but only tested them on a small nine-person database [56]. This method does not involve any statistical analysis across sample images. Cheng et. al. derived SVD basis from simple statistics that is the average face of the available samples, and projected faces onto this common basis [17]. They claimed 100% accuracy in classifying 64 images of 8 persons, but expected worse performance on a larger database. In 2000, Wang et. al. claimed that Hong's SV features can only achieve 80.9% recognition rates on ORL Face Database using simple distance matching [136]. They applied RBF neural network classifier to boost up the performance to 92%, and attributes this improvement to RBFs learning ability with negative samples. In [149] and [150], we presented some of our preliminary works on 2D Subspace framework, called the Spectral-Face Analysis, which computed 2D subspace bases using statistics such as 2D-covariance matrix, and project images to them for classification. We also applied the linear discriminant analysis to one of such methods, called the Covariance Face Method, and observed significant improvement for classification.

Since 2004, more recent works of 2D Subspace Methods have been reported. In [140], linear 2DPCA was proposed which constructs a type of 2D-covariance matrix based on 2D image data, and project each image A to a feature matrix where each column of that matrix can be obtained from a feature vector $\mathbf{x}$ as $\mathbf{y} = \mathbf{A}\mathbf{x}$ ($\mathbf{x}$ is the eigenvector of the 2D-covariance matrix). They report better performance than PCA in face recognition when the number of samples is small. In [73], Kong et. al. pointed out that 2DPCA is essentially the PCA performed on the rows of all the images, and is a unilateral projection-based scheme, where only right multiplication is taken. They extend 2DPCA to bilateral-projection-based 2DPCA (B2DPCA), which can achieve more concise feature matrix than 2DPCA, and can

better preserve the correlation information in both rows and columns of the images. However, the method finds no close-form solution, and has to be solved through an iterative algorithm. In their work, a kernel-based 2DPCA scheme (K2DPCA) is also introduced to help explore higher-order statistics among the row/column vectors of an image. Later in [72], Kong et. al. adopted a similar approach to extend their works on PCA-based methods to Generalized 2D Fisher Discriminant Analysis, and similarly proposed schemes like bilateral-projection-based 2DFDA (B2DFDA) and kernel-based 2DFDA (K2DFDA).

Recently, More developments on 2D Subspace Methods have been reported for other types of subspace methods [145] [135] [91]. In [145], 2D Bayesian subspace analysis (2D-BSA) was proposed for face recognition to achieve lower dimensional representation. Since BSA uses probabilistic measure of similarity, this elevates the requirements of getting more samples for appropriate statistical estimation. In [135], 2D image representation was applied to the maximum margin criterion (MMC) of [82] to completely circumvents the small sample size problem and improve computational efficiency. They claim this achieves better performance in terms of both recognition accuracy and training time as being demonstrated on real data set. In [91], clustering-based discriminant analysis (CDA) is also extended to use the 2D representation. It seems when we increasingly deal with problems with high dimensional data and large datasets, the 2D subspace approach has become one of the trendy solutions as it intrinsically enjoys computational efficiency and is able to achieve equivalent or even better performance than its 1D counterparts.

In this chapter, we summarize our early explorations on 2D subspace methods and show some interesting findings. Our work adopts unilateral projection-based scheme, which predates some recent successful methods in [73] and [72] that adopt bilateral projection-based scheme. Moreover, our work completes the whole 2D subspace paradigm with surveys to earlier works which are often missed in the latest research reports. We claim that our idea is the precursor to the later works by others in 2D subspace pattern recognition research.

This chapter is organized as follows: First, we define the objectives that led us to the various development and investigations, and indicate the experiment structure linked to these objectives. The proposed 2D subspace techniques and the experiment results are then illustrated according to this structure in Section 5.3. Finally, conclusion and future work are given in the last section.

5.2 Objectives and Experiment Structure

We reiterate that the main objective of our 2D approach is to solve the high dimension-very large data set problem. In this chapter, we conduct our investigations based on the following sub-objectives:

1. To show feasibility of the 2D Subspace Method, and propose some possible implementations (Section 5.3.1 and 5.3.2);

2. To try out various ideas to enhance the performance of the 2D Subspace Method (Section 5.3.3 to 5.3.5);
3. To show the existence of some relationships between the 2D image basis and the face image, and the possible usage of 2D image bases for compact face representation and recognition (Section 5.3.6).

The section below illustrates the experiment structure linked to these objectives for easy reference.

5.2.1 Experiment Structure

For Objective 1, the Singular Value Decomposition (SVD) is used to extract the 2D subspace basis from 2D image (matrix-form) representations and a framework called the Spectral-Face Analysis is proposed. Under this framework, there are two variations called the Covariance Face Method (Section 5.3.1) and the Error Face Method (Section 5.3.2). The Covariance Face Method takes an analogy to PCA but computes the covariance based on 2D matrix data. The Error Face Method, on the other hand, computes 2D basis from a pseudo-random matrix and generates features by some sort of 2D random projections. We investigate into these two methods to find out what is the important information for the 2D basis derivation to achieve good classification performance.

The empirical classification results of the proposed Spectral-Face Analysis are found to be worse than those of PCA. Thus, for Objective 2, we look at various ways to enhance the performances of the proposed methods. The first approach extends feature space dimension by using larger images (Section 5.3.3). The second approach applies class-discriminant analysis to the Covariance Face Method and results in the Discriminant Covariance Face Method (Section 5.3.4). We compare the effect of this enhancement to those with the vector-based PCA. Another way to enhance the 2D subspace methods is to slightly enhance their feature space dimension, so that the discriminant information will not be discarded too much due to highly compact representation. We design an image stacking method to allow incremental feature space dimension for the Covariance Face Method, and use it to verify our hypothesis in classification performances (Section 5.3.5).

To complete the investigations, we also look into the properties of the image SVD bases in Objective 3 (Section 5.3.6). We find that some of the SVD bases demonstrate good invariance properties corresponding to the facial features.

In next section, we present the details of the above mentioned investigations.

5.3 Derivations and Experiments

5.3.1 The Covariance Face Method

The Covariance Face Method is one type of the Spectral-Face Analysis, which uses 2D data representation for face images and derive 2D projection bases using SVD.

During classification, each face image is then represented by a vector of projection values on these 2D bases.

The Formulation

In the Covariance Face Method, we take an analogy of the PCA method, but carry out the analysis with 2D data representation. Consider N face samples, with each face, M_i , represented by a $m \times n$ matrix of image pixel values, The 2D-Covariance Matrix of all the sample images can be obtained as

$$\begin{aligned} C &= E_i\{(M_i - \bar{M})(M_i - \bar{M})^T\} \\ &= \frac{1}{N} \sum_{i=1}^N [(M_i - \bar{M})(M_i - \bar{M})^T], \end{aligned} \quad (5.1)$$

where $\bar{M} = E_i\{M_i\}$ is the average face. We stress here that M_i is a matrix and not a vector, that has been prevalently used in PCA and LDA. It can be verified that C is actually the covariance matrix of all columns of the sample images, and the size of C remains at $m \times m$ regardless of how many samples are used for training. This explains why our method has fix dimensions irrespective of the size of the database and is also amenable to any practical size image implemented.

The idea now is to find the Singular Value Decomposition (SVD), or the spectral decomposition of C to construct the 2D projection basis. This leads to

$$C = \sum_{k=1}^r s_k \mathbf{u}_k \mathbf{u}_k^T, \quad (r = \text{rank}(C)) \quad (5.2)$$

where $\mathbf{u}_k$ is the unitary left/right singular vector (SV), and s_k is the corresponding singular value [42]. It can be easily verified that, as C is symmetrical, its left and right SVs are both equal to its eigenvector, and the corresponding singular value also equates to the eigenvalue.

Without loss of generality, we can select the first K eigenvectors corresponding to the K largest singular values to construct the 2D projection basis for feature extraction. In other words, each ‘‘covariance face’’ $(M - \bar{M})(M - \bar{M})^T$ can be approximated by

$$(M - \bar{M})(M - \bar{M})^T \approx \sum_{k=1}^K a_k s_k \mathbf{u}_k \mathbf{u}_k^T, \quad (5.3)$$

where $\mathbf{u}_k \mathbf{u}_k^T$ can be viewed as the 2D projection basis. The unitary and orthogonal properties of the singular vectors suggest that $\mathbf{u}_k^T \mathbf{u}_k = 1$ and $\mathbf{u}_j^T \mathbf{u}_k = 0 (\forall j \neq k)$. Thus, in its covariance face form, each face image can be linearly projected to this orthogonal basis as a K -dimensional feature vector $\mathbf{a} = \{a_k\}$, where a_k is computed as

$$a_k = s_k^{-1} \mathbf{u}_k^T (M - \bar{M})(M - \bar{M})^T \mathbf{u}_k. \quad (5.4)$$

The extracted feature vector $\mathbf{a}$ can then be used for classification.

Figure 5.1: Example of First 18 2D Projection Bases for the Covariance Face Method.

Figure 5.2: Reconstruction with the 2D Projection Bases for the Covariance Face Method (a) original image, M (64×64 pixels); (b) covariance face, $(M - \bar{M})(M - \bar{M})^T$, (c)(d)(e) reconstructed covariance face using the first 5, 30, 64 features respectively.

