3D Object and Human Face Recognition using Appearance Manifold with View-dependent Covariance Matrix

Lina
Acknowledgements

This dissertation is formally submitted for fulfilling partial requirements for the degree of Doctor of Science in Information Science from Nagoya University, and is personally used as a memento of my four years (2005-2009) journey in Nagoya, Japan. During my time at Nagoya University, many people have had a significant influence on me in a variety of ways, both academic and personal. To the following persons, I would like to express my sincere gratitude.

I would like to acknowledge in particular the continuous support and guidance of my academic advisor in the Doctoral course, Prof. Dr. Hiroshi Murase. He welcomed me to join his Lab and generously shared his expertise in recognition. Without his encouragement and constant guidance, I could not have finished my Doctoral program in time. I certainly could not have asked for a better Ph.D. advisor.

For taking the time to proofread and mark up most of my publications, including this dissertation, I am very grateful to my co-advisor, Associate Prof. Dr. Ichiro Ide. He has always been kind to make himself available even through his very heavy travel, research and teaching schedule. He also has provided a good balance of work and play, a great manager and a friendly advisor.

I would also like to thank Prof. Dr. Noboru Ohnishi, who consistently monitored my research progress and took effort in reading and providing valuable comments on the earlier version of this dissertation. Prof. Dr. Kazuya Takeda and Associate Prof. Dr. Kensaku Mori, who gave insightful comments, also deserve a special mention.

I am also grateful to Associate Prof. Dr. Tomokazu Takahashi, whom I was fortunate to work with. He has provided invaluable research assistance, collaboration, and friendship during my graduate study.

Special thanks also go to the other members of Murase-Ide Laboratory: the secretaries, Mrs. Fumiyo Kaba (and also her family) and Mrs. Hiromi Tanaka,
for processing almost all of my Japanese documents and for making my life run smoothly in Japan. I also thank Prof. Dr. Yoshito Mekada, Dr. Daisuke Deguchi, Hiroyuki Ishida, and other students for their interesting discussions. Although I cannot name all because of the limited space, for all of you who let me “borrow” your face, I owe you one!

I would also like to acknowledge the support from the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT) in the form of Monbukagakusho scholarship throughout my many years of study at Nagoya University: Research Student course (2005-2006) and Doctoral course (2006-2009).

I am also greatly indebted to my former academic advisor at the University of Indonesia, Prof. Dr. Benyamin Kusumoputro, who first brought me into the world of research and with whom I began to learn about recognition topics. Had I not met him, I would never have dreamed pursuing my Ph.D. degree in Japan.

I am also grateful to the staff members of the Faculty of Information Technology, Tarumanagara University, especially Prof. Dr. Dali S. Naga and Dr. Tony Mulia, for their recommendations and permissions to take a few years leave from teaching activities. Also to Susany Soplanit and Lely Hiryanto for their long-term friendships.

I would also like to gratefully acknowledge the supports of these special people: Doni H. Ramdhan, Wempi Saputra, Zamroni Salim, Lee Sung-Ho, Monica Racovita, Pierre Jonniaux, Babak Mehran, and Truong Trung Dung. Their caring, encouragements and friendships have helped me get through the difficult times during my graduate study and life in Japan.

My love and thanks go to my dear family - my mother, my sister, and her family - for their love and support during these 29 years. The memories of my father, who lived a short but condensed life, also provide a persistent inspiration for my journey in this life. To them I dedicate this thesis.

Finally, I thank God, who made all these possible.
Abstract

This thesis addresses the problem of recognizing 3D objects and human faces from still-images and video-sequences. Here, the recognition problem is formulated as an appearance matching process. In the appearance-based approach, an object is represented in the form of two-dimensional image sets. To gain efficiency, these images are then projected into a low dimensional space in which the images of each object are represented as Eigenpoints. However, since the appearance of an object in a two-dimensional image is influenced by several parameters, such as shape, pose, illumination, etc., it is important for the recognition system to capture these variations for gaining high recognition performance.

A novel method which performs generalization of Eigenpoints through feature lines, called the Modified Nearest Feature Line (MNFL) method is proposed. The feature lines can be acquired by corresponding every pair of Eigenpoints in the same class and projecting every Eigenpoint to the constructed feature lines. Since the feature lines virtually provide an infinite number of Eigenpoints in each class, it expands the capacity of the available database and increases the system’s ability to capture object variations.

Moreover, focusing on the problem of pose variability, an appearance manifold with View-dependent Covariance matrix (VC) method is proposed. Being different from the feature line scheme which forms a generalization of Eigenpoints regardless of their pose, the appearance manifold scheme constructs a continuous curve (appearance manifold) by linking two Eigenpoints of consecutive poses. The appearance manifold is constructed along with view-dependent covariance matrices, so that it could capture pose variability and also learn the samples’ distribution of each pose for gaining robustness to pose changes and also degradation effects.

Finally, a new incremental unsupervised-learning framework of appearance manifolds is also proposed to present more realistic recognition applications. It is obvious that it is difficult to collect large amounts of training images which depict
an object under all poses (from left sideview to right sideview). The incremental unsupervised-learning framework allows us to train the system with initial image-sequences, and later updates the existing categories incrementally every time an unlabeled image-sequence is input. The unlabeled images are first recognized based on the minimum distance of their projected Eigenpoints to the manifolds of objects and then the results are integrated to produce the final sequence’s decision.

The performance of the proposed methods were studied through experiments and the results showed that the proposed methods could accurately recognize 3D objects and human faces from still-images and video-sequences under a wide variety of poses, expressions, and degradation effects.
# Contents

Acknowledgements ........................................ i

Abstract ................................................................ iii

List of Tables .................................................. vii

List of Figures .................................................. viii

1 Introduction ................................................. 1
   1.1 Recognition by man and machine .................... 1
   1.2 Recognition issues .................................... 3
   1.3 Recognition approaches .............................. 4
      1.3.1 Geometric-based approach .................... 4
      1.3.2 Appearance-based approach ................... 6
      1.3.3 Hybrid approach ................................ 8
   1.4 Overview ............................................. 9

2 Recognition schemes in eigenspace ................. 11
   2.1 Eigenpoint scheme .................................. 11
   2.2 Feature line scheme ................................. 13
   2.3 Appearance manifold scheme ...................... 14

3 Face recognition using modified nearest feature line 16
   3.1 Introduction .......................................... 16
   3.2 Modified Nearest Feature Line method .......... 17
   3.3 Experiments and analysis ........................... 19
   3.4 Summary ............................................ 22
4 Object recognition using view-dependent covariance manifold 23
4.1 Introduction ............................................. 23
4.2 Manifold representation .................................. 26
4.3 Embedding covariance matrices in a manifold .......... 26
  4.3.1 Simple manifold .................................. 28
  4.3.2 Constant covariance manifold ...................... 28
  4.3.3 View-dependent covariance manifold ............... 30
4.4 Recognition of objects ................................... 34
4.5 Experiments ............................................. 35
  4.5.1 Performance in various degradation conditions ..... 38
  4.5.2 Performance in various training-pose intervals ..... 43
4.6 Discussion .............................................. 46
4.7 Summary ............................................... 48

5 Face recognition from video using incremental face manifold 49
5.1 Introduction ............................................. 49
5.2 Supervised-learning framework .......................... 53
  5.2.1 Supervised manifold construction ................... 53
  5.2.2 Classification of face-sequences .................... 56
5.3 Incremental unsupervised-learning framework .......... 58
  5.3.1 Incremental unsupervised manifold construction ... 58
  5.3.2 Classification and updating of face-sequences ...... 58
5.4 Experiments and analysis ................................ 61
  5.4.1 Performance in supervised-learning ................. 64
  5.4.2 Performance in incremental unsupervised-learning .... 67
5.5 Discussion .............................................. 71
5.6 Summary ............................................... 79

6 Conclusion .............................................. 80
6.1 Summary ............................................... 80
6.2 Contributions .......................................... 82
6.3 Future Work ........................................... 82

Bibliography ............................................ 84

Publications .......................................... 91
# List of Tables

1.1 Areas and applications of 3D object recognition. ........................................ 2
1.2 Areas and applications of human face recognition. ................................. 2

3.1 The datasets used in the experiments which contain various training and testing poses. ................................................................. 20
3.2 The recognition rates of 3D face recognition systems. ............................. 21

4.1 Various construction types of appearance manifold. ................................. 27
4.2 Results for recognizing objects of two datasets from images with various ranges of degradation effects. ................................................. 39

5.1 Accuracy rates of a 10 identity categories recognition task from video sequences with 8 sequence’s length in two conditions: small pose variation and severe pose variation in a supervised-learning framework (The reference accuracy values, obtained when using pose information given by a human, are presented in the brackets). ................................................. 66
5.2 Classification rates in recognizing faces from video sequences with 8 sequence’s length in an incremental unsupervised-learning framework (The reference accuracy values, obtained when using pose information given by a human, are presented in the brackets). ................................. 70
5.3 Classification rates of two experiments with the same initial settings but different testing conditions. ................................................................. 72
5.4 Accuracy rates of a 10 identity categories recognition task from video sequences with 30° training pose differences between frames (sparse training sequences). ................................................................. 78
List of Figures

1.1 Examples of images with various degradation effects. .................. 3
1.2 An example of a geometric-based approach which uses 33 facial
points (Courtesy of Z. Zhang et al. in “Comparison between geometry-
based and gabor-wavelets-based facial expression recognition using
multi-layer perceptron”, Proc. of 3rd IEEE Automatic Face and
Gesture Recognition (FG’98)). ............................................. 5
1.3 Examples of the eigenfaces which are constructed based on an appearance-
based approach (Courtesy of M. Turk in W. Zhao and R. Chellappa
(eds.), “Face Processing: Advanced Modeling and Methods”, Aca-
demic Press, 2005) ...................................................... 7
1.4 The proposed research scheme. ............................................. 9
2.1 The eigenspace construction and the image projection processes. . . 12
2.2 The constructions of feature lines of an object in an eigenspace. . . 14
2.3 The construction of an appearance manifold of an object in eigenspace. 15
3.1 The construction process of feature lines in the NFL method. ....... 17
3.2 The construction process of feature lines in the MNFL method. . . 18
3.3 The classification process of an unlabeled face image (eigenpoint)
to a feature line in the eigenspace. ...................................... 19
3.4 The examples of the face images used in the experiments with var-
ious poses and expressions. ............................................. 21
4.1 Scheme of an appearance manifold with embedded covariance matrix. 25
4.2 Appearance manifold construction using Simple Manifold (SM). . . 29
4.3 Appearance manifold construction using Constant Covariance Ma-
trix (CC). ............................................................... 29
4.4 Appearance manifold construction using View-dependent Covari-
ance Matrix by training-Point Interpolation (VCPI). . . . . . . . . . 31
4.5 Appearance manifold construction using View-dependent Covari-
ance Matrix by Eigenvector Interpolation (VCEI). . . . . . . . . . . 31
4.6 Interpolations of the eigenvectors and eigenvalues based on the high-
dimensional rotation theory. . . . . . . . . . . . . . . . . . . . . . . 33
4.7 Samples of objects used in the experiments. . . . . . . . . . . . . . 35
4.8 Samples of images with degradation effects. . . . . . . . . . . . . . 36
4.9 Results for recognizing objects in Dataset 1 from images with vari-
ous degradation effects. . . . . . . . . . . . . . . . . . . . . . . . . 41
4.10 Results for recognizing objects in Dataset 2 from images with vari-
ous degradation effects. . . . . . . . . . . . . . . . . . . . . . . . . 42
4.11 Results for recognizing objects in Dataset 1 from images with vari-
ous pose intervals. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 44
4.12 Results for recognizing objects in Dataset 2 from images with vari-
ous pose intervals. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 45
4.13 Verification results of various methods. . . . . . . . . . . . . . . . 47

5.1 Outline of the proposed video-based face recognition system. . . . 52
5.2 Construction process of a face manifold with view-dependent co-
variance matrices. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54
5.3 Interpolation of eigenvectors and eigenvalues in a feature space. . . 55
5.4 Classification and learning algorithm of an input sequence in an
incremental unsupervised-learning framework. . . . . . . . . . . . . . 59
5.5 The preprocessing step of an image (frame) of a video sequence. . . 61
5.6 Samples of face sequences. Dataset 1 was used for training, while
Dataset 2 and Dataset 3 were used for testing. . . . . . . . . . . . . . 63
5.7 Accuracy rates in recognizing faces from video sequences of Dataset 2
with various sequence’s lengths in the supervised-learning framework. 65
5.8 Accuracy rates in recognizing faces from video sequences with var-
ious lengths using various threshold values. . . . . . . . . . . . . . . 67
5.9 Classification rates in recognizing faces from video sequences of
Dataset 2 with various sequence’s lengths in an incremental unsupervised-
learning framework. . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
5.10 A rendered image of a manifold A in a 3D eigenspace, based on the input sequence A. .......................... 74
5.11 A rendered image of manifolds A and B in a 3D eigenspace, based on the previously input sequence A and a new input sequence B (a new person). ............................................. 74
5.12 A rendered image of manifolds A, B, and C in a 3D eigenspace, based on the previously input sequences A and B, and a new input sequence C (a new person). .......................... 75
5.13 A rendered image of manifolds A, B, and C+D in a 3D eigenspace, based on the previously input sequences A, B, and C, and a new input sequence D (a trained person with new poses). ............... 75
5.14 Visualization of face manifolds of a person using the proposed VC method, (a) obtained by the supervised-learning framework, (b–c) obtained by the incremental unsupervised-learning frameworks from different unlabelled input sequences. .................. 76
Chapter 1

Introduction

1.1 Recognition by man and machine

The human ability to recognize objects and faces is remarkable. Three dimensional objects, especially human faces, are not easily described by simple shapes or patterns; yet people have the ability to recognize common objects and familiar faces at a glance. While the human brain has its limitations in the total number of persons that it can accurately remember, a key advantage of a computer system is its capacity to handle a large number of face images. With a desire to mimic the human ability in recognizing objects and faces, researches have attempted to develop recognition systems which lead to the advent of computational approaches in pattern recognition.

Computer systems that can recognize objects or identify faces are useful in many applications. Table 1.1 and 1.2 lists some of the applications of 3D object and face recognition. 3D object recognition has a wide variety of industrial and robotics applications. Some of the representative applications in industry include product inspection and bin-picking, while in the area of robotics, automatic target (object) recognition for robot vision is highly developed.

For human face recognition, the ability to model a face or distinguish it from a large number of stored face models is essential for criminal identification and security systems. Moreover, the strong need for user-friendly systems that can secure our assets and protect our privacy without losing our identity is also obvious. In law enforcement field, face recognition system is also needed for criminal identification, surveillance, and suspect tracking. In the areas of image compression, in
Table 1.1: Areas and applications of 3D object recognition.

