Face recognition in unconstrained images and videos

Nguyen, Bich Hoang Anh

2013

Nguyen, B. H. A. (2013). Face recognition in unconstrained images and videos. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/55286>

<https://doi.org/10.32657/10356/55286>

Face Recognition in Unconstrained Images and Videos

Nguyen Bich Hoang Anh

Face Recognition in Unconstrained Images and Videos

Thesis submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Engineering

by

Nguyen Bich Hoang Anh

School of Computer Engineering

Nanyang Technological University

January 2013

Abstract

In recent years, face recognition remains as one of the most attractive research topic in the computer vision field. Many methods have been proposed to deal with the large appearance changes of human face, yet it is still debatable which method works best on the unconstrained environment in ordinary photograph and video. This thesis thoroughly reviews and implements some state-of-the-art methods for image-based face recognition as well as completely studies their performance on various large scale dataset.

Motivated from the results on image datasets, a novel framework is proposed for face verification in unconstrained videos. By leveraging the pose angle information, we employ a divide and conquer approach through the following steps (a) divide the keyframes of the original video into several pose categories from extreme left to right profile face and synthesize the appearance at all missing poses (b) propose the ensemble cross-pose classifiers to recognize human faces despite pose differences by cross reference the subset of the original training data with the same pose categories. Extensive experiments on the large-scale YouTube video dataset clearly demonstrate the effectiveness and robustness to pose variation of our proposed framework.

Acknowledgments

Firstly, I would like express my utmost appreciation to my supervisor, Assoc Prof Xu Dong, for his time, support and guidance throughout the entire master programme. The journey with him has been an enjoyable and fruitful learning experiment; his ideas and guidance always keep my research in the right direction. Without his patience, I could not have completed the work in this thesis.

I would like to thank all my research project members and all colleagues in CeMNet centre, especially Mr. Zeng Zinan, Ms. Huang Yi, Mr. Li Wen and Ms. Kan Meina, for their valuable advices and discussions. I would have had many difficulties struggling on my research without their help. I am also grateful to my friend, Mr. Nguyen Duc Dung, for his various suggestions and help through reviewing this thesis.

Last but not least, I wish to thank both my beloved family and friends for their continuous encouragement and support from the beginning of my master study, making it a smooth and memorial one.

Table of Contents

Abstract.....	i
Acknowledgments.....	ii
Table of Contents.....	iii
List of Figures	v
List of Tables	vi
Introduction	1
1.1. Background	1
1.2. Challenge	2
1.3. Objective.....	3
1.4. Contributions.....	4
1.5. Thesis organization.....	5
Literature review	6
2.1. Face recognition framework	6
2.2. Feature extraction.....	8
2.2.1. Intensity	8
2.2.2. Local Binary Patterns	8
2.2.3. Gabor Wavelets.....	10
2.3. Recognition	11
2.3.1. Image recognition.....	12
2.3.2. Video recognition.....	19
2.3.3. Verification problem	21
An evaluation on Face Recognition application	23
3.1. Motivation.....	23

3.2. Implementation.....	25
3.2.1. Methods.....	25
3.2.2. Datasets	26
3.3. Result	28
3.3.1. Result on small datasets	28
3.3.2. Result on large-scale datasets.....	32
3.4. Summary	33
Face verification in Unconstrained Videos.....	34
4.1. Motivation	34
4.2. Proposed framework	36
4.2.1. Full-pose video representation	37
4.2.2. Ensemble cross-pose classifiers	40
4.3. Experimental Result.....	44
4.3.1. Dataset.....	44
4.3.2. Feature extraction	46
4.3.3. Result of individual classifier.....	47
4.3.4. Impact of each component	48
4.3.5. Performance of the unified framework	52
4.4. Summary	53
Conclusion and Future Work.....	54
5.1. Summary of current work	54
5.2. Future work	55
References	56
Appendix	60