Fig. 5.1 shows some sample images of the 2D projection bases for the Covariance Face Method. It can be seen that the first few bases carry lower spatial frequency information which capture some global structures of face images, while the subsequent ones are more on higher spatial frequency components that capture image details. In Fig. 5.2, the reconstruction of a covariance face were demonstrates at different feature size (c.f. Eq. (5.3)). It can be seen that feature sizes beyond 30 do not add visible details to the reconstructed pattern. Moreover, even at the full rank (feature size = 64), the reconstruction cannot well recover the original pattern. Although small reconstruction error is not a necessary property for features with good classification performance, it does indicate loss of information during the feature extraction process. In the next section, we will see how this affects the classification performance of the Covariance Face Method.

Figure 5.3: Performance of PCA at Different Feature Size.

Experiments and Discussion

The proposed Covariance Face Method is first compared to PCA using the ORL Face Database from AT&T laboratories [32]. The database contains 40 persons each with 10 different images. For all the performance evaluation in this chapter, we use 5 images per person for training and the rest of the images for testing. The results presented are all averaged over at least 20 runs of randomly partitioned training and testing sets.

In this section, we used 64×64 images for evaluation. The classifiers used include Nearest Neighbour (NN), Nearest Center (NC), and Nearest Feature Line (NFL) [83], which are all based on the euclidian distance between the projected samples for similarity measure. The NFL classifier proposed by Li et. al. constructs “feature lines” by connecting any two samples of the same class and measures face similarity based on distances to these feature lines. It simulates the effects of linearly interpolating between the face samples to cover more virtual appearance variations. In this chapter, we include it as part of comparison to evaluate the importance of more training samples for the Spectral-Face kind of approaches.

Fig. 5.3 and 5.4 shows the classification performance and the performance deviation of PCA over 20 random runs. Similarly, the performances of the Covariance Face Method is also shown in Fig. 5.5 and 5.6. Comparing the effect of different classifier on PCA and the Covariance Face features, we can see that in general, NC gives the worst estimation since the distribution of face samples can hardly be represented by their class centers; NN serves as a benchmark that gives the baseline performance of the extracted features; while NFL gives the best performance as

Figure 5.4: Performance Deviation of PCA at Different Feature Size.

Figure 5.5: Performance of Covariance Face Method at Different Feature Size.

indicated by highest average accuracy and lowest performance deviation. It shall be noticed that the improvement of NFL over NN is much more significant with the Covariance Face features (about 5%) than with the PCA features (about 1%). This suggests that the Covariance Face Method may not be very good at retaining class-relevant information as it works in low feature space dimension, and would

Figure 5.6: Performance Deviation of Covariance Face Method at Different Feature Size.

prefer classifiers with good sample generalization ability.

Comparing the performance of PCA and the Covariance Face Method, we can see that both of them demonstrate a saturation pattern with increased feature size. Using NFL classifier, PCA starts to reach the highest recognition rates at a feature size of 40, and eventually achieves 95% accuracy. Whereas, the Covariance Face Method starts to saturate at a feature size of 20, which gives about 84% accuracy, and the final accuracy can reach 88% at the maximum feature size of 64. The similar trend of performances of the two methods might be due to the similar covariance evaluation they use to generate the features, no matter 1D or 2D representation. However, PCA consistently demonstrates much better performance than the Covariance Face Method, which may suggest that features derived from lower dimensional space tend to have more cluttered sample distribution. This is one of the major drawback in using a lower dimensional feature space.

5.3.2 The Error Face Method

This so-called method is really found by accident. In the early work, the program was incorrectly written to compute the covariance matrix. Instead, what was really computed was the zero matrix. Due to the finite precision of the computers, the eigenvalues computed were not zero but were extremely small. We would have ignored this except that these values are seemingly random and yield associated eigenvectors which turn out to produce good classification results on the face database.

Curiosity pricked, we tried to find the reason. It seems that the zero matrix we obtained actually consisted of small random values, and displayed a rudimentary face shape (see Fig. 5.7). We call this zero matrix “error face”, and hence the name of this method. A reconstruction of its eigenvectors gave rise to a series of 2D projection bases which contained random-like high spectral information. This reminded us of the conventional Random Projection (RP) methods, in which the original high-dimensional data is projected onto a low-dimensional subspace using only some unitary random bases [64].

Recently, RP has demonstrated good performances in a number of applications, yielding results comparable to conventional dimensionality reduction techniques, such as PCA, while having much lower computational requirements [99]. RP is motivated by the Johnson-Lindenstrauss lemma [64] that states that a set of N points in a high-dimensional Euclidean space can be mapped down onto a low-dimensional subspace such that the distances between the points are approximately preserved. A proof of this theorem as well as tight bounds on dimension reduction can be found in [26] and [1]. Goel et. al. [99] showed that “RP might be an attractive alternative for dimensionality reduction in certain face recognition applications.”. By our Error Face Method, it seems that we have accidentally implemented a 2D random projection method. It gives much better classification performance than the Covariance Face Method, despite the fact that it only takes in some high-spectral information on pixel value distribution. This might suggest that the projection basis derivation can actually be less data-dependent.

The Formulation

In the Error Face Method, the basis derivation is based on

$$C = E_i\{M_i - \bar{M}\} \quad (5.5)$$

which should be mathematically zero as $E_i\{M_i\} = \bar{M}$. However, digital computing machines will always obtain a C with very small, trivial values due to the round-off error, which forms a pseudo-random matrix. A sample image of C is shown in Fig. 5.7. The pixel values of C range from 7×10^{-14} for black to 5×10^{-14} for white, and a rudimentary face appears in the central region of C as indicated by the larger contrasts of the random-like pixels. Hence, we have “accidentally” found a way to generate random face images by applying a finite machine to solve for Eq. 5.5.

Following the Spectral-Face Framework, the 2D projection bases and the face projections based on this “error face” C can be computed in a similar way as those in the Covariance Face Method. First, the spectral decomposition (SVD) of C is obtained as

$$C = \sum_{k=1}^r s_k \mathbf{u}_k \mathbf{v}_k^T, \quad (r = \text{rank}(C)) \quad (5.6)$$

where $\mathbf{u}_k$ and $\mathbf{v}_k$ are the unitary left and right singular vectors, and s_k is the corresponding singular value. This time, C is not guaranteed to be symmetrical, so $\mathbf{u}_k$ and $\mathbf{v}_k$ are unlikely to be equal although $\{\mathbf{u}_k\}$ and $\{\mathbf{v}_k\}$ are both orthogonal sets.

Figure 5.7: Error Face, $C = E_i\{M_i - \bar{M}\}$, from a computer.

We can then select the first K singular vector pairs corresponding to the largest singular values to construct the 2D projection bases for feature extraction. Thus, each image can be represented as

$$M - \bar{M} \approx \sum_{k=1}^K a_k \mathbf{u}_k \mathbf{v}_k^T, \quad (5.7)$$

where $\mathbf{u}_k \mathbf{v}_k^T$ is a pair of 2D projection bases. Applying the orthogonality of $\mathbf{u}_k$ and $\mathbf{v}_k$, the linear projection of an image M to the feature space can thus be obtained as a K -dimensional vector $\mathbf{a} = \{a_k\}$, where

$$a_k = \mathbf{u}_k^T (M - \bar{M}) \mathbf{v}_k. \quad (5.8)$$

The extracted feature vector $\mathbf{a}$ can then be used for classification.

Fig. 5.8 shows some sample 2D projection basis for the Error Face Method corresponding to the largest singular values. They contain random-like high spectral information, but as shown in the experiments, can still yield good classification rates (with a statistical average of 90%). This accidental finding led us to believe that no matter for 1D or 2D projection, the dimension reduction process can be less dependent on the data distribution. In other words, it tells us that PCA and LDA are not the only methods to find good subspace directions. Even a randomly generated direction has a probability of being good, and can lead to more organized and rationalized random approaches such as with genetic algorithms. A random but organized search could be a way of handling high dimension and large data sets.

Experiments and Discussion

The performance of the Error Face Method is evaluated on the ORL Face Database following the same experiment setting as the previous section. The classifiers involved are also NN, NC and NFL.

Figure 5.8: Example of First 18 2D Projection Basis for the Error Face Method.

Fig. 5.9 and 5.10 shows the classification results and the performance deviation of the Error Face Method over 20 random runs. It can be seen that the method also perform the best with the NFL classifier, although it demonstrates no saturation pattern. This might be due to its “random” nature, whereby the useful information is randomly captured by the orthogonal bases sets. Using NFL, the classification rates of the Error Face Method runs up to 92% at full feature size, which is much better than the 88% of the Covariance Face Method and is quite close to the 95% of PCA as shown in Fig. 5.3 and 5.5. The better performance of the Error Face Method than the Covariance Face Method suggests that a good bases derivation can be based on partial spectral information of pixel value distributions even if it is mainly in the high spatial frequency range. This shows that the RP method works well even in 2D projection.

5.3.3 The Effect of Image Size

In the previous two sections, both the Covariance Face Method and the Error Face Method cannot outperform PCA, which could be due to the low dimensional data space they use for feature extraction. Here, we prove this point by testing the classification results with the image size changed from 32×32 to 224×184 in 4 steps. This effectively enlarges the feature space dimension of the 2D subspace approaches from 32 to 224. Note that the given image size is 112×92 , thus, when resizing face images to 224×184 , virtually there is no extra information added.

Figure 5.9: Performance of Error Face Method at Different Feature Size.

Figure 5.10: Performance Deviation of Error Face Method at Different Feature Size.

Experiments and Discussion

Fig. 5.11 and 5.12 compares the classification performances of PCA, the Covariance Face Method and the Error Face Method with NN and NFL classifier under 4 different image sizes. The results use all the available features of each method and are

Figure 5.11: Performance of Various Methods over Different Image Size (NN Classifier).