<table>
<thead>
<tr>
<th>Area</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry</td>
<td>Product (manufactured-parts) inspection, bin-picking, etc.</td>
</tr>
<tr>
<td>Robotics</td>
<td>Robot vision, Automatic target recognition.</td>
</tr>
</tbody>
</table>

Table 1.2: Areas and applications of human face recognition.

<table>
<thead>
<tr>
<th>Area</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td>Biometric personal identification: bank account, PC logon, etc.</td>
</tr>
<tr>
<td></td>
<td>Smart home/office: automatic doors, tv parental control, etc.</td>
</tr>
<tr>
<td>Law enforcement</td>
<td>Surveillance: CCTV control, shoplifting, etc.</td>
</tr>
<tr>
<td></td>
<td>Criminal identification and suspect tracking.</td>
</tr>
<tr>
<td>Image/video compression</td>
<td>Image transmitting, Video partitioning.</td>
</tr>
<tr>
<td>Robotics</td>
<td>Human-robot interaction.</td>
</tr>
<tr>
<td>Computer graphics</td>
<td>Virtual reality and video animation.</td>
</tr>
</tbody>
</table>

partitioning scenes from a television show, more scenes containing people should be captured than other objects, since the audience is much more likely to care about the detail of the human actors than of complementary objects. In the area of robotics, creating “human-computer interaction” is necessary. It is important for robots to understand, communicate with, and react to humans in natural ways, such as understanding gestures, reading lips to facilitate speech recognition, and visually identifying individuals or objects. In computer graphics fields, many useful applications can be created, such as the development of an interactive graphic system for face modeling or video animation.

Motivated by such diverse applications and materials, object and face recognition have become a popular topic. Over the past 30 years extensive researches have been conducted on various aspects of object and face recognition by humans and machines. A concise review of some successfully developed recognition works is presented in Section 1.3.
1.2 Recognition issues

A general problem of machine recognition of objects and faces can be formulated as follows: given a still-image or a continuous image-sequence in video of a scene, identify or verify one or more objects/persons in the scene using a stored object/face database. Between these two types of recognitions, significant differences exist, such as the image quality, the background situation, the variability of the images of a particular individual that must be recognized, and the nature, amount, and order of input. In a controlled environment, the image acquisition and segmentation processes are relatively easy to conduct. However, if a static picture (still-image) contains a wide scene, automatic location and segmentation of an object or a face could pose serious challenges. On the other hand, if a video sequence is available, segmentation of a moving object or person can be more easily accomplished using motion as a cue. However, the small size and low image quality of faces captured from video can significantly increase the difficulty in recognition. Fig. 1.1 shows the examples of images with various degradation effects, such as incomplete region and low quality.

Though many existing systems build-in some sort of performance invariance by applying preprocessing methods such as histogram equalization or pose learning, a significant pose change can cause serious performance degradation to a recognition system. In addition, image recognition can also be affected by its capturing condition and the segmentation process. For example, low quality (blur) images might be a result of motion changes, while inaccurate segmentation processes can produce shifted images, rotated images, or even images with incomplete regions.

In this thesis, two main recognition issues are addressed: (1) recognition of 3D objects and human faces with various pose changes and (2) recognition of 3D objects/human faces from images which are influenced with various degradation effects. The recognition materials also consist of still-images and image-sequences in video.
1.3 Recognition approaches

The literatures on object and face recognition is vast and diverse. Many types of recognition techniques have been developed to fulfill the requirements of each specific task. As a consequence, a single system often involves a mixture of techniques of several different recognition principles. The usage of these mixed techniques makes it difficult to classify these systems based purely on what types of techniques they use for feature representation or classification.

Based on the psychological study of how humans use holistic and local features, there are three main categories of recognition approaches: (1) Geometric-based, (2) Appearance-based, and (3) Hybrid. The following subsections present a brief review of each approach and list their prominent works.

1.3.1 Geometric-based approach

Typically, in a geometric-based approach, local features are first extracted and their locations are input into a structural classifier. For object recognition in general, the most common approach is to extract features from objects, build some sort of a model from these features, and perform recognition by matching feature sets. Since the geometrical relationships among features are stable under varying conditions, the geometric-based approach is much more invariant (i.e. to pose and illumination changes). Thus, it can give a significant advantage if the geometric boundary descriptions are reliably calculated. Another advantage of this geometric-based approach is that it highly supports subparts recognition (e.g. number plate of a vehicle, leaf of a tree, etc.) due to its ability to control the geometrical arrangements of object’s features. On the other hand, the necessity of controlling geometrical relationships among features and the process of building 3D models from these features make this geometric-based approach much more expensive to compute than other approaches.

For face recognition, attempts to automate human face by computers began in the early 1970s, where most of the methods use structural matching, such as calculating the width of the head, the distances between the eyes and from the eyes to the mouth, etc. [1], or the distances and angles between eye corners, mouth extrema, nostrils, and chin top [2]. Kelly [1] developed heuristic goal-directed methods to measure distances in standardized images of the body and head based
on edge information, while Kanade [2], whose face identification system was the first automated system, uses a top-down control strategy directed by a generic model of expected feature characteristics of a face. Kanade’s system calculated a set of facial parameters from a single face image, comprised of normalized distances, areas, and angles between fiducial points. Thus, to match the face to one of a known set, a pure statistical approach which depends primarily on local histogram analysis and absolute gray-scale values is used.

An interactive face recognition system which is based on a face vector with 21 features (e.g. shade of hair, length of ears, lip thickness) were then proposed by Harmon and Goldstein [3]. Moreover, Harmon et al. [3] recognized face profile silhouettes by automatically choosing fiducial points to construct a 17-dimensional feature vector for recognition, while Gordon [4] also investigated face recognition using side-view facial profiles. Fig. 1.2 shows an example of a geometric-based approach which uses 33 facial points. Others have also developed automated face recognition by characterizing and recognizing face by a set of geometric parameters (e.g. [5], [6], [7], [8], [9], [10]).
Yuille et al. [11] and others have used deformable templates and parameterized models of features with given spatial relations. A Hidden Markov Model (HMM)-based method which utilizes strips of pixels that cover the forehead, eye, nose, mouth, and chin without finding the exact locations of facial features is proposed in [12] and [13]. The Elastic Bunch Graph Matching (EBGM) [14][15], which is based on the Dynamic Link Architecture (DLA) [16][17], has been recorded as one of the most successful method of the geometric-based approach. However, recently neural network techniques have also been used (e.g. [18], [19]) and attempted to move away from pure geometric-based approach.

1.3.2 Appearance-based approach

It is clear that a recognition system should be able to visually memorize an object by its appearance in order to mimic the human behaviour. Therefore, in the appearance-based approach, the recognition problem is formulated as one that matches appearance rather than shape. The advantages of this appearance-based approach are: (1) easy to process, since it uses the whole face region as the raw input to a recognition system, and (2) less expensive, since it can be accomplished without constructing a 3D model. Moreover, it is argued that a 3D recognition system can be accomplished using linear combinations of as few as four or five 2D viewpoint images [20][21].

Combined with the eigenspace concept which projects image sets into a low-dimensional space, called the eigenspace, the appearance-based method can produce a beneficial real-time recognition system. Starting from the successful low-dimensional reconstruction of faces using PCA projections [22][23], eigenpictures have been one of the major driving forces behind face representation, detection, and recognition. However, Turk and Pentland [24] were the first researchers who proposed the concept of using PCA for recognition by their “eigenface”. Fig. 1.3 shows the examples of the eigenfaces which are constructed based on an appearance-based approach.

Later on, several PCA-based works on face-parts recognition were introduced, such as eyes, nose, mouth [24][25][26][27], ears [28], expressions [24][29], etc. The modular eigenface method [25] also uses hybrid features by combining eigenfaces and other eigenmodules, such as eigeneyes, eigenmouth, and eigennose. The expectation was that the training set would contain enough variation so that it would
be modeled in the eigenfaces. However, basically the original eigenfaces framework does not explicitly account for variations in pose, lighting, scale, facial expressions, or any dynamic facial features of an individual. Thus, these variations may cause the eigenface method to fail.

Subsequent works then have improved the basic eigenface method especially in handling pose variations. A notable work of Murase and Nayar [30][31] utilized an eigenspace approach to represent and recognize general 3D objects at various poses, formulating object and pose recognition as parameterized appearance matching. Nayar [32] also recognized and determined the pose of 100 objects in real-time by creating appearance manifolds based on the learned eigenspace. Around the same time, Moghaddam et al. [33] developed a probabilistic matching algorithm that uses a Bayesian approach to separately model both interclass and intraclass distributions. This improves on the implicit assumption that the images of all
individuals have a similar distribution. Penev and Sirovich [34] then investigated the dimensionality of face space, concluding that for very large databases, at least 200 eigenfaces are needed to sufficiently capture global variations such as pose, lighting, scale, etc.

Other appearance-based recognition uses a feature line which replace the point-to-point distance with the distance between a point and a feature line linking two stored sample points [35], Fisherfaces [36][37][38][39] which use the Linear/Fisher Discriminant Analysis (LDA/FDA), SVM methods which use a support vector machine as the classifier [40], and Independent Component Analysis (ICA) which is argued to have more representative power than PCA, and hence provides better recognition performance than PCA [41].

Recent results in appearance-based recognition applied to face recognition and other tasks include more sophisticated learning methods (e.g. [42]), warping and morphing face images [43][44] to accommodate a wider range of face poses, including previously unseen poses, explicitly dealing with issues of robustness [45][46][7], better methods of modeling interclass and intraclass variations [36], and unsupervised learning which allow the training model being updated automatically at different times [48].

1.3.3 Hybrid approach

Just as the human perception system uses both local features and the whole face region to recognize a face, a machine recognition system should also use both. Thus, it is believed that this approach could potentially offer the best of the two types of approaches.

In the hybrid method category, Craw et al. [49] were the first to combine processing face shape with eigenface-based recognition. Craw normalized the face images geometrically based on 34 face landmarks in an attempt to isolate the image intensity from geometric factors. Von der Malsburg et al. [17][50] introduced several systems based on elastic graph matching, which utilizes a hybrid approach where local grayscale information is combined with the global feature structure. Cootes and Taylor [51] presented a unified approach that combines local and global information, using flexible shape models to explicitly model both shape and intensity.

Other successful systems in this category include combination with neural net-
network learning, such as Probabilistic Decision Based Neural Network (PDBNN) [52] and Convolutional Neural Network (CNN) [53], and also the Evolution Pursuit (EP) [37] methods.

1.4 Overview

This thesis focuses towards developing an appearance-based recognition system that does not require knowledge on three-dimensional information or detailed geometry of the target objects. The goal is to develop 3D object and face recognition methods which are reasonably simple and accurate, especially in in-door environments (e.g. in a laboratory, office, etc). Moreover, the proposed methods are also expected to deal with image variations, such as pose changes, low quality (blurred) images, shifted images, rotated images, and also images with incomplete regions.

The proposed scheme is based on the appearance based approach. The main benefit of using the appearance based approach is that it is less expensive, since it can be accomplished without constructing a 3D model. The recognition materials might be still-images or image-sequences in video. Recognition of still-images is performed by projecting a new image into the eigenspace and then classifying the object/face by comparing its distance with the locations of samples in an eigenspace. The same recognition process is applied for recognition of image-sequences in a video with an integration process of frame-decisions. An incremental unsupervised-learning framework that allows updating of the training category, is also introduced in this thesis. The proposed method has several advantages over

Chapter 3
Still-image based recognition:
Modified nearest feature line (MNFL)

Chapter 4
Still-image based recognition:
Appearance manifold with View-dependent Covariance matrix (VC)

Chapter 5
Video based recognition:
View-dependent manifold with incremental unsupervised learning

Figure 1.4: The proposed research scheme.
other recognition schemes in its accuracy and learning capacity for recognizing 3D objects/human faces with pose variations and from images which are influenced with various degradation effects.

The proposed research scheme is depicted in Fig. 1.4, while the structure of this thesis is organized as follows. Chapter 2 describes the recognition schemes in eigenspace along with their relevant literature on feature extraction and recognition procedures. Chapter 3 introduces the Modified Nearest Feature Line (MNFL) method for face recognition from still-images. Chapters 4 and 5 present the appearance manifold with View-dependent Covariance matrix (VC) method for recognizing 3D objects from still-images and recognizing human faces from video, respectively. In Chapter 5, a new incremental unsupervised-learning framework of appearance manifolds is also introduced to present more realistic recognition applications. Each chapter also describes the proposed method in detail, shows the experimental results and analysis, and presents the discussion and summary. Finally, Chapter 6 summarizes the main ideas and the contributions of this thesis, and also discusses the future research directions.
Chapter 2

Recognition schemes in eigenspace

2.1 Eigenpoint scheme

The eigenpoint scheme is known as the basic recognition scheme in eigenspace. It involves two main parts: (1) construction of an eigenspace and (2) recognition based on the sample point (eigenpoint) position. The eigenspace is constructed from an initial set of training images by applying the Principal Component Analysis (PCA). PCA (also known as Karhunen–Loeve transformation [54][55]) produces a set of eigenvectors along with their corresponding eigenvalues. However, only the eigenvectors which correspond to the $k$ largest eigenvalues are kept and used to define the eigenspace. After constructing the eigenspace, each image in the training set is normalized with the average image and is projected into the eigenspace as a low dimensional feature vector, called the eigenpoint. Fig. 2.1 shows the eigenspace construction and the image projection processes.

In the second part, the recognition process comprises an online procedure. When a new image is input to the system, the average image is subtracted and the result is projected into the eigenspace. This produces an input-eigenpoint which represents an input image in the eigenspace. Next, a distance metric (e.g. the Euclidean distance) is used to calculate the distances between the input-eigenpoint and the training-eigenpoints. The input-eigenpoint is then recognized as the corresponding class in which the eigenpoint has the closest distance.

The mathematical descriptions of the eigenspace construction and the linear
projection of the training images into the eigenspace are presented as follows. First, the captured images should be normalized in brightness and scaled in order to be invariant to image magnification and illumination intensity. These normalized images can be written as a vector $x$ by reading the number of pixels ($N$) in an image:

$$
x = [x_1, x_2, ..., x_N]^T \tag{2.1}
$$

where $T$ is the transpose operator.

Let $M$ be the number of the images in a training set. By subtracting the average image $c$ of all images, the training set $Y$ will be obtained by

$$
Y = [x_1 - c, x_2 - c, ..., x_M - c]. \tag{2.2}
$$

Next, the auto-correlation matrix is defined by

$$
Q = YY^T \tag{2.3}
$$

and the eigenvalues $\lambda_i$ with its corresponding eigenvectors $e_i$ are determined by solving the following eigenvector decomposition problem:

$$
Qe_i = \lambda_i e_i. \tag{2.4}
$$
In order to obtain the dimension reduction, simply ignore small eigenvalues and use only \( k \) corresponding eigenvectors with a threshold value \( \alpha \):

\[
\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{N} \lambda_i} \leq \alpha
\]

where \( k \ll N \).