List of Figures

Figure 1. Facial variations in images (taken from [18])	3
Figure 2. General framework for face recognition.....	7
Figure 3. Local Binary Patterns illustration.	9
Figure 4. 40 Gabor convolution images.....	11
Figure 5. The Eigenfaces and Fisherfaces from Yale database.	15
Figure 6. A comparison of PCA and LDA for a two class problem.	16
Figure 7. Overview of Sparse representation approach from [7].	17
Figure 8. Locality-Constraint Coding illustration.	18
Figure 9. The appearance variations presents in each of the 4 datasets.	27
Figure 10. An overview of our face verification scheme	37
Figure 11. (a) The original keyframe with the triangulation and (b) its transformations to 7 different pose categories.	39
Figure 12. Pseudo-code for the full-pose video representation.....	40
Figure 13. Samples of the videos in YouTube Faces dataset.....	44
Figure 14. Confusion matrix of cross-pose classifiers	47
Figure 15. ROC curve for the YTF dataset on LBP feature.	51
Figure 16. Performance comparison on the YTF dataset.	52

List of Tables

Table 1. Summarize of four datasets used in our experiments.....	28
Table 2. Performance of different methods and descriptors with respect to different parameters on the FERET dataset.....	29
Table 3. Rank-1 and rank-5 accuracy on FERET and LFW	30
Table 4. Average speed per test sample on FERET and LFW	31
Table 5. Rank-1 and rank-5 accuracy on MultiPIE and PubFig.....	32
Table 6. Average speed per test sample on MultiPIE and PubFig	32
Table 7. The statistics of pose categories.....	45
Table 8. Results on YouTube Faces dataset.....	49

Chapter 1

Introduction

1.1. Background

Face recognition has been an attractive research topic in the recent years due to its wide range of practical applications in various areas, such as law enforcement, access control or human computer interaction. With the release of cheap personal camera, powerful computer and the huge amount of face images on the Internet, face recognition has come in handy and it starts to appear in our daily life. Nowadays, face recognition application emerges in personal computer, smartphone, telecommunication, social networking and many other aspects of our lives; therefore, the demand of a secure and accurate face recognition system is higher than ever.

Face recognition systems generally fall into two categories: verification and identification. Face identification is a one-to-many problem that compares an input face image against all image templates in a face database to determine the identity of the query face, usually the label of the most similar face is returned. Examples of face identification application include CCTV surveillance, post-event analysis and suspect tracking. Meanwhile, face verification is a one-to-one problem that involves validating whether a query image belongs to the claimed identity. Thus, it can play a role as the final step of identity validation for self-serviced immigration clearance with a passport, or computer login with an account username and password.

Chapter 1: Introduction

Unlike other biometrics such as fingerprint and iris, face does not necessarily require a user's cooperation. That means a personal identification system based on face images can work effectively without the participant's knowledge. Such characteristic makes face recognition application especially desirable.

1.2.Challenge

Despite many compelling reasons to use face recognition, building such system with high accuracy encounters a variety of problems. They are high dimensional space, lack of training data, noisy input, and appearance variations.

The biggest challenge for face recognition comes from the huge appearance variations of face photograph, due to changes in human pose, facial expression, age and capturing conditions such as lighting, occlusions or motion. Indeed, the face variation problem is very difficult to resolve, especially with images and videos captured in totally uncontrolled environment, for instance, images from the Web or surveillance camera recording. It has been observed that the variations due to expression, illumination and viewing direction of images of the same face are often larger than variations from the change in face identity [25]. Figure 1 illustrates how different a person might look like when light source direction, age and viewpoint vary.

Over the past decade, many public face datasets have been created for researchers to improve the recognition rate and make current methods comparable. Many previous works have been proven to work well in controlled environments as evaluated on some public databases, such as ORL [8], AR [9], PIE [10] and FERET [11]. However, when applied to unconstrained images, the large appearance variations due to pose, expression, lighting condition, and occlusion prevent those methods from achieving such good results.