Figure 5.12: Performance of Various Methods over Different Image Size (NFL Classifier).

averaged over 20 random runs. According to the figures, PCA performance does not vary much with image size. Even for 32×32 images, its data space dimension is already 1024, which seems to be enough to encode the necessary class-relevant information. However, for the two 2D subspace approaches tested here, the input data dimension is only 32 for 32×32 images, which is too compact a representation. Thus, by upsizing the image, the classification performances of both the two 2D subspace approaches have been greatly enhanced with the expanded feature space.

It shall be noted that the most significant improvement from upsizing image is found in the Error Face Method, especially with the NFL classifier, where the accuracy consistently goes up to 94.2% for 224×184 images compared to the 95.5% of PCA on an equivalent setting. However, for the Covariance Face Method, the improvement is not as significant, and its performance even drops down a lot from 112×92 to 224×184 images, where virtually no new information is added. We suspect that the full feature size may not give the best performance for the Covariance

Face Method, yet for the Error Face Method, more face pattern details can still be captured in 224×184 images due to its pseudo-random nature. In the next two sections, we will explore other means to enhance the performance of the Covariance Face Method.

5.3.4 The Discriminant Covariance Face Method

In this section, class-discriminant analysis is applied to the Covariance Face Method to see how much it can help retain more class relevant information. The proposed Discriminant Covariance Face Method is implemented using a variation of LDA called “Direct LDA” from Yang et. al. [143], because 1) it reduces orthogonal feature sets so that we can implement our 2D bases projection under the framework of Spectral-Face Analysis; and 2) it avoids a separate PCA dimension reduction step of the input data space, so that we can fairly compare the enhancement of class-discriminant analysis on PCA and the Covariance Face Method. Note that this approach does not suit the Error Face Method, as its derivation does not take the form of scattering matrices.

The Direct LDA

Instead of solving a generalized eigen-decomposition problem, Yang et. al.’s Direct LDA solves the Fisher discriminant function by diagonalizing the between-class and within-class scattering matrices S_B and S_W simultaneously [143]. They claim that the null space of S_W may contain useful information if the projection of S_B is not zero in that direction, but the null space of S_B can be safely discarded. Thus, they diagonalize S_B and S_W in the way that

$$W^T S_W W = D_W, \quad \text{and} \quad W^T S_B W = I, \quad (5.9)$$

where D_W is a diagonal matrix and I is the identity matrix. This effectively discards the null space of S_B .

To solve for W in Eq. 5.9, they first eigen-decompose S_B as

$$S_B = V D_B V^T, \quad (5.10)$$

where the columns of V are consist of the eigenvectors, and D_B is a diagonal matrix containing the corresponding eigenvalues. Applying orthogonality of V , we have

$$(D_B^{-1/2} V^T) S_B (V D_B^{-1/2}) = I. \quad (5.11)$$

Then let $Z = V D_B^{-1/2}$, $Z^T S_W Z$ can also be eigen-decomposed as

$$Z^T S_W Z = U D_W U^T. \quad (5.12)$$

It can be checked that

$$U^T Z^T S_W Z U = D_W, \quad \text{and} \quad (5.13)$$

$$U^T Z^T S_B Z U = I. \quad (5.14)$$

Thus, the total LDA transformation W can be found as

$$W = ZU. \quad (5.15)$$

The projection of a input data $\mathbf{x}$ to the extracted feature space can then be computed as

$$\mathbf{y} = D_W^{-1/2} W^T \mathbf{x}, \quad (5.16)$$

where the term $D_W^{-1/2}$ is used to sphericalise the projection values.

Note that since $(ZU)^T(ZU) = I$, the Direct LDA features contained in the columns of W form orthogonal sets. As the objective is to maximize $\mathcal{J} = \frac{|W^T S_B W|}{|W^T S_W W|}$, those eigenvectors corresponding to the smallest eigenvalues in D_W are selected as the feature directions. Yang et. al. demonstrated that their Direct LDA approach unifies the PCA and LDA processes, while can retain equivalently good class separability.

Formulation of Discriminant Covariance Face Method

The class-discriminant version of the Covariance Face Method can be obtained by directly substituting the 2D image representation into the between-class scattering and within-class scattering matrices as

$$S_B = E_i\{(M_i - \bar{M})(M_i - \bar{M})^T\}, \quad \text{and} \quad (5.17)$$

$$S_W = E_k E_i\{(M_i - \bar{M}_k)(M_i - \bar{M}_k)^T\} \quad (\forall M_i \in \mathcal{C}_k), \quad (5.18)$$

where M is the image matrix, $\bar{M}$ and $\bar{M}_k$ are the average images of all samples or the Class k samples ($\forall M_i \in \mathcal{C}_k$) respectively. With 2D image representation, S_B and S_W are now of lower dimension and are more likely to be non-singular even with small number of training samples.

Here, we apply the Direct LDA approach mentioned previously to solve the Fisher Discriminant Function with the new scattering matrices, so that we can obtain orthogonal feature sets for 2D basis projection. Since both S_B and S_W are positive semi-definite, the procedure of finding the total transformation matrix W is the same as that of Direct LDA (c.f. Eq. (5.9) to (5.15)). The difference comes when we compute the 2D projection basis and extract the features for Discriminant Covariance Face Method. Considering feature directions $\mathbf{w}_k$ that comes from the columns of W , the covariance face can again be decomposed into the 2D feature space as

$$(M - \bar{M})(M - \bar{M})^T \approx \sum_{k=1}^K a_k d_k^{1/2} \mathbf{w}_k \mathbf{w}_k^T, \quad (5.19)$$

where d_k is the k -th diagonal components of D_W , and K is the selected feature size. Since W is orthogonal, the projection of a covariance face $(M - \bar{M})(M - \bar{M})^T$ to the discriminant 2D projection basis $\mathbf{w}_k \mathbf{w}_k^T$ can be computed as

$$a_k = d_k^{-1/2} \mathbf{w}_k^T (M - \bar{M})(M - \bar{M})^T \mathbf{w}_k. \quad (5.20)$$

Thus, for K feature sets, we can have a K -dimensional projection vector $\mathbf{a}$ with $(\mathbf{a})_k = a_k$ for each image to use for classification.

Table 5.1: Enhancement with Class-Discriminant Analysis.

Method	PCA	Direct-LDA	Covariance Face	Discriminant Covariance Face
Accuracy(%)	94	95.69	84.4	90.5

Experiments and Discussion

Table 5.1 compares the performances of PCA and the Covariance Face Method to their class-discriminant versions of Direct LDA and the Discriminant Covariance Face Method. We use the ORL Face Databases with 64×64 images. The training pool size is 5 images per person and the rest of the images are used for testing. The results presented are averaged over 100 random selection of the training images. All the methods compared use the maximal available feature size for classification.

It can be seen that with the class-discriminant analysis, the performance of the Covariance Face Method can be boosted up by about 6%, while the difference between PCA and Direct LDA is less than 2%. This indicates that the features from the Covariance Face Method, though only based on the left-eigenvectors, can actually preserve class relevant information. Nevertheless, the performances of the 2D subspace approach still appear to be lower than its 1D counterparts. This could be due to its lower feature space dimension; therefore, in next section, we try to scale it up using circulant stacking.

5.3.5 Feature Space Expansion with Image Stacking

The Idea

Our proposed 2D subspace approaches often gives poorer performance than PCA due to the compact feature space they work with. In the experiments of Section 5.3.2, we have shown that the feature space dimension can be enlarged using larger image size and make the performances of the 2D subspace approaches closer to that of PCA. However, for the Covariance Face Method, this improvement cannot be extended if it exceeds the original image size as virtually no extra information is added. In this section, we propose a “circulant stacking” method to enlarge the input data dimension beyond the original image size, which in effect increases the feature space dimension of the 2D subspace method. This stacking technique is designed to govern a homogeneous increment of information in the feature space. We apply it to the Covariance Face Method to see whether it can help minimize its performance gap to PCA.

For circulant stacking, a face image is first cut into $L \times L$ smaller parts, each

Figure 5.13: Circulant Stacking of a Face Image. (a) Original image ($m \times n$ pixels); (b) $2m \times 2n$ stacking by circulant expansion; (c) schematic description of the transformation from (a) to (b).

assigned with a serial number. The sub-images are then piled up to form a larger template according their part numbers, which are now arranged into a circulant matrix. Fig. 5.13 gives an example of circulant stacking with $L = 2$. In general, this technique can reshape the face images from $m \times n$ to $Lm \times Ln$, thus, it gives the freedom to enlarge the current feature space dimension by L times. Although this increases computation, we mainly use it to verify our hypothesis that with increased dimensionality of the feature space, one could expect increased separability of class samples and thereby increased recognition rates. The experiment to be described later does show some improvements.

Theoretical Justification of Image Stacking

For the Covariance Face Method, the 2D-Covariance Matrix computed by Eq. 5.1 actually captures the row correlations of the images. Therefore, an expansion of the row information would help the Covariance Face Method to obtain better features. Image stacking can help in this case as it combines rows from different indexes and therefore passed more correlations into Eq. 5.1 for analysis. The position and the sequence of the stacking may not be important, but it is preferred to manipulate the row portions in a symmetrical way, so that the information is dispersed equally. Here, we use the image stacking scheme proposed above to achieve the expansion of the feature space.

Figure 5.14: Performance of the Covariance Face Method with $4m \times 4n$ Circulant Stacking.

Experiments and Discussion

In this section, we apply $4m \times 4n$ circulant stacking to 112×92 images to see how this can help boost up the performance of the Covariance Face Method. This effectively expand the feature space dimension from 112 to 448.