The first \( k \) eigenvectors will be used to project \( M \) training samples with \( \theta \) pose of \( P \) objects onto the eigenspace. \( g_{m}^{(p)}(\theta) \) is a projected point (eigenpoint) of an image \( x_{m}^{(p)}(\theta) \) in the eigenspace and is calculated through:

\[
g_{m}^{(p)}(\theta) = [e_1, e_2, \cdots, e_k]^T (x_{m}^{(p)}(\theta) - c)
\]

By projecting all the training samples onto the eigenspace, the training features are represented efficiently as a set of discrete points in a low dimensional space.

It was Turk et al. \[24\] who first used this eigenpoint scheme to develop a face recognition application.

2.2 Feature line scheme

Despite its advantages, the basic eigenpoint scheme also has a shortcome in capturing appearance variations. For example, it cannot represent significant variations in pose, scale, orientation, translation, and lighting. As a simple yet effective idea, Li et al. \[35\] proposed to generalize the representational capacity of the available samples by using feature lines.

The feature line scheme assumes that at least two samples are available for each class and attempts to generalize these samples by using linear interpolation and extrapolation between samples in an eigenspace \[35\][56]. More specifically, the two samples (eigenpoints) are generalized by a feature line which is the line passing through the two eigenpoints. Here, the derived feature line can capture more variations of object appearances (i.e. pose, illumination, etc.) than the original points. Thus, it expands the capacity of the available database since it virtually provides an infinite number of eigenpoints of the class. The nearest distance between an unlabeled sample and the feature lines is then calculated for classification. The constructions of feature lines of an object in an eigenspace is shown in Fig. 2.2.
The feature line scheme has been used for several applications, such as face recognition [35], image classification [56], and speaker identification [57]. It was reported that the Nearest Feature Line (NFL) method [35] makes better use of the ensemble information for decision-making and outperforms the conventional eigenpoint scheme (i.e. Nearest Neighbor method).

2.3 Appearance manifold scheme

Being triggered by the motivation to deal with pose variability of the object appearance, Murase and Nayar [31] developed a continuous and compact representation of object appearance that is parameterized by the variables (i.e. pose, illumination, etc.). This new representation is referred to as the appearance manifold.

Addressing the same problem of capturing various appearance changes, while the previous feature line scheme forms a generalization of eigenpoints by corresponding every two eigenpoints that represent the same object regardless of their poses, the appearance manifold scheme constructs a continuous curve (appearance manifold) by linking two eigenpoints of consecutive poses. Li et al. [35] states that the feature line can be considered as a simpler version of the spline type appear-
Figure 2.3: The construction of an appearance manifold of an object in eigenspace.

There are several works that were reported to use the appearance manifold scheme for various applications, such as the Parametric Eigenspace method [30][32] for object recognition and the Probabilistic Eigenspace method [33][48] for face recognition.
Chapter 3

Face recognition using modified nearest feature line

3.1 Introduction

Research and development of a 3D face recognition system has grown fast along with the increasing demand of reliable security systems, especially automatic user identification systems. Classical techniques have certain drawbacks such as passwords forgotten or compromised, cards lost or stolen, and the system not able to make difference between a valid client and an impostor [58]. Therefore, methods which recognize users by their physiological features such as finger-print, iris, voice, and face are preferred.

Nowadays, many successful face recognition systems have been developed, however those works mainly focussed in recognizing images with frontal or semi-frontal poses. In this thesis, a 3D face recognition system which is based on the appearance-based approach is developed. The system recognizes human faces by comparing an input image with models that already exist in a database gallery. The proposed system uses the PCA technique to define a feature space (eigenspace) and the Modified Nearest Feature Line (MNFL) for recognition. The MNFL method is a recognition method which follows the feature line approach introduced in Section 2.2. The main advantage of the MNFL method is in the use of feature lines which allow the system to capture more feature variations than that of the feature points, so that it is expected to perform more accurate recognition of human faces. The description of the MNFL method, experimental results, and
summary are presented in the following sections of this chapter.

### 3.2 Modified Nearest Feature Line method

The Modified Nearest Feature Line (MNFL) is a recognition method which is based on the feature line recognition technique. Here, the feature line is constructed in order to capture object variations, such as pose, illumination, etc. The MNFL method modifies the basic Nearest Feature Line (NFL) method proposed by Li et al. [35].

As a brief review of Li’s NFL method [35], the illustration of the construction process of feature lines in the NFL method is shown in Fig. 3.1. The main idea of the feature line scheme can be described mathematically as follows. Suppose there are \( L \) training samples of each \( P \) different classes which have been projected into an eigenspace. Define feature lines \( g_i g_j \), which are straight lines passing through samples (eigenpoints) \( g_i \) and \( g_j \) of a same class. Any two eigenpoints in the same class are generalized by the feature line passing through them and all the feature lines in the same class constitute the feature line space of that class. In the case of Fig. 3.1, there are three eigenpoints of a same class, namely \( g_1 \), \( g_2 \), and \( g_3 \) in the eigenspace, the feature lines that can be constructed in the NFL method are \( g_1 g_2 \), \( g_2 g_3 \), and \( g_3 g_1 \). In general, the number of feature lines of each class \( nol_{NFL} \)
that can be constructed from $L$ Eigenpoints in the NFL method can be calculated by:

$$nol_{NFL} = \frac{L(L - 1)}{2}$$  \hspace{1cm} (3.1)

Meanwhile, in the proposed MNFL method, more feature lines can be constructed by projecting every eigenpoint to all available feature lines, as illustrated in Fig. 3.2. These additional feature lines are constructed without any additional eigenpoints in the eigenspace. First, all available feature points $g_1$, $g_2$, and $g_3$ are connected in order to obtain some basic feature lines $g_1g_2$, $g_2g_3$, and $g_3g_1$. Then, the feature points are projected to the basic feature lines to obtain perpendicular lines which are defined as additional feature lines $g_1 \perp g_2g_3$, $g_2 \perp g_3g_1$, and $g_3 \perp g_1g_2$. $g_1 \perp g_2g_3$ represents a feature line which is constructed by projecting perpendicularly feature point $g_1$ to the feature line $g_1g_2$, $g_2 \perp g_3g_1$ is a feature line which is obtained by projecting feature point $g_2$ to the feature line $g_3g_1$ perpendicularly, and so on. Therefore, the total number of feature lines of each class of the MNFL method $nol_{MNFL}$ that can be constructed from $L$ eigenpoints are:

$$nol_{MNFL} = \frac{L(L - 1)^2}{2}$$  \hspace{1cm} (3.2)

The classification process of an unlabeled image $z$ to the feature line $g_1g_2$ in the eigenspace is depicted in Fig. 3.3. First, $z$ is projected to each feature line as
point $b_{ij}^{(p)}$ in perpendicular position using:

$$b_{ij}^{(p)} = g_{i}^{(p)} + \eta (g_{j}^{(p)} - g_{i}^{(p)}) \tag{3.3}$$

with $\eta$ is a position-parameter of the projection point $b_{ij}^{(p)}$ to $g_{i}^{(p)}$. The parameter $\eta$ can be calculated by:

$$\eta = \frac{(z - g_{i}^{(p)})^T (g_{j}^{(p)} - g_{i}^{(p)})}{(g_{j}^{(p)} - g_{i}^{(p)})^T (g_{j}^{(p)} - g_{i}^{(p)})} \tag{3.4}$$

The distance between an unlabeled image $z$ and a feature line is then defined by:

$$d(z, \overline{g_{i}^{(p)}} \overline{g_{j}^{(p)}}) = \| z - b_{ij}^{(p)} \| \tag{3.5}$$

Finally, an unlabeled image $z$ is classified as the same category as the feature line which has the minimum distance.

\section*{3.3 Experiments and analysis}

In order to evaluate the recognition performance of the proposed MNFL method, a face recognition system has been developed to recognize human faces with different poses and expressions. The developed system recognized images of eight persons with four different expressions (normal, smile, angry, and laugh), which are taken from different viewpoints. Table 3.1 shows three datasets that were used in the experiments.
Table 3.1: The datasets used in the experiments which contain various training and testing poses.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training pose (°)</th>
<th>Total train/category (images)</th>
<th>Testing pose (°)</th>
<th>Total test/category (images)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-90, -30, 30, 90</td>
<td>4 poses x 4 expressions = 16</td>
<td>-75, -60, -45, -15, 0, 15, 45, 60, 75</td>
<td>9 poses x 4 expressions = 36</td>
</tr>
<tr>
<td>2</td>
<td>-90, -45, 0, 45, 90</td>
<td>5 poses x 4 expressions = 20</td>
<td>-75, -60, -30, -15, 15, 30, 60, 75</td>
<td>8 poses x 4 expressions = 32</td>
</tr>
<tr>
<td>3</td>
<td>-90, -60, -30, 0, 30, 60, 90</td>
<td>7 poses x 4 expressions = 28</td>
<td>-75, -45, -15, 15, 45, 75</td>
<td>6 poses x 4 expressions = 24</td>
</tr>
</tbody>
</table>
Figure 3.4: The examples of the face images used in the experiments with various poses and expressions.

The examples of face images are shown in Fig. 3.4, while the recognition rates of the MFNL and the NFL methods for recognizing faces under various poses and expressions are presented in Table 3.2. For the dataset with sparse training pose (60° interval), the recognition rate for Dataset 1 was 49.76% for the NFL method, while the proposed MNFL method gave higher recognition accuracy with 57.45%. For datasets with denser training poses (45° and 30°), the recognition rates increased. For Dataset 2 with 45° training pose interval, 64.90% and 85.33% recognition accuracies were achieved for the NFL and the MNFL methods, re-

Table 3.2: The recognition rates of 3D face recognition systems.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset 1 (%)</th>
<th>Dataset 2 (%)</th>
<th>Dataset 3 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nearest Feature Line (NFL)</td>
<td>49.76</td>
<td>64.90</td>
<td>71.88</td>
</tr>
<tr>
<td>Modified Nearest Feature Line (MNFL)</td>
<td>57.45</td>
<td>85.33</td>
<td>95.67</td>
</tr>
</tbody>
</table>
spectively. Finally, the highest recognition accuracy for the MNFL method was 95.67%, while 71.88% recognition accuracy was achieved by the NFL method for Dataset 3 with 30° training pose interval. The results prove the theorem that the MNFL method with more feature lines can give a better recognition accuracy than the NFL method.

3.4 Summary

The 3D face recognition system using the MNFL method has been presented in this chapter. Compared with the NFL method, the proposed MNFL method increases the number of constructed feature lines without increasing the number of eigenpoints. Therefore, the MNFL method captures more face variations through the additional feature lines, especially pose and expression changes. Experimental results showed that the proposed MNFL method gives higher recognition accuracies compared to that of the conventional NFL method for various training pose intervals. It was also confirmed that the developed face recognition system using the MNFL method could recognize face images with various expressions.
Chapter 4

Object recognition using view-dependent covariance manifold

4.1 Introduction

The appearance-based framework combined with the eigenspace concept has been widely used in many recognition tasks, such as object recognition and face recognition. Some of the earlier works in this domain include the application of characterizing the human face using PCA technique in eigenpictures by Kirby and Sirovich [22][23] and eigenfaces by Turk and Pentland [24]. Later, Wiskott et al. [15] pointed out a major disadvantage of PCA, that it could not capture even the simplest invariance unless this information is explicitly provided in the training data. Since in a non-controlled environment, some variances of pose, illumination, occlusion, shifting, rotation, and so on, might occur and change the appearance of objects in captured images, the eigenpoint method tends to fail when there are significant variations.

Addressing the above problems, researchers developed the appearance manifold scheme in the eigenspace. For years, various types of appearance manifolds have been proposed, such as the simple appearance manifold in [31] which could handle pose and illumination variations, the appearance manifold with probabilistic techniques in [33], [59], [60], [61], [62] for handling various facial changes, the layer-transparent manifold in [63] for recognizing occluded objects, and other types
of appearance manifolds which address different problems. First, we focus on the work of Murase and Nayar, called the Parametric Eigenspace method [31], due to its simplicity and applicability to pose variation problems in general. However, although it can deal with pose and illumination changes, this model only works well with the assumption that there are no degradation effects. Unfortunately, this assumption is not realistic in real-world applications. In an image capturing process or segmentation process, some degradation effects usually occur and influence the original image. When some significant variations exist, the positions of non-degraded images and the images which are influenced with some degradation effects might be placed far from each other in an eigenspace. Thus, making the learning process rely on a simple manifold to capture image variations is not sufficient.

To overcome these limitations, in this thesis, a novel method to construct the appearance manifold with embedded view-dependent covariance matrix is proposed. Here, the covariance matrix is used to learn the samples distribution of each pose for gaining noise-invariance. However, since the appearance of an object in the captured image will be different for every different pose, the covariance matrix value will also change. Thus, the idea is to embed a view-dependent covariance matrix in an appearance manifold in order to accurately capture the distribution information.

In the proposed view-dependent covariance matrix method, the mean vectors and covariance matrices are analyzed and have different values for each training pose. Further, two view-dependent covariance matrix methods are proposed: (1) View-dependent Covariance matrix by training-Point Interpolation (VCPI) and (2) View-dependent Covariance matrix by Eigenvector Interpolation (VCEI). In the VCPI method, to cover the untrained poses, the view-dependent appearance manifold is constructed by interpolating every training-point from one pose to its consecutive poses. While in the VCEI method, it is not necessary to critically control the correspondences between every training image of each pose to a consecutive pose, so only the eigenvectors and eigenvalues of two consecutive poses are interpolated. Thus, besides its noise-invariant characteristic, the VCEI method is also efficient.
Figure 4.1: Scheme of an appearance manifold with embedded covariance matrix.
4.2 Manifold representation

Appearance-based approaches use sets of training images in various poses. These images are usually represented in a very high dimensional space, thus, processing them directly in the image space are computationally expensive. To efficiently process these images, a projection technique using PCA method which transforms a collection of images into eigenpoints in the eigenspace domain is necessary. In this thesis, the eigenpoint projection steps have been described in Section 2.1.

4.3 Embedding covariance matrices in a manifold

After projecting all the training samples onto the eigenspace, training features are represented efficiently as a set of eigenpoints in a low dimensional space. In this thesis, the idea to overcome the problem of recognizing objects from images which are influenced with degradation effects is conducted by taking into account the correlation of the data sets for gaining noise invariant in the appearance manifold.