Chapter 1: Introduction

*Figure 1. Facial variations in images (taken from [18])
 (b) noise, (c) expression, (d) illumination, (e) pose, and (f) ageing.*

This problem is even worse in video, since many real-world videos are captured by amateurs. They also suffer from poor lighting conditions, large variations of poses, frequent occlusions, and motion blur. Moreover, limitations in terms of storage and capturing devices may result in low resolution and compression artifacts, making video recognition task challenging even for human.

1.3.Objective

There are several desired characteristics for a face recognition system, including high accuracy, low false positive rate, robustness to occlusion, rapid processing and low storage, simplicity, etc. In the scope of this thesis, we focus on the following two objectives: the ability to generalize to unconstrained environment and the practicability in real application.

The main objective is being able to recognize human faces under completely unconstrained environments. Although having decades of history with promising result on the controlled environment, face recognition in unconstrained environment still remains unsolved for the most part. The reason being unconstrained images and

Chapter 1: Introduction

videos usually expose wide range of pose and lighting conditions as addressed in the previous part. Thus our priority is to develop methods that are robust to the intra-personal variance in unconstrained images and videos.

The second objective is to put face recognition into practice, in which the system can achieve both excellent accuracy and speed while remains as simple as possible. Ideally, the face recognition component should make prompt decision within seconds since longer response time is intolerable for human-computer interaction. In practice, there is often a trade-off between accuracy and speed. Thus, the objective here is to balance between the performance and computational time.

1.4. Contributions

The major contributions of this thesis can be summarized as follows:

- First, we review various existing face recognition methods in the literature. We implement some popular approaches on both features and learning methods and extensively test these implementations with both controlled and unconstrained datasets. We present our findings regarding suitable feature and learning method for a face recognition application.
- Knowing the strengths and weaknesses of each feature and learning method, we proceed to tackle the problem of face recognition in unconstrained videos. Specifically, we exploit the pose information and propose a novel pose-robust framework for video-based face verification. Our main contributions include: (a) the full-pose video representation by using sample frames for all possible viewpoints which helps to capture the face variations from multiple angles, and (b) the novel ensemble cross-pose classifiers to facilitate the recognition task across all pose categories.

1.5. Thesis organization

The structure of this thesis is as follows.

In Chapter 2, we first review the framework of face recognition task and move forward to describe popular approaches to the face recognition problem.

In Chapter 3, all the methods are evaluated on various controlled as well as uncontrolled datasets. Our goal is to enumerate descriptors and algorithms and find out which one among them performs well despite the large appearance variations in real application.

Motivated from the result of image-based face recognition, chapter 4 introduces the problem of face verification in videos and describes our proposed framework to tackle this problem. We then present and discuss our result on the large-scale video database collected from the YouTube website.

Finally, in Chapter 5, we summarize the works in this thesis and propose some future directions.

Chapter 2

Literature review

In the previous chapter, we have discussed the face recognition problem and motivation on resolving the difficulties in uncontrolled environment. In this chapter, we present a general framework for face recognition systems. We focus on two main stages for face recognition task, namely feature extraction and recognition. Representative methods of each stage are reviewed in details, whilst their performances will be presented in the following chapter. The chapter also captures the essence of the face verification problem.

2.1.Face recognition framework

Many works have been done towards developing a face recognition system. In general, the system consists of four stages: face localization, face normalization, feature extraction and recognition. These stages are described as follows:

- *Face localization*: In the first stage, the face-bounding box is detected from input image or video frame. Since there are multiple frames in a video, a face-tracking component is usually used to track the face across these frames. The face region is then sent to a landmark localization module that locates the face components including eyes, nose and mouth.
- *Face normalization*: This stage is necessary since a face recognition system should be able to recognize face images with varying pose and lighting condition. After the face component information has been extracted from the

Chapter 2: Literature Review

image, the face region will be aligned based on these components. Some face recognition systems use face warping or face morphing to lessen the pose variance. The face is then normalized based on illumination and gray scale information to a standard lighting condition. The result will also be scaled to a standard size.