Fig. 5.14 compares the results from different feature size using NFL classifier, and the results are averaged over 20 random runs. It can be seen that at larger feature space, the performance of the Covariance Face Method can be boosted up to 93.5% at the optimal feature size of 120, after which, adding more features actually gives poorer results. This conforms with our hypothesis that the higher feature space dimension can better preserve class-relevant information. However, careful feature selection shall be incorporated to achieve the best possible performance. In practice, this stacking technique may only be moderately used to boost up the performance. Expanding with a factor greater than 4 would involve too much computations in both basis training and face projection.

5.3.6 SVD Basis for Facial Pose Estimation and Face Recognition

Many 2D subspace approaches concern with the projections of images onto a subspace generated by Singular Value Decomposition (SVD) [56] [17] [150]. In this section, we evaluate how the SVD bases of face images capture 2D structural face information, and propose possible usages of their invariance properties for face recognition and facial pose estimation.

We found that these 2D image bases provide very compact representations, and they capture vertical and horizontal pixel value distributions at different spectral level. Some of the SVD bases demonstrate good invariance properties corresponding to the facial features. Our preliminary tests show that they might be useful for face recognition and extraction of pose-related information. The details are given in the follows.

Figure 5.15: Singular Value Decomposition (SVD) on a Face Image.

Spatial Information Captured in Image SVD Basis

Any matrix M can always be decomposed by Singular Value Decomposition as given by

$$M = U S V^T = \sum_{k=1}^r s_k \mathbf{u}_k \mathbf{v}_k^T, \quad (r = \text{rank}(M)) \quad (5.21)$$

where s_k is a non-zero singular value, and $\mathbf{u}_k$ and $\mathbf{v}_k$ called left singular vector (SV) and right singular vector. Usually, $\mathbf{v}_k$ is first found which maximizes $s_k = \|M\mathbf{v}_k\|$, and $\mathbf{u}_k$ are then computed as $\mathbf{u}_k = \frac{M\mathbf{v}_k}{\|M\mathbf{v}_k\|}$. For a matrix with rank r , there are altogether r pairs of left and right SVs with non-zero singular values. The computation routine of SVD ensures that both $\{\mathbf{u}_k\}$ and $\{\mathbf{v}_k\}$ consist of unitary orthogonal sets.

If we use M to represent an image, $\mathbf{u}_k$ and $\mathbf{v}_k$ can be treated as SVD basis of that image. Fig. 5.15 gives a graphical interpretation of the SVD process on a 112×92 face image. In this sense, each pair of the SVD bases, $\mathbf{u}_k$ and $\mathbf{v}_k$, actually defines some spatial spectral (structural) information of the image gray-value distribution that is orthogonal to the spectral information captured by other pairs. Whereas, the corresponding singular value measures the strength of the estimated structures in that image.

Without loss of generality, we first analyze the singular values and the SVD bases of an average face image taken from a training set of the ORL database. Fig. 5.16 shows the average face together with all the singular values s_k derived from it, while the ensembles of the left and the right SVs, $\mathbf{u}_k$ and $\mathbf{v}_k$ are given in Fig. 5.17 and Fig. 5.18. There are two major observations: 1) The SVD basis can capture structural information of an image in a highly compact way. This is evident by the rapidly diminishing s_k values. From Fig. 5.16, the values after s_8 are almost zero and should have a negligible effect on image reconstruction. 2) The left and the right SVs depict the pixel value distribution in vertical and horizontal directions at different spectral levels. Only the first few (say, 8, as shown in Fig. 5.17 and 5.18)

Figure 5.16: Average Face and its Singular Values s_k . Figure 5.17: $\mathbf{u}_k$ of Average Face (First 17 and Last 3).

SVD bases contain global structural information. The rest of them focus more and more on high-frequency local details which shall be less relevant for classification.

From the above analysis, we can see that using SVD basis may have a few advantages in face recognition and face image analysis. Firstly, the first few SVD bases only contain some global structure of vertical or horizontal side of the image and shall be less sensitive to local appearance changes such as wearing spectacles

Figure 5.18: $\mathbf{v}_k$ of Average Face (First 17 and Last 3).

and mild expression variation. Secondly, due to the unitary nature of the SVD bases, their values are normalized to a common range, which shall be less affected by overall light intensity change (not the light angle change) of the image. Thirdly, the vertical or horizontal basis on certain spectrum may reveal pose information such as eye and nose positions, which could be useful for facial pose estimation. In the experiments below, we present some preliminary investigation results with respect to observations.

Experiments and Discussion

To evaluate the invariant nature and facial feature correspondence of the SVD bases, we first compared vertical and horizontal SVs ($\mathbf{u}_k$ and $\mathbf{v}_k$) at various spectrum across different class samples. The images used are from the ORL face database [32]. Fig. 5.19 shows the first pair of the bases for some images of two persons, where $\mathbf{u}_1$ and $\mathbf{v}_1$ are plotted along vertical or horizontal directions to show correspondence to the face images. We can see that $\mathbf{u}_1$ takes approximately the form of a face profile shape in vertical direction, which is quite person-invariant regardless of some in-depth face rotations (i.e. yawing about the vertical axis), wearing of spectacles or some light intensity change. This is because $\mathbf{u}_1$ corresponds to the largest singular value and represents the most important vertical global structure of a face image, thus, the small local variations get ignored at this spectral level. This partial view independence of the vertical face profile is in fact a very good property considering

Figure 5.19: Face Images and the Corresponding $\mathbf{u}_1$ and $\mathbf{v}_1$. Figure 5.20: Face Images and the Corresponding $\mathbf{u}_4$ and $\mathbf{v}_4$.

that most current face recognition algorithms are not view independent. Moreover, as $\mathbf{u}_1$ is normalized to be unitary, it will not reflect the overall intensity level changes. For these characteristic properties, we call $\mathbf{u}_1$ the Vertical Face Profile and will later investigate on its usage for compact face representation and classification.

Similarly, we can call $\mathbf{v}_1$ the Horizontal Face Profile. From Fig. 5.19, we can see it gives a face profile along the horizontal direction, with the major arc representing the face area, and a small, near-central peak aligned to the nose tip. However, this horizontal face profile appears less consistent for one person and is more sensitive to lighting angles, rotation of the face, e.t.c.. This is because the facial features projected to the horizontal direction are more cluttered as compared to the vertical case, which also means it will be less distinctive for different persons. But on the other hand, we can use it to trace appearance-related properties of a face image such as horizontal nose position, and a rough direction of the light source (as indicated by the different levels of the $\mathbf{v}_1$ values between left and right face halves).

For most of the subsequent pairs of the SVD bases, there are no obvious pattern for consistent representation of facial features. However, on the 4th pair, we can find clear indication of eye positions from the horizontal basis. Fig. 5.20 shows $\mathbf{u}_4$ and $\mathbf{v}_4$ for the same set of the sample images as in Fig. 5.19. It can be seen that $\mathbf{v}_4$ generally forms two valleys near the centre peak, which could indicate the positions of the two eyes, although $\mathbf{u}_4$ does not really show consistent pattern for each person. It is also possible to use $\mathbf{v}_4$ for the detection of horizontal eye positions.

Last but not least, we test on how well the vertical face profiles (i.e. $\mathbf{u}_1$) can serve as a compact face representation and be used for classification. Fig. 5.21 and 5.22 show the plots of some sample face images for different persons with the corresponding vertical face profiles. They conform with our previous findings that this basis shows consistent pattern for each person, despite of the mild changes in light intensity, pose and expression. We then use the vertical face profiles to represent each image and test their recognition performance on the ORL face database with 112×92 images and simple NN classifier. The average recognition rates over 20 random runs are 92%, which is very good considering the 1D shape representation the vertical face profile takes. This result shows that the first vertical basis alone can possibly serve as a compact representation of one person's face images. Its classification performance can be further improved using proper and more advanced classifiers such as Support Vector Machines (SVM).

5.4 Conclusions

In this chapter, we investigate on 2D subspace methods that use 2D image representation. The prime motivation was to solve the problem of high dimension and very large data set, which upper limits the usefulness of other subspace methods such as PCA and LDA. A framework called the Spectral-Face Analysis is proposed, which consists of two implementations: the Covariance Face Method and the Error Face Method. The initial performances of both the two methods are not as good as that of PCA due to the low dimensional feature space used. We apply class-discriminant analysis and proposed a unique image stacking method for the Covariance Face Method, which effectively boost up its performance by either extracting more class-relevant information or expanding the feature space dimension.

Figure 5.21: Face Images and the Corresponding Vertical Face Profiles ($\mathbf{u}_1$) - Sample Group A.

Figure 5.22: Face Images and the Corresponding Vertical Face Profiles ($\mathbf{u}_1$) - Sample Group B.

An interesting finding is that the Error Face Method, which actually implements 2D random projection, can achieve good classification performance. Moreover, its performance can be consistently boosted up by using larger images, even beyond the original image size. We also look into the SVD bases of face images and find some invariance properties corresponding to certain facial features. Some of these bases may serve as good compact representation of face images. Our preliminary tests show that they might be useful for face recognition and extraction of pose-related information.

Since the proposed 2D subspace methods demonstrate best performances with a classifier having good generalization ability, in the future, we can try to use SVM with them to see whether this can further improve their performance. To generalize our findings, we can also test them on other benchmark face databases. For the use of SVD bases of face images, further investigations are needed to find out how they can actually contribute to classification and pose estimation. For example, one may use the facial feature locations that reside on certain SVD basis to normalize face positions, while use the invariance properties of other representative SVD bases to pre-group faces samples into sub-clusters. By this hierarchical approach, we can break large data sets to smaller, manageable clusters, and better class separation could also be achieved. Last but not least, it is worthwhile to study the connection between our 2D subspace methods and the recently proposed ones in [140] [73] and [72] to find out what is the working factor that makes their methods achieve better classification results.