Fig. 4.1 shows the scheme of the construction process of an appearance manifold with embedded covariance matrix. In order to construct the appearance manifold with embedded covariance matrix, first, the system is trained with images for each training pose. Note that for each pose, only a single camera-captured image is necessary, while the other images are generated by adding artificial noises to the original camera-captured image. All the training samples are then projected onto the eigenspace and the mean vector and the covariance matrix are calculated to represent the center position of samples and sample-distribution for every training pose. Finally, to cover the unlearned poses, the appearance manifold is constructed by interpolating the mean vector and covariance matrix of one pose to its consecutive poses. As it embeds the covariance matrix in the appearance manifold, the appearance manifold takes into account the correlations of the data set. Thus, it will be noise invariant.

This section presents various models of appearance manifold with embedded covariance matrix, describes the idea and explains the process of each construction type, and presents the classification technique for the recognition step. Various construction types of appearance manifold are shown in Table 4.1.
Table 4.1: Various construction types of appearance manifold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean vector ($\mu$)</th>
<th>Covariance Matrix ($\Sigma$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Manifold (SM)</td>
<td>Linear interpolation</td>
<td>Identity matrix</td>
</tr>
<tr>
<td>Constant Covariance (CC)</td>
<td></td>
<td>Average covariance matrix</td>
</tr>
<tr>
<td>View-dependent Covariance matrix by training-Point Interpolation (VCPI)</td>
<td></td>
<td>Interpolation of training points</td>
</tr>
<tr>
<td>View-dependent Covariance matrix by Eigenvector Interpolation (VCEI)</td>
<td></td>
<td>Interpolation of eigenvectors and eigenvalues</td>
</tr>
</tbody>
</table>
4.3.1 Simple manifold

The Simple Manifold (SM) method constructs the manifold based on interpolation of mean vectors of samples, and applies an identity matrix for the covariance matrix for each pose. Moreover, the SM model basically relies only on the mean vectors, without considering the information of sample distributions. In case of using only one image sample for each pose, this method will be equivalent to Murase and Nayar’s Parametric Eigenspace method [30][31].

In the SM method, after projecting all training images onto the eigenspace, the mean vector \( \mu^{(p)}(\theta) \) and the covariance matrix \( \Sigma^{(p)}(\theta) \) for each object \( p \) of pose \( \theta \) are calculated. The mean vector is typically estimated by:

\[
\mu^{(p)}(\theta) = \frac{1}{L} \sum_{l=1}^{L} g^{(p)}_l(\theta) \tag{4.1}
\]

where \( L \) is the number of training samples from each class and \( g^{(p)}_l(\theta) \) is the image sample \( l \) of object \( p \) with pose \( \theta \) in the eigenspace. Meanwhile, the value of identity matrix for its covariance matrix is given by:

\[
\Sigma^{(p)}(\theta) = I \tag{4.2}
\]

Finally, to construct the appearance manifold, any interpolation method to the mean vector of two-consecutive poses may be used, along with an identity matrix value as the covariance matrix for each training pose. The construction model of an appearance manifold using the Simple Manifold (SM) method is shown in Fig. 4.2.

4.3.2 Constant covariance manifold

The construction model of the Constant Covariance matrix (CC) method is depicted in Fig. 4.3. Here, the appearance manifold is constructed by interpolating mean vectors and applying the same (average) value to all covariance matrices for every pose. Thus, the CC method has a covariance matrix with constant values for every pose in the manifold.

In the CC method, the mean vector is calculated as in Eq. 4.1 and then an interpolation method is used to obtain mean vectors for the untrained poses. Meanwhile, the covariance matrix for each training pose is calculated and then its average value is applied to every covariance matrix in the manifold. The covariance
Figure 4.2: Appearance manifold construction using Simple Manifold (SM).

Figure 4.3: Appearance manifold construction using Constant Covariance Matrix (CC).
matrix for each pose is typically estimated by:

\[
\Sigma^{(p)}(\theta) = \frac{1}{L} \sum_{l=1}^{L} (g_{l}^{(p)}(\theta) - \mu^{(p)}(\theta))(g_{l}^{(p)}(\theta) - \mu^{(p)}(\theta))^T
\]  

(4.3)

and the average value of covariance matrices for all \(W\) poses is calculated by:

\[
\bar{\Sigma}^{(p)} = \frac{1}{W} \sum_{w=1}^{W} \Sigma^{(p)}(\theta)
\]  

(4.4)

Finally, the average value of covariance matrix is applied to every covariance matrix for each pose in the manifold by:

\[
\Sigma^{(p)}(\theta) = \bar{\Sigma}^{(p)}
\]  

(4.5)

### 4.3.3 View-dependent covariance manifold

In order to capture the distribution information of each pose more accurately, I propose a manifold model which has a different mean vector and a different covariance matrix for each pose. There are two construction methods of appearance manifold with the view-dependent covariance matrix: (1) View-dependent Covariance matrix by training-Point Interpolation (VCPI) and (2) View-dependent Covariance matrix by Eigenvector Interpolation (VCEI).

In the VCPI method, the appearance manifold is obtained by interpolating every training-point in each pose to the training-points in a consecutive pose that has the same characteristics, such as same degradation effects. The equation for linearly interpolating two training-points from two consecutive training-poses \(g^{(p)}(\theta)\) and \(g^{(p)}(\theta+\text{interval})\) is given by:

\[
g^{(p)}(\theta + \zeta) = (1 - \zeta)g^{(p)}(\theta) + (\zeta)g^{(p)}(\theta+\text{interval})
\]  

(4.6)

where \(\zeta\) is the fractional part which indicates how far the pose value \(\theta\) changes.

After the interpolation process, the mean vector and covariance matrix for each pose can be calculated using Eq. 4.1 and Eq. 4.3, respectively. Here, the appearance manifold will have different values of mean vector and covariance matrix for each pose. Fig. 4.4 shows the construction model of the VCPI method.

For the VCEI method, the appearance manifold is constructed by interpolating the mean vectors and the eigenvectors and eigenvalues of two consecutive poses. The covariance matrix for each pose is then calculated from the eigenvectors and
Figure 4.4: Appearance manifold construction using View-dependent Covariance Matrix by training-Point Interpolation (VCPI).

Figure 4.5: Appearance manifold construction using View-dependent Covariance Matrix by Eigenvector Interpolation (VCEI).
the eigenvalues. Here, the appearance manifold will also have different values of mean vector and covariance matrix for each pose. Fig. 4.5 shows the construction model of the VCEI method. Further, the advantages of the VCEI method is its accurateness and efficiency, since this model constructs the appearance manifold without critically controlling the correspondences of every training-sample for the interpolation step.

The construction of an appearance manifold using the VCEI method consists of two stages: the interpolation of mean vectors and the interpolation of eigenvectors and eigenvalues. The mean vector represents the center of samples in each learning pose, while the eigenvectors and eigenvalues represent the covariance matrix as distribution of samples in each pose.

To cover the information of untrained poses, the interpolation processes for the mean vectors can be done by simply selecting one of several existing algorithms. Meanwhile, the interpolation of eigenvectors and eigenvalues are done based on the high-dimensional rotation theory. Here, the eigenvectors and eigenvalues can be considered as axes directions and lengths of a hyper-ellipsoid in an eigenspace. Thus, we consider to obtain the covariance matrices of untrained poses by rotating hyper-ellipsoids from every two-consecutive trained poses. Fig. 4.6 shows the interpolations of the eigenvectors and eigenvalues based on the high-dimensional rotation theory. Buja et al. in [64] also developed a similar mathematical base of high-dimensional rotation for interactive data visualization.

The following three steps were taken to interpolate the eigenvectors and eigenvalues:

**[Step1] Correspond axes directions** The process of checking axes directions between two eigenvectors from two-consecutive poses is necessary, since these axes directions will be used to define the rotation angle of hyper-ellipsoids. Let \( \mathbf{E}_0 \) and \( \mathbf{E}_1 \) be matrices formed by aligning each pair of eigenvectors \( \mathbf{e}_{0j} \) and \( \mathbf{e}_{1j} \) \((j = 1, 2, \cdots, k)\) of covariance matrices \( \Sigma_0 \) and \( \Sigma_1 \). The same process on eigenvalues should be done also by aligning eigenvalues \( \lambda_{0j} \) and \( \lambda_{1j} \) \((j = 1, 2, \cdots, k)\) to form \( k \)-dimensional vectors \( \lambda_0 \) and \( \lambda_1 \). To obtain the correspondences of axes, sort the eigenvectors of \( \mathbf{E}_0 \) and \( \mathbf{E}_1 \) based on their eigenvalues \( \lambda_0 \) and \( \lambda_1 \) to form \( \mathbf{E}'_0 \) and \( \mathbf{E}'_1 \), respectively. Next, perform the same sorting task for the eigenvalues to form \( \lambda'_0 \) and \( \lambda'_1 \) from \( \lambda_0 \) and \( \lambda_1 \). Then, check the angle between the corresponded axes and invert the
eigenvector $e'_{1j}$ $(j = 1, 2, \cdots, k)$ if $e'_{0j}^T e'_{1j} < 0$ so that the angle between corresponded axes is less than or equal to $\pi/2$. For covariance matrix $\Sigma_v$, the eigenvalues $\lambda_{vj}$ $(j = 1, 2, \cdots, k)$ is simply calculated by

$$
\lambda_{vj} = \left( (1 - v) \sqrt{\lambda'_{0j}} + v \sqrt{\lambda'_{1j}} \right)^2 \quad (4.7)
$$

while, its eigenvectors $E_v$ are calculated using a $k \times k$ rotation matrix by

$$
E_v = R(v \phi) E'_0 \quad (4.8)
$$

$R$ represents an interpolated rotation when $0 \leq v \leq 1$. Here, $\phi = [\phi_1, \cdots, \phi_r]$ where $\phi$ is the parameter vector from $r$ number of rotation angles to define the rotation matrix $R$.

**Step 2** Determine the rotation matrix As the elements of $E'_0$ and $E'_1$ are orthogonal, thus, a rotation matrix could be defined by

$$
R(\phi) = E'_1 E'_0^T \quad (4.9)
$$
Refering to the Special Orthogonal (SO) rule, $R(\phi)$ can be diagonalized with a $k \times k$ unitary matrix $U$ and a diagonal matrix $D$ including complex elements as

$$R(\phi) = UD(\phi)U^\dagger,$$  \hspace{1cm} (4.10)

where $U^\dagger$ represents a complex conjugate transpose matrix of $U$. Furthermore, the result of the diagonalization process is a $k$-dimensional diagonal matrix $D(\phi)$ which always has $r$ pairs of complex conjugate roots $e^{i\phi}$ where $e^{i\phi} = \cos \phi + i \sin \phi$. Thus, since each pair of complex conjugate root has a rotation angle, in order to check how many rotation angles there are in $k$-dimensional $D(\phi)$, we calculate $r = \lfloor k/2 \rfloor$. If $k$ is an even number ($k = 2r$), then $D(\phi) = \text{diag}(e^{i\phi_1}, e^{-i\phi_1}, \ldots, e^{i\phi_r}, e^{-i\phi_r})$. However, if $k$ is an odd number ($k = 2r + 1$), then the first diagonal element of the complex conjugate roots is always 1. Thus, $D(\phi) = \text{diag}(1, e^{i\phi_1}, e^{-i\phi_1}, \ldots, e^{i\phi_r}, e^{-i\phi_r})$, where $e^{i\phi} = \cos \phi + i \sin \phi$.

[Step3] Interpolate eigenvectors and eigenvalues  Finally, $R(v\theta)$ can be obtained simply by applying linear interpolation on the vectors. Next, $\Sigma_v$ is calculated by

$$\Sigma_v = E_v \Lambda_v E_v^T$$  \hspace{1cm} (4.11)

where $\Lambda_v$ represents a diagonal matrix with $\lambda_{vj}(j = 1, 2, \ldots, k)$ as the diagonal elements.

### 4.4 Recognition of objects

In the recognition procedure, the Mahalanobis metric provides a sufficient way to classify images in terms of considering their related characteristics and likelihood in each pose class. The Mahalanobis distance becomes a useful way of determining similarity of an unknown sample to known sets.

In order to recognize an input image, it is first projected onto the eigenspace using Eq. 2.6. Then, the distance $d^{(\rho)}(z)$ between the projected-image in the
eigenspace $z$ and the manifold of an object $p$ is calculated as follows:

$$d^{(p)}(z) = (z - \mu^{(p)}(\theta))^T \Sigma^{(p)}(\theta)^{-1}(z - \mu^{(p)}(\theta))$$ (4.12)

Finally, the input image $u$ will be recognized as object $p$ which has the minimum $d^{(p)}(z)$.

### 4.5 Experiments

A series of experiments were designed to evaluate the performance of the proposed methods. First, images of objects were captured using a CCD camera, taken at pose intervals of one degree along the horizontal axis. This corresponds to 360 images per object. The images were then cropped and rescaled to grayscale images with a uniform black background with the size of 32×32 pixels. Here, two datasets of objects were used in the experiments. Dataset 1 contains seven objects with block shapes, while Dataset 2 consists of ten objects with toy figures. The samples of objects used in the experiments are shown in Fig. 4.7.

There were also three pose-interval sets used in the experiments. In the first set, the system was trained with a total of 6,552 images. Each object consists of 36 poses with 10 degree intervals of horizontal poses ($0^\circ$, $10^\circ$, $20^\circ$, $30^\circ$, $40^\circ$, $50^\circ$, $60^\circ$, $70^\circ$, $80^\circ$, $90^\circ$), and each pose consists of 26 training images with a camera-captured image and 25 generated images with various degradation effects. The generated images were obtained by synthesizing artificial noises, such as left and right translations (3, 6, 9, 12, and 15 pixels), clockwise and counter-clockwise rotations ($5^\circ$, $10^\circ$, $15^\circ$, $20^\circ$, and $25^\circ$), and motion blur (5%, 10%, 15%, 20%, 25%). These artificial noises were
generated using MATLAB functions. As the second set, for 30 degree training-pose intervals (0°, 30°, 60°, · · · , 330°), each object has 12 training poses and 2,184 images were used as training images. For the third set, the 60 degree training-pose interval (0°, 60°, 120°, · · · , 300°), there were 6 poses trained for each object. Thus, the total number of training images in this dataset was 1,092 images.

The samples of images of an object in each dataset are shown in Fig. 4.8. The camera-captured images are shown in Fig. 4.8(a). While Fig. 4.8(b) shows the images with blur effects, Fig. 4.8(c) shows the images which are influenced with translation (shift) effects, and Fig. 4.8(d) are images with rotation effects.

The features were extracted using PCA and were projected onto the eigenspace. The appearance manifolds were then created based on each construction method. The interpolation method was uniformed by applying the linear interpolation method to the mean vectors, while the covariance matrices were obtained according to the construction methods explained in Section 4.3.