- *Feature extraction*: The next stage involves extracting the facial feature from the normalized face. The feature descriptor from this stage should be robust to the appearance variations so as to easily distinguish faces of different persons.
- *Recognition*: In this final stage, a distance metric between two faces is learnt for comparing facial features extracted from the previous stage. In the case of face verification, the matcher can directly answer Yes or No for the two input images. For face identification task, the system returns the most similar identity from the enrolled faces in the database. The challenge in this stage is to find such a distance metric that yields high accuracy.

Figure 2. General framework for face recognition

Figure 2 shows the general framework for face recognition. Among all these components, the feature extraction and recognition steps are the most important and they greatly contribute to the final performance of the system. Hence, the majority of works in the literature focus on these two directions: feature-oriented and

Chapter 2: Literature Review

recognition-oriented approaches. The former research direction aims to extract robust and effective features for face representation, while the latter focuses on the learning and face similarity computation. In the following sections, we will review and discuss in details these two topics.

2.2. Feature extraction

Many feature extraction methods have been proposed for face recognition in order to provide a robust representation of face image. In this section, three of the most widely used features are presented: Intensity, Local Binary Patterns and Gabor Wavelets.

2.2.1. Intensity

The gray-scale face image is normally stored in a 2D matrix of size $[image_height \times image_width]$. The intensity descriptor directly makes use of the gray-level and concatenates all the pixels value in the 2D array to a single vector. The dimension of the intensity feature is thus equals to the number of pixels in the original face image. Being the simplest representation, the descriptor has significant drawbacks such as it does not capture the edge, shape or variance and thus extremely sensitive to the lighting, pose and expression.

2.2.2. Local Binary Patterns

Local Binary Patterns (LBP) [12] is one of the most popular features for face representation. Despite its simplicity and compactness, LBP always achieves top performance in the recognition task. Given the face region, the LBP value for each pixel is constructed by thresholding the value of its 3x3 neighborhood against itself, the result is represented as a binary number (0 for smaller than the threshold value,

Chapter 2: Literature Review

and 1 otherwise) of length equals to the number of neighborhood pixels. The example in Figure 3 illustrates how LBP value is calculated. The LBP filter normally comes with a parameter to indicate the size of the neighborhood. While it is certainly interesting to calculate the LBP value and/or plot them out, normally for face descriptor the whole face is divided into smaller regions and the histogram of LBP code is computed for each part.

Figure 3. Local Binary Patterns illustration.

There are also few other extensions that help to improve the original operator, a useful one among them is uniform pattern [12]. Motivated from the fact that some LBP patterns appear much more often than the others, in this extension LBP codes are divided into 2 types: uniform and non-uniform. The uniform patterns are those that have at most 2 transitions in the binary code. Statistically they appear frequently in the LBP code as opposed to the non-uniform patterns due to the fact that neighborhood pixels often have similar values. Therefore, each uniform pattern is represented by a separated label and all the non-uniform patterns share a common label. For example, when using the 8-surrounding neighborhood for thresholding, there are a total of $2^8 = 256$ patterns, in which 58 among them are uniform hence totally there are 59 different labels.

2.2.3. Gabor Wavelets

Gabor is a well-known descriptor for face recognition task, although Gabor filters are initially used for edge detection [6]. Gabor features are constructed by applying convolution operator on an input image and multiple Gabor filters of different scales and orientations. The Gabor filter is defined as:

$$g(x, y) = \exp \left(-\frac{x'^2 + \gamma y'^2}{2\sigma^2} \right) \exp (j2\pi f_0 x')$$

where $x' = x \cos \theta + y \sin \theta$

and $y' = -x \sin \theta + y \cos \theta$

There are two important parameters involved in the construction of Gabor features, which are scale (σ) and rotation (θ). The dimension of Gabor feature thus equals to the original image size times number of different scale and rotation combinations used in the convolution.