Chapter 6

Conclusions and Recommendation

In this chapter, the conclusions of the thesis are summarized. The contents of the previous chapters are reviewed. Some recommendations for future work are also discussed.

6.1 Conclusions

Subspace methods were shown to be very efficient in reducing data dimension and extracting statistical features. Over the years, it is widely applied to pattern classification problems with multi-class labels. Recently, some of the subspace methods such as PCA and LDA have been successfully extended into nonlinear domain using the kernel transformations. With the new flexibility and efficiency in defining the classifier boundary, it is worthwhile to refocus on the subspace methods to furnish up its design principle and resolve some prolonged performance issues.

In this thesis, we first study on how to evaluate the geometric margin measure among the classes for the subspace methods, and derived a method called Margin-Maximization Discriminant Analysis (MMDA). It is based on an additive-form discriminant function $J(\mathbf{w}) = \mathbf{w}^T(S_B - \beta S_W)\mathbf{w}$, which has been proved to be able to approximately maximize the average margin or the non-overlapping regions between the classes with the help of a class-spread regulator β . This additive formulation naturally avoids the singularity problem of S_W that is often encountered in LDA for Small Sample Size (SSS) Problem. As a result, MMDA can achieve direct dimension reduction of the input data in the discriminant domain. This avoids potential loss of discriminant information due to any special treatments of the singularity problem (as often required by LDA and its variants), and is also computationally more effective. A computational trick for MMDA is proposed to make it more effective when data dimension is much larger than the number of the samples, and more importantly, to achieve true dimension reduction in the discriminant domain. This preserves discriminant information and is a key advantage over other LDA variants which have been shown to potentially lose discriminant information. The conditions for MMDA to generate optimal margins/non-overlaps are evaluated, and the nonlinear formulation of MMDA is also derived using the “kernel trick”. Experi-

ments with both toy data and three face databases show that compared to LDA, MMDA can find feature sets with larger average margin between the classes, and often achieve better classification results. Moreover, while LDA is much affected by the pre-process of PCA data dimension reduction and the selection of optimal feature size, MMDA performance is quite stable with large enough β values. As a rule of thumb, we recommend to set $\beta = 9$, which is approximately equivalent to assume most of the class samples are within $3std$ of the class center. Note that the assumption above is statistically valid with almost all kinds of practical data distribution. We have however only showed this for Face databases.

To deal with the “peaking phenomenon” and the overfitting problems that is persistent with LDA and its variants, we looked into feature relevance measurement and proposed a simple but novel feature relevance weighting scheme, called Relevance-Weighted Discriminant Analysis (RWDA), to resolve these issues. Feature relevance is an important consideration in feature selection and feature extraction for pattern classification/recognition research. We propose our definition of feature relevance based on the amount of non-overlap between the classes. This can be an important and correct measure of relevance applicable to classification in general. The advantage of our relevance definition is that it led to analytically computable relevance measures based on Fisher discriminant values and a constant spread regulator M . Moreover, it does not assume any model of sample distribution and can be well applied to multi-class problems. The effectiveness of such relevance weightage is theoretically justified by analyzing the probability of classification error on weighted and unweighted features. Our intensive experiments with the UMIST, ORL and YALE Face Database show that RWDA can completely eliminate the peaking phenomenon of LDA, which alleviates the requirements of selecting optimal feature size. Moreover, RWDA demonstrates better generalization for classification, comparing to the direct use of the same set of LDA features. This analysis suggests a new insight into the root cause of overfitting for classifiers using distance metric, and offers an efficient way to resolve the issue of selecting optimal feature size. Our finding is that as long as a feature has relevance, it can be gainfully used if properly weighted. Its value is commensurate with its relevance to enhance the generalization ability of the classifier. This is a novel finding as compared to the traditional view of overfitting that features after the optimal feature size will only deteriorate the classification results. The RWDA we proposed here extend this new finding and is proved to provide correct relevance measure for difficult multi-classification problems.

To overcome the computation intractability for problems of high dimension and very large data set, the design of 2D subspace methods that use 2D image representation were also explored in this thesis. The proposed framework is called the Spectral-Face Analysis, which uses SVD to compute 2D projection basis and therefore compute projection vectors of each image. Two implementations were presented: the Covariance Face Method, which is based on 2D covariance matrix, and the Error Face Method, which accidentally implements 2D Random Projection for feature extraction. Experiments with the ORL face database show that the per-

performances of both the proposed methods are initially worse than PCA due to the low dimensional feature space used. However, the 2D subspace method can gain more significant improvement than PCA by employing class-discriminant analysis or using larger image sizes. An interesting finding is that the Error Face Method, which is only based on partial spectral information of the training images, can achieve better classification performance than the Covariance Method. This, in general, suggests that good features for classification can be derived from partial spectral information of pixel value distributions. We also proposed a unique image stacking technique to boost up the performance of the 2D subspace methods by expanding the feature space dimension. Lastly, we evaluate the invariant properties of SVD basis from face images as a compact face representation, and discuss their possible usages for face recognition and pose estimation.

6.2 Future Works

1) Integrate MMDA with Relevance Weights

Since MMDA is also a class-discriminant subspace method and it also computes the within-class scatter and the between-class scatter, it is possible to apply the feature relevance weights we proposed for RWDA to the MMDA features to see whether this can further boost up the recognition performance. It is expected that these two proposed ideas can work in tandem and are not isolated or in conflict with each other.

2) Combining MMDA with the Manifold Learning Methods

Manifold Learning methods such as LLE and LPP [112] [53] [27] are good at preserving the local structure of the data distributions, and hence produce compact representation for high dimensional data, but they work irrespective of preserving class-discriminant information. Our MMDA can be applied to the reduced feature set from LLE or LPP to generate features that are better for classification. Hence our work is not in competition with manifold learning but is complementary to this recent learning method. It is interesting to see how these two can work together to resolve large scale classification problems.

3) Apply Relevance Weights to Kernel Extensions of LDA/FDA

The formulation of our proposed relevance measurement in RWDA allows it to be readily extended to the non-linear version of LDA/FDA using the kernel tricks. In the future, we may look into how our relevance measure works with the Kernel Discriminant Analysis (KDA) [141]. Since the kernel mapping implicitly exploded data dimension and in effect make the sample distribution even more sparse, we hope to see the feature usage can be better combined using the relevance weighting. If this is true, it can resolve the problem of selecting optimal feature size for kernel-transformed subspace methods.

4) Extending the Ideas of MMDA and RWDA to 2D subspace methods

Considering the computation intractability for real large-scale classification problems, it is also worthwhile to extend the ideas of MMDA and RWDA to 2D subspace methods. These two methods show better generalization ability than conventional LDA, which can hopefully work better than other 2D subspace approaches.

5) Study the Connection between Our 2D Subspace Methods And The Recently Proposed Ones

The classification results of our 2D subspace methods seem to be worse than the recently developed ones as in [140] [73] and [72]. It is worthwhile to study the connection between our methods and the recently proposed ones to find out the working factor of their methods.

6) Use SVD basis to Form Sub-Clustered Class Samples for Face Recognition

For the use of SVD basis of face images, it is possible to rely on their invariant properties over some facial features to form sub-clusters class samples based on the appearance. In this way, we can treat a class with multiple models, each of smaller within-model variations, which may help to produce more consistent features for face recognition.

7) Perform Experiments On Other Databases

We have tested our algorithms mainly on three well-known face databases. Albeit, these being very difficult high-dimensional database to handle, we would like to further verify that our claims can conclusively be general. To generalize our results, we also recommend to perform experiments on other databases such as FERET face database, UIC, and other benchmark datasets for general multi-class problems. It is expected that our proposed methods should give improved performance over conventional LDA since our proposed algorithms were derived for the general case, not limited to face recognition. As most data base are non-linearly separable or non-separable, kernel transformations must be implemented to MMDA and the relevance weightage of RWDA. For databases with large within-class variation, for example, those follow multi-modal distribution, sub-clustering of class samples may also need to be incorporated in order to show the real strength of MMDA and RWDA. This is because these two methods, though do not assume, but should work better with center-concentrated classes/subclasses, as they are more descriptive for margin and relevance measures. It would be interesting to compare the classification results with or without the sub-clustering of the class samples.

Author's Publications

- [1] Yan Zhu and Eric Sung, "From Spectral-Face to Eigen-Face", *International Conference on Control, Automation, Robotics and Vision (ICCARV)*, Singapore, 2000.
- [2] Yan Zhu and Eric Sung, "The Spectral-Face Analysis for Face Recognition", *The 5th Asian Conference on Computer Vision (ACCV)*, Australia, 2002.
- [3] Yan Zhu and Eric Sung, "Margin-Maximization Discriminant Analysis for Face Recognition", *International Conference on Image Processing (ICIP)*, Vol. 1, pp. 609 - 612, 2004.
- [4] Xuchun Li, Yan Zhu, and Eric Sung, "Sequential Bootstrapped Support Vector Machines – A SVM Accelerator", *IEEE International Joint Conference on Neural Networks(IJCNN)*, Montreal, Canada, 2005.
- [5] Eric Sung, Yan Zhu, and Xuchun Li, "Accelerating the SVM Learning for Very Large Data Sets", *Proc. of International Conference on Pattern Recognition*, 20-24 August 2006, Hong Kong.
- [6] Yan Zhu and Eric Sung, "Relevance-Weighted Discriminant Analysis", *Submitted to Computer Vision and Image Understanding (under minor revision)*.
- [7] Yan Zhu and Eric Sung, "Maximizing Non-Overlap between Classes in Discriminant Analysis", *To submit to Pattern Recognition*.