Here, two methods for constructing the appearance manifold based on the Simple Manifold (SM) model: the Simple Manifold with Non-Degraded center (SMND) and Simple Manifold with Mean center (SMM) methods were evaluated.
In the SMND method, the center of the samples’ distribution is based on the non-degraded (original camera-captured) image. Meanwhile, in the SMM method, the center of the sample distribution is based on the mean vector of image samples in each pose. For both the SMND and SMM methods, an identity matrix was applied to all the covariance matrices in the manifold. For the Constant Covariance matrix (CC) method, the mean vector of image samples become the center of the samples’ distribution for each pose and the average value of all covariance matrices was applied to every covariance matrix along the manifold. For the View-dependent Covariance matrix by training-Point Interpolation (VCPI) and the View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) methods, the center of the samples’ distribution is the mean vector of image samples of each pose. Each pose in the appearance manifold also has a different covariance matrix. However, in the VCPI method, the covariance matrices for the untrained poses were obtained by interpolating each training-point of each pose to a consecutive training-pose. Meanwhile, in the VCEI method, the covariance matrices of untrained poses were obtained by interpolating only the eigenvectors and eigenvalues of two consecutive training-poses, as explained in Section 4.3.3.

Finally, the system was tested with input images which were different from the learning images in horizontal poses ($5^\circ$, $15^\circ$, $25^\circ$, ···, $355^\circ$). Additional experiments were also conducted to observe the degradation in performance of the proposed approaches when the images were influenced with various types of degradation effects. For classification, the Mahalanobis distance measurement was applied.

The performance of the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods) were compared with that of the Simple Manifold with Non-Degraded center (SMND) method, the Simple Manifold with Mean center (SMM) method, and the Constant Covariance matrix (CC) method. The performances were evaluated on various degradation conditions and various training-pose intervals.
4.5.1 Performance in various degradation conditions

To evaluate the performance of the proposed view-dependent covariance matrix methods in various degradation conditions, several experiments were conducted. First, the system was trained with images of 7 objects of Dataset 1 (blocks), then images of 10 objects of Dataset 2 (toy figures) were trained to another system. For both systems, each object consists of 36 poses with 10 degree intervals of horizontal poses (0°, 10°, 20°, · · · , 350°), and each pose consists of 26 training images with a camera-captured image and 25 generated images with various degradation effects.

Table 4.2 presents the average recognition accuracies for the SMND, SMM, CC, VCPI, and VCEI methods for two datasets in two ranges of degradation conditions. The ranges of the degradation conditions are (0–10)% and (0–25)%. The (0–n)% range means that the testing images were influenced with various degradation effects (translation, rotation, blur) within the range of 0% up to n% of image size. For example, the testing images in the (0–25)% range were influenced with 0 pixel (no translation) up to 9 pixels of translation effects, 0° (no rotation) up to 25° of rotation effects, and 0% (no blur) up to 25% of blur effects.

For recognizing objects in Dataset 1 which consists of various block shapes, for (0–10)% range, both the VCPI and VCEI methods achieved higher recognition rates compared with that of the CC method, and the simple manifold methods (the SMND and SMM methods). The VCPI method achieved 90.04% and the VCEI method achieved 89.20%, while the CC, SMND, and SMM methods only achieved 76.41%, 73.68%, and 73.54%, respectively. Next, in the (0–25)% range case, where the degradation effects became more severe, the recognition accuracies of all methods decreased along with the increment level of the degradation effects. However, both the VCPI and VCEI methods maintained their superiority.

In the next experiment, for recognizing toy figures in Dataset 2 with (0–10)% range, the VCPI and VCEI methods also achieved higher recognition rates compared with that of the CC, SMND, and SMM methods. The VCPI method achieved 98.93% and the VCEI method achieved 98.77%. Meanwhile, the CC, SMND, and SMM methods only achieved 77.22%, 58.70%, and 69.35%, respectively. The VCPI and VCEI methods also successfully performed their robustness upon the increment level of the degradation effects. It is shown in the next (0–25)% range case, where the VCPI method still achieved 98.92% and the VCEI method achieved 98.62% of recognition accuracy.
Table 4.2: Results for recognizing objects of two datasets from images with various ranges of degradation effects.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset 1 (0-10%) (%)</th>
<th>Dataset 2 (0-10%) (%)</th>
<th>Dataset 1 (0-25%) (%)</th>
<th>Dataset 2 (0-25%) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Manifold with Non-Degraded center (SMND)</td>
<td>73.68</td>
<td>58.70</td>
<td>55.25</td>
<td>41.50</td>
</tr>
<tr>
<td>Simple Manifold with Mean center (SMM)</td>
<td>73.54</td>
<td>69.35</td>
<td>56.78</td>
<td>53.50</td>
</tr>
<tr>
<td>Constant Covariance matrix (CC)</td>
<td>76.41</td>
<td>77.22</td>
<td>60.91</td>
<td>64.32</td>
</tr>
<tr>
<td>View-dependent Covariance matrix by training-Point Interpolation (VCPI)</td>
<td>90.04</td>
<td>98.93</td>
<td>82.23</td>
<td>98.92</td>
</tr>
<tr>
<td>View-dependent Covariance matrix by Eigenvector Interpolation (VCEI)</td>
<td>89.20</td>
<td>98.77</td>
<td>79.28</td>
<td>98.62</td>
</tr>
</tbody>
</table>
In overall, as shown in Table 4.2, the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods) outperformed the CC method and simple manifold methods (the SMND and SMM methods) in recognizing different types of objects in various degradation conditions for both datasets. In the next sections, we take the case of Dataset 1 in order to present more detailed analysis on the recognition results of the appearance manifold methods.

More detail results are presented in Fig. 4.9 and Fig. 4.10 for recognizing images which are influenced with motion blur effects, translation effects, and rotation effects of Dataset 1 and Dataset 2, respectively.

Fig. 4.9 indicates that the proposed VCPI and VCEI methods always achieved higher recognition accuracies compared with the SMND, SMM, and CC methods for various degradation conditions. For motion blur effects, the VCPI and VCEI methods always give higher recognition accuracies compared with that of the CC and SMM methods. Although the SMND method showed the highest recognition accuracies for 5% and 10% blur effects, when the level of the blur effects were increased to 15%, 20% and 25%, the proposed VCPI and VCEI methods outperformed the SMND method. Next, for recognizing objects with various left and right translation effects (L6 = 6 pixels left translation, L3 = 3 pixels left translation, N = no translation, R3 = 3 pixels right translation, and R6 = 6 pixels right translation), the proposed VCPI and VCEI methods always gave higher recognition accuracies than the SMND, SMM, and CC methods. Finally, the same trend also appeared in recognizing objects with various rotation effects (C10 = 10° clockwise rotation, C5 = 5° clockwise rotation, N = no rotation, CC5 = 5° counter-clockwise rotation, and CC10 = 10° counter-clockwise rotation), where the proposed VCPI and VCEI methods outperformed the SMND, SMM, and CC methods.

Fig. 4.10 of Dataset 2 also shows the robustness of the VCPI and VCEI methods to various degradation conditions. Here, the proposed VCPI and VCEI methods always achieved higher recognition accuracies compared with the SMND, SMM, and CC methods for all conditions: {5%, 10%, 15%, 20% and 25%} motion blur, {L6 = 6 pixels left, L3 = 3 pixels left, N = no, R3 = 3 pixels right, and R6 = 6 pixels right} translation, and {C10 = 10° clockwise, C5 = 5° clockwise, N = no, CC5 = 5° counter-clockwise, and CC10 = 10° counter-clockwise} rotation.
Figure 4.9: Results for recognizing objects in Dataset 1 from images with various degradation effects.
Figure 4.10: Results for recognizing objects in Dataset 2 from images with various degradation effects.
4.5.2 Performance in various training-pose intervals

Several experiments have also been conducted to observe the degradation performance of the proposed view-dependent covariance matrix methods in various training-pose intervals and influenced with various degradation effects. The system was trained with $10^\circ$, $30^\circ$, and $60^\circ$ pose intervals.

In the first set, the 10 degree training-pose interval ($0^\circ, 10^\circ, 20^\circ, \cdots, 350^\circ$), there were 36 poses trained for each object. As the second set, for 30 degree training-pose interval ($0^\circ, 30^\circ, 60^\circ, \cdots, 330^\circ$), each object has 12 training poses. Finally, in the third set, the 60 degree training-pose interval ($0^\circ, 60^\circ, 120^\circ, \cdots, 300^\circ$), there were 6 poses trained for each object. For all pose interval sets, each pose consists of 26 training images (1 camera-captured image + 25 generated images) with various degradation effects, which were obtained by synthesizing artificial noises, such as left and right translations (3, 6, 9, 12, and 15 pixels), clockwise and counter-clockwise rotations ($5^\circ, 10^\circ, 15^\circ, 20^\circ, \text{and} 25^\circ$), and motion blur (5%, 10%, 15%, 20%, 25%).

Fig. 4.11 shows the recognition accuracies of the proposed VCEI method in various training-pose intervals for recognizing objects in Dataset 1 (blocks). For the motion blur case, the decrease in recognition accuracies from $10^\circ$ training-pose interval to $30^\circ$ training-pose interval was 5.32% in average. While higher decrease in recognition accuracies of 7.06% occurred when the training-pose interval was changed from $30^\circ$ to $60^\circ$. In the case of translation effects, the decrease in recognition accuracies from $10^\circ$ training-pose interval to $30^\circ$ training-pose interval was 5.52% in average and 4.80% for changing the training-pose interval from $30^\circ$ to $60^\circ$. Finally, for rotation effects, the decrease in recognition accuracies from $10^\circ$ to $30^\circ$ training-pose interval and from $30^\circ$ to $60^\circ$ training-pose interval were relatively small; 4.56% and 2.74%, respectively. In overall, the recognition accuracy of the proposed VCEI method only decreased 5.13% when $30^\circ$ wider training-pose interval was used.

Fig. 4.12 shows the recognition accuracies of the proposed VCEI method in various training-pose intervals for recognizing objects in Dataset 2 (toy figures). Here, for all degradation conditions, the decrease in recognition accuracies from $10^\circ$ training-pose interval to $30^\circ$ training-pose interval was smaller than the decrease in recognition accuracies from $30^\circ$ training-pose interval to $60^\circ$ training-pose interval. The decrease in recognition accuracies from $10^\circ$ training-pose interval to
Figure 4.11: Results for recognizing objects in Dataset 1 from images with various pose intervals.
Figure 4.12: Results for recognizing objects in Dataset 2 from images with various pose intervals.
30° training-pose interval for motion blur was 0.33% in average. While higher decrease in recognition accuracies of 7.67% occurred when the training-pose interval was changed from 30° to 60°. For the translation effects, the decrease in recognition accuracies from 10° training-pose interval to 30° training-pose interval was 0.01% in average and 11.67% for changing the training-pose interval from 30° to 60°. Finally, in the case of rotation effects, the decrease in recognition accuracies from 10° to 30° training-pose interval and from 30° to 60° training-pose interval were 2.56% and 10.05%, respectively. In overall, for 30° wider training-pose interval used, the recognition accuracy of the proposed VCEI method only decreased 5.38%.

4.6 Discussion

From the results shown in Section 4.5, it is proved that the idea to embed view-dependent covariance matrix in an appearance manifold works well to overcome the problem of recognizing objects from images which are influenced with degradation effects. Novel manifold construction models have been presented, called the View-dependent Covariance matrix by training-Point Interpolation (VCPI) and View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) methods. The VCEI method which constructs the manifold by using only the interpolation of the eigenvectors and the eigenvalues outperformed the VCPI method which needs to interpolate each training image in every trained-pose to its consecutive poses.

Finally, verification results are shown in Fig. 4.13, which shows the visualization of covariance matrices. Here, the contours of the covariance matrix are ellipses whose axes are aligned with the eigenvectors with lengths that are proportional to the corresponding eigenvalues. The directions and lengths of the bold (longer) lines represent the first eigenvectors and its eigenvalues, while the directions and lengths of the regular (shorter) lines represent the second eigenvectors and its eigenvalues. The covariance matrix constructions in Fig. 4.13 were obtained by slicing the appearance manifold of each method on an untrained 45° viewpoint, where each appearance manifold was constructed from 0° and 90° viewpoints. This condition was intentionally set in order to emphasize the difficulty in modelling of interpolation results from two extremely different learning viewpoints.

Fig. 4.13(a) shows the ground truth of a covariance matrix construction, ob-
Figure 4.13: Verification results of various methods.
tained from real image projections, while Fig. 4.13(b), Fig. 4.13(c), Fig. 4.13(d), and Fig. 4.13(e) show the construction results of covariance matrix from the SM, CC, VCPI, and VCEI methods, respectively. Fig. 4.13(d) of the proposed VCPI method presents the most similar construction result of the covariance matrix with the result of real image projections depicted in Fig. 4.13(a). The VCEI method then followed as the second most similar construction result of the covariance matrix with Fig. 4.13(e). Meanwhile, the CC method with its construction result depicted in Fig. 4.13(c) and the SM method in Fig. 4.13(b) were less similar in shape and direction from the ground truth in Fig. 4.13(a). These results confirm their weak recognition capability compared with the proposed view-dependent covariance matrix methods (the VCPI and VCEI methods).

4.7 Summary

In this chapter, novel methods for constructing an appearance manifold with embedded view-dependent covariance matrix have been presented. First, the View-dependent Covariance matrix by training-Point Interpolation (VCPI) method constructs the appearance manifold by interpolating every training-point from one pose class to the training-points in a consecutive pose class. Meanwhile, the View-dependent Covariance matrix by Eigenvector Interpolation (VCEI) method is based on the eigenvalues and eigenvectors interpolations to form a view-dependent covariance matrix model. The proposed view-dependent covariance matrix methods have been implemented in a 3D object recognition system to recognize 3D objects from images that are influenced by various degradation effects, such as translation, rotation, and quality-degradation (motion blur). Experimental results proved that the VCPI and VCEI methods are superior to the other manifold construction types, such as the Constant Covariance matrix (CC) and simple manifold (the SMND and SMM) methods. Their performances also seemed to be consistent even when some degradation factors exist, such as images influenced with different types of degradation effects and when lesser numbers of training images were used. Here, the advantage of the VCEI method is that, since it obtains the view-dependent covariance matrix by interpolating only the eigenvectors and eigenvalues, it is more efficient compared with the VCPI method which has to correspond every training-image in each pose to their consecutive poses.
Chapter 5

Face recognition from video using incremental face manifold

5.1 Introduction

Face recognition has long been an active area of research. Over the years, numerous works have been proposed that focus on recognizing 3D objects and human faces from still-images, such as [7], [22], [24], [30], [31], [32], [33]. Recently, video-based face recognition has attracted much attention since face recognition using video presents various advantages as well as challenges over still-image based recognition.

For an efficient video-based face recognition process, first, every image (frame) in a video sequence is input in a feature extraction module, so that it becomes a low-dimensional vector in a feature space. One most widely used feature extractor in the pattern recognition field is the Principal Component Analysis (PCA) with its eigenspace representation [22][24]. In a video, face images may vary significantly due to environmental changes, such as lighting condition, pose, facial expression, and so on. In addition, various degradation effects might also influence the images in a video sequence, such as low-quality video and cropping errors due to inaccuracies of a tracking system. Therefore, a robust recognition system should be able to handle various image variations.

It is well known that an appearance manifold could capture image variations, especially pose changes, in eigenspace. Addressing various problems, many appearance manifold models have been proposed, such as the simple manifold [30][31], the probabilistic appearance manifold [33][60][61], the layer-transparent manifold [63],
etc. Moreover, various models of appearance manifold with view-dependent covariance matrix which could robustly recognize 3D objects from still-images under various degradation effects [7] were introduced in Chapter 4.

In the past years, several works have reported the use of appearance manifold for face recognition from video sequences. Among them, Raytchev and Murase [65] proposed a pairwise clustering method which calculates the interaction levels (attraction and repulsion) of every input sequence. A merging or splitting process is then applied according to the classification result. Zhou et al. [66] proposed a probabilistic approach which uses joint posterior distribution of motion vectors and estimates the temporal information of video sequences by propagating the identity variables over time. Similar to [66], the work of Lee et al. in [48], [67], utilized local linear models and a transition matrix which propagates the probabilistic likelihood of the linear models to capture temporal information. Meanwhile, Liu et al. [68] processed the temporal information of video sequences using the adaptive Hidden Markov Models (HMM) method.

Taking a different approach from the existing works, the novelty of the approach proposed in this thesis lies in the scheme of embedding view-dependent covariance matrices in appearance manifolds for recognizing faces from video sequences in an incremental unsupervised-learning framework. The advantages of this model are, first, the appearance manifold with view-dependent covariance matrix model is robust to pose changes and also noise invariant since the embedded covariance matrices are calculated based on their poses in order to learn the samples’ distributions along the manifold. Secondly, the proposed incremental unsupervised-learning framework is more realistic for a real-world face recognition application. It is obviously difficult to collect large amounts of face sequences under complete poses (from left sideview to right sideview) for training purpose. Here, an incremental unsupervised-learning framework allows us to train the system with the available initial sequences, and later update the system’s knowledge incrementally every time an unlabelled sequence is input.

In addition, the integration of a pose estimation system in the appearance manifold with view-dependent covariance matrix model also plays an important role in improving the classification accuracy. Since the images of the same object under a different viewing condition is different [65], it is clear that one is likely to obtain the classification results based on the availability (similarity) of a pose
instead of its identity. Therefore, to increase the classification accuracy of the system, the most similar pose of a face in each manifold is searched (i.e. using a pose estimation system), before identifying the person. The pose estimation system also helps the merging process in the incremental unsupervised-learning framework to easily detect video sequences with overlapped poses.

In this chapter, a video-based face recognition problem using the appearance manifold with view-dependent covariance matrix in two learning frameworks: the supervised-learning and the incremental unsupervised-learning is addressed. Fig. 5.1 shows the outline of the proposed video-based face recognition system. In the supervised-learning framework, the training samples are labelled and used to represent the identity categories in the form of appearance manifolds with view-dependent covariance matrices. Meanwhile, in the incremental unsupervised-learning framework, the system first learns the initial categories through the initial manifolds and later updates its knowledge on identity categories by learning the unlabeled input sequences incrementally. Since the structure and the number of the category (class) changes every time a new pattern comes into the system, it is necessary to update the system’s knowledge, either by constructing a new category or by modifying the structure of the existing categories through merging processes between sequences which have some overlapped poses with strong similarities.

The rest of this chapter is organized as follows. The proposed appearance manifold model and the classification process for video-based face recognition in a supervised-learning framework is described in Section 5.2. In Section 5.3, the detailed classification process of video-based face recognition in an incremental unsupervised-learning framework is presented. Experimental results and discussions are described in Sections 5.4 and 5.5, respectively. Finally, the summary of this chapter is presented in Section 5.6.
Figure 5.1: Outline of the proposed video-based face recognition system.
5.2 Supervised-learning framework

Typically, a supervised video-based face recognition system consists of two stages: training and classification. In the training stage, a feature extraction module finds the appropriate features for representing the input patterns and the appearance manifolds are constructed to model the appearance variation of each object. Here, since the objects are human faces, the construction results of the appearance manifolds are called “face manifolds”. Meanwhile, in the classification stage, the classifier assigns an unlabeled input sequence to one of the pattern categories which has the highest similarity based on a distance measurement. The detailed procedures of each module are described in Sections 5.2.1 and 5.2.2.

5.2.1 Supervised manifold construction

In the pattern recognition field, it is well known that appearance-based approaches use sets of images in various poses to represent an object. Here, the role of a feature extraction module is to determine a low dimensional pattern representation compared with the image space. One widely used feature extractor is the Principal Component Analysis (PCA) [22] [24], that computes \( k \)-eigenvectors with the largest corresponding-eigenvalues to project the training samples onto an eigenspace. The linear transformation of the eigenspace representation is defined as:

\[
g_l^{(p)}(\theta) = [e_1, e_2, \cdots, e_k]^T(x_l^{(p)}(\theta) - c)
\]

where \( x_l^{(p)}(\theta) \) is the \( l \)-th sample image of person \( p \) with pose \( \theta \), \( e_i \) \((i = 1, 2, \cdots, k)\) is the eigenvector, \( c \) is the mean vector of the training samples, and \( g_l^{(p)}(\theta) \) is the vector representation of image \( x_l^{(p)}(\theta) \) in the eigenspace. Note that the eigenvectors \( e_i \) in Eq. 5.1 are used only for image projections into the eigenspace, thus, are processed globally regardless to their poses. Meanwhile, later in the construction process of the face manifolds, the eigenvectors and the eigenvalues are view-dependent, since they are derived from the covariance matrix of each training-pose.

The construction process of a face manifold with view-dependent covariance matrices is shown in Fig. 5.2. The input for the construction process is \( x_l^{(p)}(\theta) \) which is the \( l \)-th sample image of person \( p \) with training pose \( \theta \). Next, the construction process of a face manifold with view-dependent covariance matrices consists
of two steps:

**Step 1. Calculation of mean vectors and covariance matrices:** For each training pose, a mean vector and a covariance matrix is calculated. For this purpose, new images are generated by adding noise to each image in video-captured sequences. The type, level and number of the artificial noise are not limited and could be applied freely in various forms, such as geometric distortion (i.e. shift, rotation, etc.), quality degradation (i.e. blur, salt and pepper noise, etc.), illumination changes, etc.

**Step 2. Interpolation of mean vectors and covariance matrices:** In order to obtain the mean vectors and the covariance matrices of the untrained poses, interpolation processes are performed to each pair of mean vectors and covariance matrices of two consecutive training poses. For the mean vectors, the interpolation process is done by simply using one of the several existing interpolation algorithms. Meanwhile, the interpolation of the
covariance matrices is done by interpolating the corresponding eigenvectors and eigenvalues of two consecutive training poses.

The eigenvectors and eigenvalues interpolation process is described as follows. First, the matrices of eigenvectors $E_0$ and $E_1$ which consist of pairs of eigenvectors $e_{0j}$ and $e_{1j}$ ($j = 1, 2, \ldots, k$) and the matrices of eigenvalues which consist of pairs of eigenvalues $\lambda_{0j}$ and $\lambda_{1j}$ ($j = 1, 2, \ldots, k$) of covariance matrices $\Sigma_0$ and $\Sigma_1$ are formed. Next, in order to correspond the axes, the eigenvectors of $E_0$ and $E_1$ are sorted based on their eigenvalues $\lambda_0$ and $\lambda_1$ to form $E'_0$ and $E'_1$, respectively. The same task for the eigenvalues are then performed to form $\lambda'_0$ and $\lambda'_1$ from $\lambda_0$ and $\lambda_1$. Then, check and invert if the eigenvector $e'_{0j}^T e'_{1j} < 0$ so that the angle between corresponded axes is less than or equal to $\pi/2$.

For covariance matrix $\Sigma_v$, the eigenvalues are calculated by $\lambda_{vj} = \left( (1 - v)\sqrt{\lambda_{0j}} + v\sqrt{\lambda_{1j}} \right)^2$ ($j = 1, 2, \ldots, k$) and $E_v = R(v\phi)E'_0$ for the eigenvectors. Here, $R$ represents an interpolated rotation when $0 \leq v \leq 1$ and $\phi = [\phi_1, \ldots, \phi_r]$ represents the parameter vector of rotation angles to define the rotation matrix. Since the rotation angles always come in pairs in the complex conjugate roots process, $r = \lfloor k/2 \rfloor$.
The rotation matrix is defined by $R(\phi) = E_1'E_0^T$ and diagonalized with the Special Orthogonal (SO) rule by $R(\phi) = UD(\phi)U^\dagger$ where $U^\dagger$ represents a complex conjugate transpose matrix of $U$. The complex conjugate roots are then processed by $D(\phi) = diag(e^{i\phi_1}, e^{-i\phi_1}, \cdots, e^{i\phi_r}, e^{-i\phi_r})$ if $k = 2r$. Meanwhile, if $k = 2r + 1$, then $D(\phi) = diag(1, e^{i\phi_1}, e^{-i\phi_1}, \cdots, e^{i\phi_r}, e^{-i\phi_r})$ where $e^{i\phi} = \cos \phi + i \sin \phi$. Finally, the rotation matrix $R(v\theta)$ is interpolated and the covariance matrix is calculated for the untrained poses using $\Sigma_v = E_v\Lambda_v E_v^T$ with $\Lambda_v = diag(\lambda_v)$. Fig. 5.3 shows the illustration of the interpolation of the eigenvectors and the eigenvalues in a 2D feature space.

The output of the construction process of the face manifold with view-dependent covariance matrix are the mean vectors $\mu^{(p)}(\theta)$ and the covariance matrices $\Sigma^{(p)}(\theta)$ of the training poses and poses obtained by the interpolation.

### 5.2.2 Classification of face-sequences

In the testing stage, unlike still-image recognition, video-based recognition needs to integrate the classification results of each frame to produce the decision of a sequence. Given a face sequence $S$ with $h$ frames $S = [f_1, f_2, \cdots, f_h]$ in an eigenspace, the classification process of a face image $f_i$ ($i = 1, 2, \cdots, h$) is based on its similarity to the trained face manifolds $O$. Meanwhile, the final sequence’s classification decision is based on the minimal cumulative distance of each $f_i \in S$.

In this thesis, the integration of a pose estimation system to provide pose information of each test image $f_i \in S$ is also proposed for improving the classification accuracy and to easily detect the sequences with overlapped poses for the merging process in the incremental unsupervised-learning framework. Here, various existing algorithms can be selected for developing the pose estimation system, since it is fully independent from the classification system. In this paper, the pose estimation system is based on the Nearest Neighbor algorithm which basically classifies an input feature vector to a class with the nearest distance in a feature space.

For classification purpose, first, a pose vector $\theta^{(p)} = [\theta_1, \theta_2, \cdots, \theta_a]$ which consists of training poses of $a$ frames in a face manifold $O$ is constructed. Then, the classification process is defined as follows. The pose $\phi_i$ of each unlabelled input image $f_i$ ($i = 1, 2, \cdots, h$) is determined by a pose estimation system by:
\[ \varphi_i = \text{pose\_estimation}(f_i) \]  \hspace{1cm} (5.2)

Next, the normalization of the test image can be calculated by:

\[ f'_i = f_i - \mu^{(p)}(\varphi_i) \]  \hspace{1cm} (5.3)

and the distance measurement of the input image is defined by:

\[
d_f^{(p)}(p) = \begin{cases} 
(f'_i)^T \left( \Sigma^{(p)}(\varphi_i) \right)^{-1} (f'_i) & (\varphi_i \in \theta^{(p)}) \\
0 & \text{(otherwise)}
\end{cases} \hspace{1cm} (5.4)
\]

Equation 5.4 shows that the Mahalanobis distance is calculated only when the pose \( \varphi_i \) of an input image exists as a member of pose vector \( \theta^{(p)} \) of manifold \( O \) (defined as an overlapped pose). On the other hand, when the pose of the input image \( \varphi_i \) is not a member of \( \theta^{(p)} \), a zero value is given to \( d_f^{(p)} \) as the distance of the input image \( f_i \) to the manifold \( O \).

Finally, the classification decision of an input sequence \( S_i \) is made upon integrating the classification results of all input images \( f_i \in S \). The identity \( p^* \) is determined by finding the manifold \( O \) with the minimal cumulative distance of all \( f_i \in S \), as follows:

\[ p^* = \arg \min_p \left( \frac{\sum_{i=1}^{h} d_f^{(p)}(p)}{\text{noo}(p)} \right) \hspace{1cm} (5.5) \]

where \( h \) is the number of frames in the input sequence \( S \) and \( \text{noo}(p) \) is the number of overlapped poses between the input sequence \( S \) with the existing manifold \( O \).
5.3 Incremental unsupervised-learning framework

Considering the practical interest, a face recognition system is expected to deal with unconstrained, dynamic and unpredictable environments. The system should be able to correctly identify every input and learn it to update the system's knowledge on identity categories (persons). The purpose of the incremental unsupervised-learning introduced in this paper is to assign a set of unlabelled face sequences into their corresponding identity categories (persons) in an unsupervised manner. The input sequence can be assigned to one of the existing categories or as a new identity category. The major difficulty of this approach is in finding the proper balance between not to overlook the existing category structure and at the same time not to superimpose a new structure. The proposed approach is based on central clustering which classifies an unlabelled input sequence into an identity category according to its similarity to the central feature of each category with prior guidance of a pose estimation system. Fig. 5.4 shows the summary of the identity classification (clustering) and learning algorithm in an incremental unsupervised-learning framework, while the detailed processes are described in the following sections.

5.3.1 Incremental unsupervised manifold construction

In the incremental unsupervised-learning framework, the system allows us to initially construct the appearance manifolds with initial training sequences. Later, an updating process is conducted to incrementally update the system’s knowledge every time an unlabelled sequence is input. The initial appearance manifolds can be constructed by using the same procedure with the supervised manifold construction, explained in Section 5.2.1. Meanwhile, the updating process of the appearance manifold is described in Section 5.3.2.