Bibliography

- [1] D. Achlioptas. Database-Friendly Random Projections. *in ACM Symposium on the Principles of Database Systems*. pp. 274-281, 2001.
- [2] E.L. Allwein, R.E. Schapire, and Y. Singer. Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. *in Journal of Machine Learning Research*. vol. 1, pp. 113-141, 2000.
- [3] F.R. Bach and M.I. Jordan. Kernel Independent Component Analysis. *in Journal of Machine Learning Research*. vol. 3, pp. 1-48, 2002.
- [4] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 19, pp. 711-720, 1997.
- [5] M. Belkin and P. Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *in Neural Computation*. vol. 15(16), pp. 1373-1396, 2003.
- [6] R.E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ., 1957.
- [7] M. Bicego, U. Castellani, and V. Murino. Using Hidden Markov Models and Wavelets for Face Recognition. *in Proceedings of IEEE International Conference on Image Analysis and Processing (ICIAP03)*. pp. 52-56, 2003.
- [8] Z. Biuk and S. Loncaric. Face Recognition from Multi-Pose Image Sequence. *in Proceedings of 2nd Int'l Symposium on Image and Signal Processing and Analysis*. pp. 319-324, 2001.
- [9] A.L. Blum and P. Langley. Selection of Relevant Features and Examples in Machine Learning. *in Artificial Intelligence*. vol. 97(1-2), pp. 245-271, 1997.
- [10] B.E. Boser, I.M. Guyon, and V.N. Vapnik. A Training Algorithm for Optimal Margin Classifiers. *in D. Haussler, editor, Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*. pp. 144-152, Pittsburgh, PA, July 1992. ACM Press.

- [11] K. Bowyer, K. Chang, and P. Flynn. A Survey Of 3D and Multi-Modal 3D+ 2D Face Recognition. *Notre Dame Department of Computer Science and Engineering Technical Report*. January 2004.
- [12] E.J. Bredensteiner and K.P. Bennett. Multicategory Classification by Support Vector Machines. in *Computational Optimizations and Applications*. vol. 12, pp. 35-46, 1999.
- [13] R. Brunelli and T. Poggio. Face Recognition through Geometrical Features. in *Proc. ECCV '92*. pp. 792-800, 1992.
- [14] H. Cevikalp, Wilke M. Neamtu, and A. Barkana. Discriminative Common Vectors for Face Recognition. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 27(1), pp. 4-13, 2005.
- [15] R. Chellappa, C.L. Wilson, and S. Sirohey. Human and Machine Recognition of Faces: A Survey. in *Proceedings of IEEE*. vol. 83(5), pp. 705-740, 1995.
- [16] P.-H. Chen, C.-J. Lin, and B. Schölkopf. A Tutorial on ν -Support Vector Machines. in *Applied Stochastic Models in Business and Industry*. vol. 21, pp. 111-136, 2005.
- [17] Y.Q. Cheng, K. Liu, J.Y. Yang, and H.F. Wang. A Robust Algebraic Method for Human Face Recognition. in *Proceedings, International Conference on Pattern Recognition*. vol. 2, pp.221-224, 1992.
- [18] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active Shape Models - Their Training and Application. in *Computer Vision and Image Understanding*. vol. 61, pp. 18-23, 1995.
- [19] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active Appearance Models. in *IEEE Trans. on Pattern Analysis and Machine Intelligence*. vol. 23, pp. 681-685, 2001.
- [20] C. Cortes and V. Vapnik. Support vector networks. in *Machine Learning*. vol. 20, pp. 273-297, 1995.
- [21] I.J. Cox, J. Ghosn, and P.N. Yianilos. Feature-Based Face Recognition Using Mixture-Distance. in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pp. 209-216, 1996.
- [22] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman & Hall, London, 1994.
- [23] P. Cunningham. Machine Learning for Access and Retrieval. *Lecture Notes*. Machine Learning Group, Trinity College Dublin.

- [24] W. Daelemans, S. Gillis, and G. Durieux. The Acquisition of Stress: a Data-Oriented Approach. in *Computational Linguistics*. vol. 20(3), pp. 421-451, 1994.
- [25] D.Q. Dai and P.C. Yuen. Regularized Discriminant Analysis and its Application to Face Recognition. in *Pattern Recognition*. vol. 36, pp. 845-847, 2003.
- [26] S. Dasgupta and A. Gupta. An Elementary Proof of the Johnson-Lindenstrauss Lemma. in *Technical Report TR-99-006*. International Computer Science Institute, Berkeley, CA, 1999.
- [27] D. de Ridder and R.P.W. Duin. Locally Linear Embedding for Classification. *Technical Report PH-2002-01, Pattern Recognition Group, Department of Imaging Science and Technology, Delft University of Technology*. The Netherlands, 2002.
- [28] T.G. Dietterich and G. Bakiri. Solving Multiclass Learning Problems via Error-Correcting Output Codes. in *Journal of Artificial Intelligence Research*. vol. 2, pp. 263-286, 1995.
- [29] B.A. Draper, K. Baek, M.S. Bartlett, and J.R. Beveridge. Recognizing Faces with PCA and ICA. in *Computer Vision and Image Understanding*. vol. 91, pp.115-137, 2003.
- [30] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification (2nd ed.)*. New York: John Wiley & Sons, 2001.
- [31] R.P.W. Duin. Classifiers in Almost Empty Spaces. in *International Conference on Pattern Recognition (ICPR'00)*. vol. 2, pp. 2001-2007, 2000.
- [32] The ORL face database. [online]. available: <http://www.uk.research.att.com/facedatabase.html>.
- [33] The UMIST face database. [online]. available: <http://images.ee.umist.ac.uk/-danny/database.html>.
- [34] The YALE face database. [online]. available: <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>.
- [35] R.-E. Fan, P.-H. Chen, and C.-J. Lin. Working Set Selection Using the Second Order Information for Training SVM. in *Journal of Machine Learning Research*. vol. 6, pp. 1889-1918, 2005.
- [36] S. Fine and K. Scheinberg. Efficient SVM Training Using Low-Rank Kernel Representations. in *Journal of Machine Learning Research*. vol. 2, pp. 243-264, 2001.

- [37] M. Fleming and G. Cottrell. Categorization of Faces Using Unsupervised Feature Extraction. in *Proc. International Joint Conference on Neural Networks IEEE IJCNN*. pp. 65-70, 1990.
- [38] B.J. Frey, A. Colmenarez, and T.S. Huang. Mixtures of Local Linear Subspaces for Face Recognition. in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*. pp. 32-37, 1998.
- [39] J.H. Friedman. Regularized discriminant analysis. *J. of the American Statistical Assoc.* vol. 84, pp. 165-175, 1989.
- [40] K. Fukunaga. Introduction to Statistical Pattern Recognitions. 2nd Ed., Academic Press, Inc., 1990.
- [41] Y. Gao and M. Leung. Face recognition using line edge map. in *IEEE Trans. Pattern Analysis and Machine Intelligence*. vol. 24(6), pp. 764-779, 2002.
- [42] J.L. Goldberg. *Matrix Theory with Applications*. McGraw-Hill International Edition, Chapter 8, 1992.
- [43] A.J. Goldstein, L.D. Harmon, and A.B. Lesk. Identification of Human Faces. in *Proc. IEEE*. vol. 59, pp. 748, 1971.
- [44] S. Gong, S.J. McKenna, and A. Psarrou. *Dynamic Vision: From Images to Face Recognition*. Imperial College Press, London, 2000.
- [45] G.G. Gordon. Face Recognition Based on Depth and Curvature Features. in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pp. 808-810, 1992.
- [46] G.D. Guo, S.Z. Li, and K.L. Chan. Face Recognition by Support Vector Machines. in *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*. pp. 196-201, 2000.
- [47] P. Gustafson, P. Carbonetto, N. Thompson, and N. de Freitas. Bayesian Feature Weighting for Unsupervised Learning, with Application to Object Recognition. in *Proc. Ninth Intl Workshop Artificial Intelligence and Statistics (AISTAT03)*. 2003.
- [48] I. Guyon and A. Elisseeff. An Introduction to Variable and Feature Selection. in *Journal of Machine Learning Research*. vol. 3, pp. 1157-1182, 2003.
- [49] A. Hadid and M. Pietikäinen. From Still Image to Video-Based Face Recognition: An Experimental Analysis. in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FGR04)*. pp. 813-818, 2004.

- [50] F. Han and C.S. Chua. 3D Race Recognition Using Point Signature. *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*. 2000.
- [51] P.J.B. Hancock, V. Bruce, and A.M. Burton. Recognition of Unfamiliar Faces. *Trends in Cognitive Sciences*. Vol. 4, pp. 330-337, 2000.
- [52] L.D. Harmon, M.K. Khan, R. Lasch, and P.F. Ramig. Machine Identification of Human Faces. *in Pattern Recognition*. vol. 13, pp. 97-110, 1981.
- [53] X.F. He and P. Niyogi. Locality Preserving Projections. *in Proceedings of the Conference on Advances in Neural Information Processing Systems*. 2003.
- [54] B. Heisele, P. Ho, and T. Poggio. Face Recognition with Support Vector Machines: Global versus Component-Based Approach. *in Proceedings of IEEE International Conference on Computer Vision*. pp.688-694, 2001.
- [55] Z.-Q. Hong and J.-Y. Yang. Optimal Discriminant Plane for a Small Number of Samples and Design Method of Classifier on the Plane. *Pattern Recognition*. vol. 24(4), pp. 317-324, 1991.
- [56] Z.Q. Hong. Algebraic Feature Extraction of Image for Recognition. *in Pattern Recognition*. vol. 24(3), pp.211-219, 1991.
- [57] H. Hotelling. Analysis of a Complex of Statistical Variables into Principal Components. *in Journal of Educational Psychology*. vol. 24, pp. 498-520, 1933.
- [58] A.J. Howell and H. Buxton. Invariance in Radial Basis Function Neural Networks in Human Face Classification. *Neural processing letters*. vol. 2(3), pp. 26-30, 1995.
- [59] J. Hua, Z. Xiong, J. Lowey, E. Suh, and E.R. Dougherty. Optimal Number of Features as a Function of Sample Size for Various Classification Rules. *in Bioinformatics*. vol. 21, no. 8, pp. 1509-1515(7), April 2005.
- [60] J. Huang, V. Blanz, and B. Heisele. Face Recognition Using Component-Based SVM Classification and Morphable Models. *in SVM 2002*. LNCS 2388, pp. 334-341, 2002.
- [61] R. Janardan J. Ye, C.H. Park, and H. Park. An Optimization Criterion for Generalized Discriminant Analysis on Undersampled Problems. *in IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 26(8), pp. 982-994, August 2004.
- [62] T. Joachims. Making Large-Scale Support Vector Machine Learning Practical. *in B. Schölkopf, C. Burges and A. Smola, eds, Advances in Kernel Methods-Support Vector Machines*. MIT Press, Cambridge, MA, 1998.

- [63] G.H. John, R. Kohavi, and K. Pfleger. Irrelevant Features and the Subset Selection Problem. in *Proceedings of the Eleventh International Conference on Machine Learning*. pp. 121-129, 1994.
- [64] W. Johnson and J. Lindenstrauss. Extensions of Lipschitz Mapping into a Hilbert Space. in *Conference on Modern Analysis and Probability*. pp. 189-206, 1984.
- [65] T. Kanade. Picture Processing by Computer Complex and Recognition of Human Faces. *Technical Report, Kyoto University, Dept. of Information Science*. 1973.
- [66] G.J. Kaufman and K.J. Breeding. Automatic Recognition of Human Faces from Profile Silhouettes. in *IEEE Trans. on Systems, Man and Cybernetics*. SMC-6(2), pp. 113-121, 1976.
- [67] S. Kawaguchi and R. Nishii. Hyperspectral Image Classification by Bootstrap AdaBoost With Random Decision Stumps. in *PAMI*. vol. 45(11), pp. 3845-3851, 2007.
- [68] Y. Kaya and K. Kobayashi. A Basic Study on Human Face Recognition. in *S. Watanabe, Editor, Frontiers of Pattern Recognition*. pp. 265, 1972.
- [69] S. Keerthi, S. Shevade, C. Bhattacharyya, and K. Murthy. A Fast Iterative Nearest Point Algorithm for Support Vector Machine Classifier Design. in *IEEE Transactions on Neural Networks*. vol. 11(1), pp. 124-136, 2000.
- [70] R. Kohavi, P. Langley, and Y. Yun. The Utility of Feature Weighting in Nearest-Neighbor Algorithms. in *Proceedings of the Ninth European Conference on Machine Learning*. Prague: Springer-Verlag, 1997.
- [71] V.V. Kohir and U.B. Desai. Face Recognition Using DCT-HMM Approach. in *Workshop on Advances in Facial Image Analysis and Recognition Technology (AFIART)*. June 1998.
- [72] H. Kong, E.K. Teoh, J.-G. Wang, and C. Kambhamettu. Generalized 2D Fisher Discriminant Analysis. in *British Machine Vision Conference (BMVC05)*. 2005.
- [73] H. Kong, L. Wang, E.K. Teoh, X. Li, J.-G. Wang, and R. Venkateswarlu. Generalized 2D Principal Component Analysis for Face Image Representation and Recognition. in *Neural Networks*. vol. 18(5-6), pp. 585-594, 2005.
- [74] H. Kramer and M. Mathews. A Linear Coding for Transmitting a Set of Correlated Signals. in *IRE Trans. on Information Theory*. vol. 2, pp. 41-46, 1956.

- [75] V. Krueger and S. Zhou. Exemplar-Based Face Recognition from Video. *in Fifth IEEE International Conference on Automatic Face and Gesture Recognition*. pp. 175-180, 2002.
- [76] W.J. Krzanowski, P. Jonathan, W.V McCarthy, and M.R. Thomas. Discriminant Analysis with Singular Covariance Matrices: Methods and Applications to Spectroscopic Data. *in Applied Statistics*. vol. 44, pp. 101-115, 1995.
- [77] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic Interpretation and Coding of Face Images using Flexible Models. *in IEEE Trans. on Pattern Analysis and Machine Intelligence*. vol. 19(7), pp. 734-756, 1997.
- [78] M.H.C. Law, M.A.T. Figueiredo, and A.K. Jain. Simultaneous Feature Selection and Clustering Using Mixture Models. *in IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 26(9) September 2004.
- [79] S. Lawrence, C.L. Giles, A. Tsoi, and A. Back. Face Recognition: A Convolutional Neural Network Approach. *in IEEE Trans. on Neural Networks*. vol. 8(1), pp. 98-113, 1997.
- [80] J.C. Lee and E. Milios. Matching Range Image of Human Faces. *Third International Conference on Computer Vision*. pp. 722-726, 1990.
- [81] Y. Lee, Y. Lin, and G. Wahba. Multicategory Support Vector Machines: Theory and Application to the Classification of Microarray Data and Satellite Radiance Data. *in Journal of the American Statistical Association*. vol. 99, pp. 67-81, 2004.
- [82] H. Li, T. Jiang, and K. Zhang. Efficient and Robust Feature Extraction by Maximum Margin Criterion. *in Advances in Neural Information Processing Systems*. vol. 16, 2003.
- [83] S.Z. Li and J. Lu. Face Recognition Using the Nearest Feature Line Method. *in IEEE Transactions on Neural Networks*. vol. 10(2), pp. 439-443, 1999.
- [84] C. Liu and H. Wechsler. A Gabor Feature Classifier for Face Recognition. *in Proceedings of Eighth IEEE International Conference on Computer Vision*. vol. 2, pp. 270-275, 2001.
- [85] C. Liu and H. Wechsler. A Shape and Texture Based Enhanced Fisher Classifier for Face Recognition. *in IEEE Trans. Image Processing*. vol. 10(4), pp. 598-608, 2001.
- [86] C. Liu and H. Wechsler. Enhanced Fisher Linear Discriminant Models for Face Recognition. *in Proc. the 14th International Conf. on Pattern Recognition*. 1998.

- [87] C. Liu and H. Wechsler. Evolutionary Pursuit and its Application to Face Recognition. in *IEEE Trans. on Pattern Analysis and Machine Intelligence*. vol. 22(6), pp. 570-582, 2000.
- [88] X. Liu and T. Chen. Video-Based Face Recognition Using Adaptive Hidden Markov Models. in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR03)*. pp. 340-345, 2003.
- [89] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Face Recognition Using Kernel Direct Discriminant Analysis Algorithms. in *IEEE Transactions on Neural Networks*. vol. 14(1), pp. 117-126, Jan. 2003.
- [90] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Regularized Discriminant Analysis for the Small Sample Size Problem in Face Recognition. *Pattern Recognition Letter*. vol. 24(16), pp.3079-3087, 2003.
- [91] B. Ma and H.-S. Wong. 2D Clustering Based Discriminant Analysis for 3D Head Model Classification. in *Pattern Recognition*. vol. 39(3), pp. 491-494, 2006.
- [92] A.M. Martinez and A.C. Kak. PCA versus LDA. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 23(2), pp. 228-233, 2001.
- [93] T. Martiriggiano, M. Leo, T. D'Orazio, and A. Distanti. Face Recognition by Kernel Independent Component Analysis. in *Proceedings of the 18th international conference on Innovations in Applied Artificial Intelligence*. pp. 55-58, 2005.
- [94] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.
- [95] D. Modha and W. Scott-Spangler. Feature Weighting in k-Means Clustering. in *Machine Learning*. vol. 52(3), pp. 217-237, 2003.
- [96] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian Face Recognition. in *Pattern Recognition*. vol. 22(2000), pp. 1771-1782, 2000.
- [97] B. Moghaddam and A.P. Pentland. Probabilistic Visual Learning for Object Representation. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 19(7), pp. 696-710, 1997.
- [98] Bartlett M.S, J.R. Movellan, and T.J. Sejnowski. Face Recognition by Independent Component Analysis. in *IEEE Transactions on Neural Networks*. vol. 13(6), pp.1450-1464, 2002.
- [99] G. Bebis N. Goel and A.V. Nefian. Face Recognition Experiments with Random Projections. in *SPIE Conference on Biometric Technology for Human Identification*. 2005.