5.3.2 Classification and updating of face-sequences

The classification process of an input image $f_i$ ($i = 1, 2, \cdots, h$) in a sequence $S = [f_1, f_2, \cdots, f_h]$ which consists of $h$ frames starts by measuring the similarity of each $f_i \in S$ using Eq. 5.4. Next, a threshold value $\beta$ is defined in order to determine whether an input sequence is classified to one of the existing categories (accepted) or assigned as a new identity category (rejected).
Initialization:

\[ S = [f_1, f_2, \ldots, f_h] = \text{a face sequence consisting of } h \text{ images} \]
\[ p = (1, 2, \ldots, P) = \text{the existing identity categories} \]
\[ df_i^{(p)} = \text{the distance of image } f_i \text{ to manifold } O \]
\[ noo_i^{(p)} = \text{the number of overlapped poses between } S \text{ with manifold } O \]
\[ \beta = \text{threshold} \]

Begin

If satisfy the condition \( df_i^{(p)} < \beta \) \((i = 1, \ldots, h)\)

Then

**Accepted:**
1. Identify a sequence \( S \) according to the most similar identity category \( p \)
   \[ p^* = \arg \min_p \left( \sum_{i=1}^{h} df_i^{(p)} / noo_i^{(p)} \right) \]
2. Construct the manifold of \( S \) (\( O_{\text{input}} \))
3. Merge parts of \( O_{\text{existing}} \) and \( O_{\text{input}} \) which have overlapped poses

Else

**Rejected:**
1. Identify the sequence \( S \) as a new identity class \( p^* = P + 1 \)
2. Construct the manifold of \( S \) (\( O_{\text{input}} \))
3. Add \( O_{\text{input}} \) to the learning stage

End

Figure 5.4: Classification and learning algorithm of an input sequence in an incremental unsupervised-learning framework.
An input sequence is *accepted* if it is classified to one of the existing categories where the distance of every image in the input sequence are less than a threshold value $\beta$. Otherwise, the input sequence is *rejected* and registered as a new identity category. The identity category is determined as follows:

$$p^* = \begin{cases} \arg \min_p \left( \sum_{i=1}^{h} d_i^{(p)} / \text{noo}^{(p)} \right) & \text{(Accepted)} \\ P + 1 & \text{(Rejected)} \end{cases} \tag{5.6}$$

with $h$ is the number of frames in the input sequence $S$ and $\text{noo}^{(p)}$ is the number of overlapped poses between the input sequence $S$ with the existing manifold $O$.

The next process after the identity classification is to construct the face manifold of the input sequence in order to update the system’s knowledge. Here, the face manifold of the input sequence is constructed using the same model with the initial face manifolds by embedding the view-dependent covariance matrix (see the construction process in Section 5.3.1). Moreover, for each *accepted* result, it is also necessary to perform a merging process between the manifold of the input sequence with the existing manifolds which have the same identity category. However, the manifold merging process is not performed for any *rejected* result. The details of the manifold merging process is described in the next paragraph.

In the manifold merging process, only the overlapped parts of two face manifolds with view-dependent covariance matrix will be merged. Therefore, it is necessary to determine the overlapped parts of both manifolds by detecting the overlapped poses $\omega_i$. Fortunately, the detecting process of the overlapped poses $\omega_i$ can be performed easily by the help of an integrated pose estimation system.

The merging process of two face manifolds with view-dependent covariance matrix is done as follows. First, the mean vectors $\mu_i^{(p^*)}(\omega_i)$ of overlapped poses $\omega_i$ are merged through:

$$\mu_{i,\text{update}}^{(p^*)}(\omega_i) = \gamma \mu_{i,\text{exist}}^{(p^*)}(\omega_i) + (1 - \gamma) \mu_{i,\text{new}}^{(p^*)}(\omega_i) \tag{5.7}$$

Next, the covariance matrices $\Sigma_i^{(p^*)}(\omega_i)$ are merged using:

$$\Sigma_{i,\text{update}}^{(p^*)}(\omega_i) = \gamma \Sigma_{i,\text{exist}}^{(p^*)}(\omega_i) + (1 - \gamma) \Sigma_{i,\text{new}}^{(p^*)}(\omega_i) \tag{5.8}$$

where $\gamma$ is an updating weight value, $\mu_{i,\text{exist}}^{(p^*)}(\omega_i)$ and $\Sigma_{i,\text{exist}}^{(p^*)}(\omega_i)$ are the mean vectors and the covariance matrices of the existing manifold $p^*$ with overlapped poses.
Figure 5.5: The preprocessing step of an image (frame) of a video sequence.

\[ \omega_i, \mu_{i,new}^{(p^*)}(\omega_i) \text{ and } \Sigma_{i,new}^{(p^*)}(\omega_i) \] are the mean vectors and the covariance matrices of the input manifold and \[ \mu_{i,update}^{(p^*)}(\omega_i) \text{ and } \Sigma_{i,update}^{(p^*)}(\omega_i) \] are the mean vectors and the covariance matrices of the newly merged manifold \( p^* \) with overlapped poses \( \omega_i \).

5.4 Experiments and analysis

Several experiments have been conducted to evaluate the performance of the proposed method in recognizing human faces from video sequences. For the experiments, 60 motion videos of 20 persons with pose changes from \(-90^\circ\) (left sideview) to \(+90^\circ\) (right sideview) from the frontal pose have been collected. For each person, three motion videos are taken in a different time under different conditions and are represented in three datasets. In the preprocessing step, first, the motion videos are trimmed with a frame rate of 30 frames/second, and images from a video sequence with \(10^\circ\) pose differences from each other were taken as a face sequence. Here, the sampling processes of the face sequences were performed in order to obtain a same frame-density condition within the face sequences, which is useful for fair evaluations of methods. However, in a real system, the sampling process is not necessary. Next, each image of the face sequences were manually cropped and downsampled to 32x32 pixels image size. In a real application system, however, a face tracking (cropping) system may be used to automatically detect and crop the face images. The preprocessing step of an image (frame) of a video sequence is shown in Fig. 5.5.
Figure 5.6 shows the samples of face sequences of four persons from three datasets: (a) Dataset 1, (b) Dataset 2: small face variations taken in a different time, and (c) Dataset 3: severe face variations. It is clearly seen in Fig. 5.6 that the datasets contain many instances of noise data, in the form of pose variations, natural expression variations, and erroneous face cropping (misalignments). In the experiments, face sequences of Dataset 1 were used for training, while those of Dataset 2 and Dataset 3 were used as testing data.

The experiments were conducted in two frameworks: supervised-learning and incremental unsupervised-learning, in which each of them has varying degrees of difficulty. In this thesis, the results of the proposed appearance manifold with view-dependent covariance matrix (VC) method were compared with that of the simple manifold (SM) method (known as the Parametric Eigenspace method in [31]). Both of these methods use a manifold to capture pose variabilities of a face. However, in the Simple Manifold (SM) method, an appearance manifold is constructed based on the interpolations of the mean vectors of samples and uses an identity matrix as the covariance matrix for each pose. Therefore, the SM method can only capture the pose changes. Meanwhile, in the proposed VC method, a view-dependent covariance matrix is also embedded to the appearance manifold for each pose so that it includes the ability to learn the distribution of samples. Therefore, the proposed VC method has the ability to (1) capture pose variability and (2) learn the distribution of samples. Here, both the VC and the SM methods are also combined with a pose estimation system using the same Nearest Neighbor algorithm. The experimental results are presented in the next sections.
(a) Dataset 1.

(b) Dataset 2 (small pose variations).

(c) Dataset 3 (severe pose variations).

Figure 5.6: Samples of face sequences. Dataset 1 was used for training, while Dataset 2 and Dataset 3 were used for testing.
5.4.1 Performance in supervised-learning

As a first experiment in the supervised-learning framework, the training set consists of 26 face sequences/person in Dataset 1. Among these 26 face sequences, only one sequence was captured by a motion video, while the other 25 sequences were generated by applying noise effects to the video-captured sequence. The noise effects applied in this experiment are the motion blur effects and the shift and rotation effects which usually occur in the capturing process of moving objects. For the testing set, the face sequences from Dataset 2 and Dataset 3 (which are different from the training set) were chosen and were arranged into 200 partial face sequences for each of the 9 different sequence’s lengths. Here, one sequence’s length represents 10° pose width.

Figure 5.7(a) shows the accuracy rates of the VC and the SM methods when recognizing faces with 10 identity categories (persons) from video sequences of Dataset 2 with various sequence’s lengths in the supervised-learning framework. For recognizing faces with 10 identity categories (persons), the system was trained with 26 face sequences for each of the 10 different persons in Dataset 1 and was tested with 20 different face sequences for each 10 persons with 9 different sequence’s length in Dataset 2. The results in Fig. 5.7(a) shows that the proposed VC method gave higher recognition accuracies in all categories compared with that of the SM method. Here, the longer the sequence’s length, the higher accuracies could be achieved by the system, since a long sequence could usually give more appearance variation information compared with a short sequence. The highest accuracy rate for the VC method was 98%, while the highest accuracy rate for the SM method was only 67%.

Next, for a 20 identity categories (persons) recognition task, the system was trained with 26 face sequences for each of the 20 different persons in Dataset 1 and was tested with 10 different face sequences for each 20 persons with 9 different sequence’s length in Dataset 2. As depicted in Fig. 5.7(b), the accuracies of the recognition system of the 20 identity categories recognition task decreased compared with that of the 10 identity categories recognition task. However, the proposed VC method still outperformed the SM method, with 88% highest recognition accuracy for the VC method, while the SM method only achieved 45% as its highest recognition accuracy.

Furthermore, an experiment to recognize faces from video sequences of Dataset 3
Figure 5.7: Accuracy rates in recognizing faces from video sequences of Dataset 2 with various sequence’s lengths in the supervised-learning framework.

(a) 10 identity categories (persons) task

(b) 20 identity categories (persons) task
Table 5.1: Accuracy rates of a 10 identity categories recognition task from video sequences with 8 sequence’s length in two conditions: small pose variation and severe pose variation in a supervised-learning framework (The reference accuracy values, obtained when using pose information given by a human, are presented in the brackets).

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy of Dataset 2 with small pose variations</th>
<th>Accuracy of Dataset 3 with severe pose variations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Manifold (SM)</td>
<td>67% (73%)</td>
<td>67% (60%)</td>
</tr>
<tr>
<td>View-dependent Covariance (VC)</td>
<td>98% (98%)</td>
<td>75% (88%)</td>
</tr>
</tbody>
</table>

which contains severe image variations were also conducted. Table 5.1 summarizes the accuracy rates of the 10 identity categories recognition task from video sequences with 8 sequence’s length in two conditions: small face variation and severe face variation. The system was tested with 20 different face sequences for each 10 persons. It could be well understood that the recognition accuracies of all methods decreased when recognizing sequences with severe face variations. However, the VC method could still maintain its superiority on the SM method. Under severe face variation conditions, the highest recognition accuracy achieved by the VC method was 75%, while the SM method gave only 67% accuracy as its highest recognition result. For reference purpose, the accuracy rates of the recognition system for each method when the pose information was given by a human were also presented, as shown as values within brackets in Table 5.1.

All experimental results described above showed that the proposed VC method outperforms the SM method in the supervised-learning framework in various conditions, such as various numbers of identity category, various numbers of sequence’s length, different levels of image variation (small or severe), and different pose estimation techniques (human estimator or system estimator).
5.4.2 Performance in incremental unsupervised-learning

In the incremental unsupervised-learning framework, the system first learns the initial manifolds and later the existing manifolds are updated automatically by learning the sequences which are input incrementally into the system. Therefore, in this experiment, 10 initial manifolds of 10 persons using 26 (1 original and 25 generated) face sequences of Dataset 1 per person were constructed. Next, several parameters such as the threshold value $\beta$ (see Fig. 5.4) and a merging weight $\gamma$ (see Eq. 5.7 and Eq. 5.8) were defined experimentally. Fig. 5.8 shows the accuracy rates of the VC and the SM methods in recognizing short (1 sequence’s length), medium (4 sequence’s length), and long (8 sequence’s length) sequences with manual pose estimation and using various threshold values. Based on the results in Fig. 5.8, the parameters were set as $\beta = 2.0$ and $\gamma = 0.5$ as the optimal threshold and update weight values for the face database.

![Figure 5.8: Accuracy rates in recognizing faces from video sequences with various lengths using various threshold values.](image-url)
Figure 5.9 presents the classification rates for recognizing faces from video sequences in an incremental unsupervised-learning framework. The system was tested with 10 face sequences for each different 20 persons (10 initial persons and 10 new persons) with 9 various sequence’s lengths. For the experiments in the incremental unsupervised-learning framework, the evaluation parameters were the accuracy rate, the false acceptance rate, and the false rejection rate. The definitions of each evaluation parameter are defined as follows:

- Accuracy rate: the ratio of the number of classes that are correctly classified by the system to the total number of tests.
- False acceptance rate: the ratio of the number of pairs of different classes that are incorrectly matched by the system to the total number of match attempts.
- False rejection rate: the ratio of the number of pairs of the same class that are not matched by the system to the total number of match attempts.

Each evaluation rate is presented in a separate figure in Fig. 5.9 in order to give a clear presentation of the experimental results. Figure 5.9(a), Fig. 5.9(b), and Fig. 5.9(c) show the accuracy rates, the false acceptance rates, and the false rejection rates of the VC and the SM methods when recognizing faces from video sequences in Dataset 2 with various sequence’s lengths. The results in Fig. 5.9(a) show that the proposed VC method gave higher accuracy rates compared with that of the SM method in all categories. Similar with the results in the supervised learning framework, the longer the face-sequences, the higher accuracy rates could be achieved by the system. For the error rates, it can be seen from Fig. 5.9(b) that the false acceptance rates of the proposed VC method were lower than that of the SM method. Moreover, the false acceptance rates for the VC method decreased along with the increment of the sequence’s length. On the contrary, the false rejection rates of the VC method increased along with the increment of the sequence’s length. The reason is as follows: tracing back the proposed identity classification (clustering) algorithm in Fig. 5.4 where the distance of every image in a face sequence should be less than the threshold value, it could be well understood that the difficulty level of fulfilling this criteria (being accepted) increases along with the increment of the sequence’s length. On the other hand, the false acceptance rate of the SM method increased along the increment of the
Figure 5.9: Classification rates in recognizing faces from video sequences of Dataset 2 with various sequence’s lengths in an incremental unsupervised-learning framework.
Table 5.2: Classification rates in recognizing faces from video sequences with 8 sequence’s length in an incremental unsupervised-learning framework (The reference accuracy values, obtained when using pose information given by a human, are presented in the brackets).

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>False acceptance</th>
<th>False rejection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Manifold (SM)</td>
<td>34% (37%)</td>
<td>100% (100%)</td>
<td>33% (27%)</td>
</tr>
<tr>
<td>View-dependent Covariance (VC)</td>
<td>83% (87%)</td>
<td>0% (0%)</td>
<td>35% (26%)</td>
</tr>
</tbody>
</table>

sequence’s length, while the false rejection rate of the SM method decreased along the increment of the sequence’s length. In more detail, the false acceptance rates for the SM were nearly 100% for all categories, which means that the SM method was not able to differentiate new persons from the trained persons. This condition, on the contrary triggers the decrement of the false rejection rate.

Table 5.2 summarizes the classification rates for recognizing faces from video sequences of Dataset 2 with 8 sequence’s length in an incremental unsupervised-learning framework, which also shows the highest recognition accuracies achieved by both methods. Here, the system was initially trained with 10 initial manifold of 10 persons using 26 (1 original and 25 generated) face sequences of Dataset 1 per person and was tested with 10 face sequences for each different 20 persons (10 initial persons and 10 new persons) of Dataset 2 with 8 sequence’s length. The accuracy rate achieved by the VC method was 83%, with 0% false acceptance rate and 35% false rejection rate. Meanwhile, the SM method only gave 34% accuracy rate with 100% false acceptance rate and 33% false rejection rate. For reference purpose, the accuracy rates of the recognition system for each method when the pose information was given by a human were also presented, as shown as values within brackets in Table 5.2.

It is clearly seen from all evaluation parameters in Fig. 5.9 and Table 5.2 that the proposed VC method outperformed the SM method in all categories in an incremental unsupervised-learning framework.
5.5 Discussion

In Section 5.4, the performances of the proposed VC method and its comparisons with the SM method were presented, where all results showed that the VC method outperforms the SM method in various conditions in both supervised and incremental unsupervised-learning frameworks. This section discusses in more detail the performance of the proposed VC method in incremental unsupervised-learning framework.

As has been mentioned earlier, the advantage of an incremental unsupervised-learning framework is that the system could learn and update its knowledge automatically as the unlabelled sequences are input incrementally into the system. However, one critical point in the incremental unsupervised-learning approach is that the classification ability of the system is highly dependent on the initial settings (i.e. the threshold value, the updating weight, the number of the initial manifold, etc.) and the condition of the input sequences (i.e. the sequence’s length, the number of overlapped poses, the input order, etc.). Thus, using different initial settings and/or processing different conditions of input sequences could give different classification results.

Table 5.3 presents the classification rates of two experiments which have the same initial settings but different conditions of input sequences. For the initial setting, 10 initial manifolds of 10 persons were constructed, $\beta = 2.0$ and $\gamma = 0.5$ were defined, and the Nearest Neighbor algorithm was used as the pose estimation system. In the testing process, the sequence’s length and the input order of the unlabelled sequences were set differently. In the first experiment, the system relatively processed longer input sequences within the range of 5–8 sequence’s length than in the second experiment whose sequence’s length was within the range of 1–8. The number of unlabelled input sequences were 100 sequences (randomly input from 5 sequences for 20 persons) which also shows how many times the system was updated. From Table 5.3, it can be seen that for both experiments, the VC method outperformed the SM method with 85% highest accuracy rate, 12% false acceptance rate and 18% false rejection rate. Meanwhile, the SM method only gave 18% accuracy rate, with 100% false acceptance and 64% false rejection rates for both experiments.
Table 5.3: Classification rates of two experiments with the same initial settings but different testing conditions.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Method</th>
<th>Sequence’s Length</th>
<th>Accuracy (%)</th>
<th>False Acceptance (%)</th>
<th>False Rejection (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Simple Manifold (SM)</td>
<td>5–8</td>
<td>18</td>
<td>100</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>View-dependent Covariance Matrix (VC)</td>
<td>5–8</td>
<td>85</td>
<td>12</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>Simple Manifold (SM)</td>
<td>1–7</td>
<td>18</td>
<td>100</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>View-dependent Covariance Matrix (VC)</td>
<td>1–7</td>
<td>78</td>
<td>22</td>
<td>22</td>
</tr>
</tbody>
</table>
Fig. 5.10, Fig. 5.11, Fig. 5.12, and Fig. 5.13 show the rendered images of manifolds in a 3D eigenspace. The experiments were carried out following the concept of the VC method in an incremental unsupervised-learning framework. First, the face sequences are input sequentially into an eigenspace, then the system classifies them. A new manifold is constructed in the eigenspace when a face sequence is classified as a new category (person), while two manifolds with the same category (person) will be merged.

Fig. 5.10 shows an input sequence A of person 1. Since sequence A was classified as a new category, a manifold A was constructed in a 3D eigenspace. Next, Fig. 5.11 shows a new input sequence B of person 2. Since sequence B was also classified as a new category, a manifold B was constructed in the existing eigenspace. Fig. 5.11 also shows the position of a previously constructed manifold A in a 3D eigenspace. Another manifold C was also added into the existing eigenspace when a sequence C of person 3 (a new person) was input. Here, the constructed manifold C was shorter than the other manifolds since the input sequence C consists of less poses. Finally, when a sequence D of person 3 which consists of new poses of a trained person was input, a manifold D was constructed and was merged to manifold C in a 3D eigenspace since it belongs to a same category (person 3). These results show that the proposed VC method worked well in the incremental unsupervised-learning framework.

Further, Fig. 5.14 shows the visualization of face manifolds using the VC method with poses from \(-90^\circ\) (left sideview) to \(+90^\circ\) (right sideview) from the frontal pose. Fig. 5.14(a) shows the visualization of a face manifold of a person which was constructed from the sequences of Dataset 1 and Dataset 2 in a supervised-learning framework. Meanwhile, Fig. 5.14(b) and Fig. 5.14(c) show the visualization of face manifolds of a same identity (person) in the incremental unsupervised-learning frameworks (the initial manifolds were constructed from sequences of Dataset 1, and later the unlabelled sequences of Dataset 2 were input incrementally to update the initial manifold). The input sequences for Fig. 5.14(b) had a 7 sequence’s length, while, in Fig. 5.14(c), shorter sequences with 3 sequence’s length were input. From Fig. 5.14, it can be seen that the constructed manifolds in the incremental unsupervised-learning frameworks are similar to each other and also to the construction result of the manifold in the supervised-learning framework.
Figure 5.10: A rendered image of a manifold A in a 3D eigenspace, based on the input sequence A.

Figure 5.11: A rendered image of manifolds A and B in a 3D eigenspace, based on the previously input sequence A and a new input sequence B (a new person).
Figure 5.12: A rendered image of manifolds A, B, and C in a 3D eigenspace, based on the previously input sequences A and B, and a new input sequence C (a new person).

Figure 5.13: A rendered image of manifolds A, B, and C+D in a 3D eigenspace, based on the previously input sequences A, B, and C, and a new input sequence D (a trained person with new poses).
Figure 5.14: Visualization of face manifolds of a person using the proposed VC method. (a) obtained by the supervised-learning framework, (b–c) obtained by the incremental unsupervised-learning frameworks from different unlabelled input sequences.
Finally, in order to emphasize the superiority of the proposed method, the accuracy rates of the PCA and nonlinear RBF Kernel PCA in combination with the simple Nearest Neighbor (NN) classifier and the appearance manifold with View-dependent Covariance (VC) are presented in Table 5.4. For the combinations with NN, the training classes included only the samples of the training poses. Meanwhile, for the combinations with VC, the manifolds with view-dependent covariance matrix were constructed. Here, constructing a manifold means attaining estimation of continuous poses by interpolating the classes (the mean vectors and the covariance matrices) of two consecutive training poses to obtain those of the untrained poses. Thus, the proposed VC method synthesized more pose variabilities than the NN method, since it has training classes with more complete poses (training poses + interpolation poses).

Due to the fact that the appearance of a person’s face is highly dependent on its pose, it is obvious that an appearance-based method which can capture more pose variabilities can give more accurate recognition. Moreover, in the proposed VC method, the covariance matrices were embedded in the appearance manifold. Therefore, the advantage of the proposed VC method includes the abilities to capture pose variability and also learn the sample’s distribution of each pose. As the consequence, the proposed VC method can give higher recognition accuracies than the NN method. Table 5.4 shows that for both the linear PCA and the RBF Kernel PCA feature extraction techniques, the proposed VC method gave higher recognition accuracies than that of the NN method. The results also show that the VC method worked better for both the linear PCA and the nonlinear RBF Kernel PCA feature extraction techniques.

Furthermore, the structure of a manifold in the proposed VC method is feature-space independent because a manifold is constructed only by the interpolation of classes of two consecutive training poses. As an interpolation technique can be applied to any feature-space, the structure of the constructed manifold is also not affected by the linearity or non-linearity of the feature-space.
Table 5.4: Accuracy rates of a 10 identity categories recognition task from video sequences with 30° training pose differences between frames (sparse training sequences).

<table>
<thead>
<tr>
<th>Method</th>
<th>Training classes include samples of</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA + Nearest Neighbor (NN)</td>
<td>Training poses only</td>
<td>80</td>
</tr>
<tr>
<td>PCA + View-dependent Covariance (VC)</td>
<td>Manifold (Training poses + Interpolation)</td>
<td>92</td>
</tr>
<tr>
<td>RBF Kernel PCA + Nearest Neighbor (NN)</td>
<td>Training pose only</td>
<td>83</td>
</tr>
<tr>
<td>RBF Kernel PCA + View-dependent Covariance (VC)</td>
<td>Manifold (Training poses + Interpolation)</td>
<td>84</td>
</tr>
</tbody>
</table>
5.6 Summary

In this chapter, two learning frameworks of the appearance manifold with view-dependent covariance matrix have been presented. In the supervised-learning framework, the training samples are labelled and an appearance manifold with view-dependent covariance matrix is used to represent an identity category. Meanwhile, in the incremental unsupervised-learning framework, the system first learns the initial categories through the initial manifolds, then the unlabelled input sequences are incrementally learned in order to update the existing identity categories of the system.

The advantages of this method are, first, the appearance manifold with view-dependent covariance matrix model is robust to pose changes, since the embedded covariance matrices are calculated based on their poses in order to learn the samples’ distributions along the manifold. Second, the proposed incremental unsupervised-learning framework is more realistic for a real-world face recognition application, since it allows us to train the system with the available initial sequences, and later update the system’s knowledge incrementally every time an unlabelled sequence is input. The integration of a pose estimation system in the appearance manifold with view-dependent covariance matrix model is intended to improve the classification accuracy of the system and to easily detect the overlapped poses in video sequences which is useful for the merging process in the incremental unsupervised-learning framework. The merging process is performed in order to merge the manifolds which have some overlapped poses with strong similarities. The experimental results showed that in both frameworks, the proposed appearance manifold with view-dependent covariance matrix method outperforms the simple manifold model in recognizing faces from video sequences.
Chapter 6

Conclusion

6.1 Summary

In this thesis, the recognition problem is formulated as an appearance matching process rather than shape. Therefore, the proposed methods follow the appearance-based approach. Three-dimensional object and face recognition systems have been developed and their performances were studied through several experiments. The developed systems meet the objectives stated in Chapter 1 from the points of view in accuracy and robustness to pose changes, degradation effects, and face expressions.

- The first proposed approach was the Modified Nearest Feature Line (MNFL) method (see Chapter 3). The MNFL method proposed to construct feature lines of objects, which are the lines passing through every pair of samples (Eigenpoints), in order to capture object variations. These feature lines are constructed by applying the linear interpolation and extrapolation techniques between samples in an Eigenspace. The derived feature lines virtually provide an infinite number of Eigenpoints of the class, thus, they expand the capacity of the available database. The experimental results confirmed that the developed face recognition system using the MNFL method accurately recognizes face images with various pose and expressions and outperforms the conventional NFL method.

- In Chapter 4, a novel construction method of an appearance manifold with embedded view-dependent covariance matrix for 3D object recognition has
been proposed. The main advantage of the proposed appearance manifold with view-dependent covariance matrix method is that it has the ability to capture pose variability and also learn the samples' distribution of each pose. With these abilities, the appearance manifold with view-dependent covariance matrix method is robust to pose changes and also invariant to various degradation effects. A 3D object recognition system has been developed to recognize 3D objects from images that are influenced by various degradation effects, such as translation, rotation, and quality-degradation (motion blur). Experimental results proved that the proposed appearance manifold with view-dependent covariance matrix method is superior to the other manifold construction types, such as the Constant Covariance matrix (CC) and the Simple Manifold (SM) methods. The performance of the VC method also seems to be consistent even when lesser numbers of training images are used.

- A new incremental unsupervised-learning framework of the appearance manifold with view-dependent covariance matrix has been proposed in Chapter 5. Two learning frameworks: the supervised-learning and the incremental unsupervised-learning were used to develop a face recognition application. In the supervised-learning framework, the training (manifold construction) process was conducted only once and the trained class categories could not be updated. Meanwhile, in the incremental unsupervised-learning framework, the system first learned the initial categories through the initial manifolds, then the unlabelled image-sequences were incrementally input and learned automatically in order to update the existing identity categories of the system. The advantages of the proposed appearance manifold with view-dependent covariance matrix in the incremental unsupervised-learning framework are (1) its robustness to pose changes and degradation effects, since it uses the appearance manifold with view-dependent covariance matrix model, and (2) more realistic for real-world face recognition applications, since it allows the updating process of the initial categories. The experimental results showed that in both frameworks, the proposed appearance manifold with view-dependent covariance matrix method outperforms the simple manifold model in recognizing faces from video sequences.
6.2 Contributions

The main technical contribution of this thesis is the development and implementation of:

- The appearance manifold with view-dependent covariance matrix method.
  The proposed appearance manifold with view-dependent covariance matrix method includes the ability to capture pose variability and also learn the samples’ distribution of each pose, so that it is more accurate and robust to various degradation effects.

- The incremental unsupervised-learning framework.
  By using this framework, the system gains two advantages: (1) robustness to pose changes and degradation effects, since it uses the appearance manifold with view-dependent covariance matrix model, and (2) more realistic for real-world object/face recognition applications, since it allows updating of the initial training categories everytime a sample belonging to a new category is input.

The proposed method can be applied in a large number of real-world applications. Indoor security system (e.g. in offices, labs, etc.), target recognition in robot vision system, and automatic learning and identification system of human faces in a newspaper/photo album/video are some examples of projects that can be developed using the proposed method.

6.3 Future Work

There are many challenging problems left for further improvement of the proposed appearance manifold with view-dependent covariance matrix method. One such area is improving the method’s invariance to changes in both horizontal and vertical poses. Constructing manifold of horizontal pose and vertical pose may be useful for capturing both horizontal and vertical pose changes. However, it is likely that for large databases the manifolds will be clash into each other. From the limited experiments of this thesis, it seems that it may be best to separate the manifolds for each category (person/object).
Further works should also be done in the presence of very large databases and data acquisition process done in uncontrolled environments in order to see how the problem scales with size and real degradation effects.
Bibliography


Publications

Journals


Peer-reviewed Conferences


Other Conferences


Journal (as co-author)


Peer-reviewed Conferences (as co-author)


Tomokazu Takahashi, Lina, Ichiro Ide, Yoshito Mekada, and Hiroshi Murase, “Interpolation of high-dimensional rotation matrix and its application (in