- [100] O. Nakamura, S. Mathur, and T. Minami. Identification of Human Faces Based on Isodensity Maps. *in Pattern Recognition*. vol. 24(3), 263-272, 1991.
- [101] A.V. Nefian and H.H. Hayes III. Hidden Markov Models for Face Recognition. *in IEEE International Conference on Acoustic, Speech and Signal Processing*. vol. 5, pp. 2721-2724, May 1998.
- [102] E. Oja. *Subspace Methods of Pattern Recognition*. Research Studies Press, 1983.
- [103] M. Pei, E.D. Goodman, and W.F. Punch. Feature Extraction Using Genetic Algorithms. *in Proceeding of International Symposium on Intelligent Data Engineering and Learning'98 (IDEAL'98)*. 1998.
- [104] P. Penev and J. Atick. Local Feature Analysis: a General Statistical Theory for Object Representation. *in Neural Systems*. vol. 7(477), pp. 500, 1996.
- [105] A. Pentland, B. Moghaddam, and T. Starner. View-Based and Modular Eigenspaces for Face Recognition. *in IEEE Conf. Computer Vision and Pattern Recognition*. pp. 84-91, 1994.
- [106] P.J. Phillips, P. Grother, R. Micheals, D.M. Blackburn, E. Tabassi, and J.M. Bone. FRVT 2002 Evaluation Report. *Technical Report*. March, 2003, available: <http://www.frvt.org>.
- [107] P.J. Phillips, W.T. Scruggs, A.J. OToole, P.J. Flynn, K.W. Bowyer, C.L. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 Large-Scale Results. *NIST-TIR 7408*,. March 2007.
- [108] J. Platt. Fast Training of Support Vector Machines Using Sequential Minimal Optimization. *in B. Schölkopf, C. Burges and A. Smola, eds, Advances in Kernel Methods-Support Vector Learning*. MIT Press, 1999.
- [109] L. Portinale and L. Saitta. Feature Selection. *in Deliverable, D14.1, IST Project MiningMart IST-11993*. 2002.
- [110] G. Ratsch, S. Mika, B. Schölkopf, and K.R. Muller. Constructing Boosting Algorithms from SVMs: an Application to One-Class Classification. *in PAMI*. vol. 24(9), pp. 1184-1199, 2002.
- [111] S. Raudys and R.P.W. Duin. On Expected Classification Error of the Fisher Linear Classifier with Pseudo-Inverse Covariance Matrix. *in Pattern Recognition Letters*. vol. 19(5-6), pp. 385-392, 1998.
- [112] S. Roweis and L. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *in Science*. vol. 290(5500), pp. 2323-2326, 2000.

- [113] B. Schölkopf, J. Platt, J. Shawe-Taylor, A.J. Smola, and R.C. Williamson. Estimating the Support of a High-Dimensional Distribution. *in Neural Computation*. vol. 13, 2001.
- [114] B. Schölkopf and A.J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.
- [115] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.
- [116] M. Skurichina and R.P.W. Duin. Regularization of Linear Classifiers by Adding Redundant Features. *in Pattern Analysis and Applications*. vol. 2(1), pp. 44-52, 1999.
- [117] M. Skurichina and R.P.W. Duin. Stabilizing Classifiers for Very Small Sample Size. *in Proceedings of International Conference on Pattern Recognition*. pp. 891-896, 1996.
- [118] C.W. Stanfill. Memory-Based Reasoning Applied to English Pronunciation. *in Proceedings of the Sixth National Conference on Artificial Intelligence*. pp. 577-581, Seattle, 1987.
- [119] E. Sung, Y. Zhu, and X. Li. Accelerating the SVM Learning for Very Large Data Sets. *in Proc. of International Conference on Pattern Recognition*. pp. 20-24 August 2006, Hong Kong.
- [120] D.L. Swets and J. Weng. Using Discriminant Eigenfeatures for Image Retrieval. *in IEEE Transaction on Pattern Analysis and Machine Intelligence*. vol. 18(8), pp. 831-836, 1996.
- [121] H.T. Tanaka, M. Ikeda, and H. Chiaki. Curvature-Based Face Surface Recognition Using Spherical Correlation. *Third IEEE International Conference on Automatic Face and Gesture Recognition*. pp. 372-377, 1998.
- [122] E.K. Tang, P.N. Suganthan, X. Yao, and A.K. Qin. Linear Dimensionality Reduction Using Relevance Weighted LDA. *in Pattern Recognition*. vol. 38(4), pp. 485-493, April 2005.
- [123] X. Tang and Z. Li. Video Based Face Recognition Using Multiple Classifiers. *in Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FGR04)*. pp. 345-349, 2004.
- [124] Q. Tao, G.W. Wu, and J. Wang. A New Maximum Margin Algorithm for One-Class Problems and Its Boosting Implementation. *in Pattern Recognition*. vol. 38(7), pp. 1071-1077, 2005.
- [125] W. Tian, M. Barbero, Z. Gu, and S. Lee. Image Classification by the Foley-Sammon Transform. *Opt. Eng.* vol. 25(7), pp. 834-840, 1986.

- [126] K.M. Ting. Discretization of Continuous-Valued Attributes and Instance-Based Learning. *Technical Report No. 491*. University of Sydney, Basser Department of Computer Science, 1994.
- [127] R. Tjahyadi, W. Liu, and S. Venkatesh. Application of the DCT Energy Histogram for Face Recognition. in *Proceedings of the 2nd International Conference on Information Technology for Application (ICITA 2004)*. pp. 305-310, 2004.
- [128] S. Tong and D. Koller. Support Vector Machine Active Learning with Applications to Text Classification. in *Journal of Machine Learning Research*. vol. 2, pp. 45-66, 2001.
- [129] V. Tresp. Scaling Kernel-Based Systems to Large Data Sets. in *Data Mining and Knowledge Discovery*. vol. 5(3), pp. 197-211, 2001.
- [130] I.W. Tsang, J.T. Kwok, and P.-M. Core Vector Machines: Fast SVM Training on Very Large Data Sets. in *Journal of Machine Learning Research*. vol. 6, pp. 363-392, 2005.
- [131] M. Turk and A. Pentland. Eigenfaces for Recognition. in *Journal of Cognitive Neuroscience*. vol. 3, pp. 72-86, 1991.
- [132] V. Vapnik. *Statistical Learning Theory*. Wiley, NY, 1998.
- [133] V. Vapnik and A. Chervonenkis. *Theory of Pattern Recognition (in Russian)*. Nauka, Moscow, 1974 (German Translation: W. Wapnik and A. Tscherwonkis, *Theorie der Zeichenerkennung*, AkademieVerlag, Berlin, 1979).
- [134] V.N. Vapnik. *Estimation of Dependences Based on Empirical Data*. New York, NY: Springer-Verlag, 1982.
- [135] H. Wang, S. Chen, and Z. Hu. Image Recognition Using Weighted Two-Dimensional Maximum Margin Criterion. in *Third International Conference on Natural Computation (ICNC 2007)*. pp. 582-586, 2007.
- [136] Y. Wang, T. Tan, and Y. Zhu. Face Verification Based on Singular Value Decomposition and Radial Basis Function Neural Network. in *Proc. 4th. Asian Conference on Computer Vision (ACCV)*. pp.432-436, Taipei, Taiwan, 2000.
- [137] S. Watanabe, P.F. Lambert, C.A. Kulikowski, J.L. Buxton, and R. Walker. Evaluation and Selection of Variables in Pattern Recognition. in *Computer and Information Sciences II, J. Tou (Ed.)*, New York: Academic Press, 1967.
- [138] J. Weston and C. Watkins. Support Vector Machines for Multiclass Pattern Recognition. in *Proceedings of the Seventh European Symposium on Artificial Neural Networks*. ed. M. Verleysen, Brussels, Belgium: D-Facto Public, pp. 219-224, 1999.

- [139] L. Wiskott, J. Fellous, N. Krüger, and C. v.d. Malsburg. Face Recognition by Elastic Bunch Graph Matching. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 19(7), pp. 775-779, 1997.
- [140] J. Yang, D. Zhang, A.F. Frangi, and J. Yang. Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2004.
- [141] M.-H. Yang. Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods. *Proc. of the Fifth Inter. Conf. on Automatic Face and Gesture Recognition (FG 2002)*. pp. 215-220, 2002.
- [142] J. Ye and Q. Li. A Two-Stage Linear Discriminant Analysis via QR-Decomposition. in *IEEE Transaction on Pattern Analysis and Machine Intelligence*. vol. 27(6), pp. 929-941, 2005.
- [143] H. Yu and J. Yang. A Direct Lda Algorithm for High-Dimensional Data with Application to Face Recognition. *Pattern Recognition*. vol. 34, pp. 2067-2070, 2001.
- [144] A.L. Yuille, D.S. Cohen, and P.W. Hallinan. Feature Extraction from Faces using Deformable Templates. in *International Journal of Computer Vision*. vol. 8, pp. 99-112, 1992.
- [145] D. Zhang. Two-Dimensional Bayesian Subspace Analysis for Face Recognition. in *Advances in Neural Networks*. vol. 4492, pp. 778-784, Springer Berlin/Heidelberg, 2007.
- [146] W. Zhao, R. Chellappa, and A. Krishnaswamy. Discriminant Analysis of Principal Components for Face Recognition. in *Proceedings of International Conference on Automatic Face and Gesture Recognition*. pp. 336-341, 1998.
- [147] W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld. Face Recognition: A Literature Survey. *ACM Computing Surveys*. vol. 35(4), pp. 399-459, 2003.
- [148] X.S. Zhou and T.S. Huang. Small Sample Learning during Multimedia Retrieval using BiasMap. in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Kauai, Hawaii, pp. 11-17, 2001.
- [149] Y. Zhu and E. Sung. From Spectral-Face to Eigen-Face. in *International Conference on Control, Automation, Robotics and Vision (ICCARV)*. Singapore, 2000.
- [150] Y. Zhu and E. Sung. The Spectral-Face Analysis for Face Recognition. in *The 5th Asian Conference on Computer Vision (ACCV)*. Australia, 2002.