Figure 4 shows the convolution results of a sample face image with 40 Gabor filter. Although being one of the best features for face recognition, Gabor has a huge time and space trade-off since the feature itself having much higher dimension than the original face image. For example, if 5 different scales and 8 rotation values are used in the face recognition system, a total of 40 Gabor filters will be used for each image. Hence, an image of size 80 x 80 will have a Gabor feature vector of 80 x 80 x 40 = 256,000 dimensions.

Chapter 2: Literature Review

Figure 4. 40 Gabor convolution images.

Apart from Intensity, LBP and Gabor, there are other widely used features for face representation, such as SIFT [13]. More sophisticated method has been successfully applied to the face recognition problem and shown promising result. For instance, Wolf et al. [26] proposed an improved patch-based LBP to encode the similarities between neighbouring patches in addition to neighbouring pixels. In [27], Pinto et al. analyzed the usage of biologically-inspired visual representations to model various stages of visual cortex in the brain. Moreover, Kumar et al. [15] used the similarity of facial components to a reference set of known persons as features. The advanced methods help to improve the accuracy with huge time and computational trade-off. Thus, the choice of suitable features highly depends on the face recognition system requirements.

2.3. Recognition

The recognition stage makes use of the feature extracted from the face region to predict the human identity or authenticate whether the photo/video comes from the claimed identity in the verification case. In this section, we present some of the

Chapter 2: Literature Review

most efficient methods in image and video recognition. Subsequently, we define the face verification problem which is the main focus of the thesis.

2.3.1. Image recognition

The target of the image recognition stage is learning a classifier that is robust to pose, lighting condition and expression while keeping the person identity. Traditionally, subspace analysis methods such as Eigenfaces and Fisherfaces have been widely use for the image recognition task due to their simplicity yet promising performance. Recently, Wright et al. [7] has presented robust face recognition via sparse representation and the method has received significant attention due to its robustness to noises, occlusions and corruptions.

Eigenfaces

The Eigenfaces method was proposed by M. Turk and A. Pentland [3] as the first application of Principal Component Analysis (PCA) [28] in face recognition. The main idea is to extract a set of basis vectors by performing PCA on the set of training face images. These basis vectors have the same dimension as the input images, and are called eigenfaces. With these standard eigenfaces, any face in the database can be represented by a combination of them. The corresponding multipliers from the reconstruction will be the feature descriptor for the face image.

More formally, for the set of n images denoted as $x_1, x_2, \dots, x_n$ in d dimensional space. The objective is to find an optimal linear projection w that preserves as much information as possible. After the projection, the new feature vectors y is calculated as:

$$y_k = w^T x_k$$

Chapter 2: Literature Review

The total scatter matrix S_T is defined as:

$$S_T = \sum_{k=1}^n (x_k - \mu)(x_k - \mu)^T$$

where μ is the mean face defined as $\mu = \overline{x_k}$. With the above formulation, the scatter matrix after performing the linear transformation is $w^T S_T w$. In PCA, the optimal projection dimensions w_{opt} is chosen to maximize the determinant of the resulting scatter matrix:

$$w_{opt} = \arg \max_w |w^T S_T w|$$

Solving the equation, w_{opt} is the set of m eigenvectors (or eigenfaces) corresponding to the m largest eigenvalues. PCA has several advantages as it only involves a single eigen decomposition step which is therefore fast to compute. Moreover, PCA is very good for dimension reduction because it preserves a large amount of information in a small number of dimensions. Besides, it is worth noting that PCA is an unsupervised learning method, so no label information is required.

The limitation is that PCA is not designed to solve the classification problem. While it maximizes the variations in projected dimensions, it is unclear these variations represent the difference of identities or just the difference of the same person.

Fisherfaces

Fisherfaces is a supervised method introduced by Belhumeur et al. [4], which makes use of label information available at the training time to improve the discriminative ability of the linear projection. The idea is to find an optimal projection so that the face images of the same person will be close together in the new subspace, and face

Chapter 2: Literature Review

images of different person will be far from each other. The method is a derivative from Linear Discriminant Analysis (LDA).

With similar notation of x , y , and each image belongs to one of c class labels X_i , the author defined the between-class and the within-class scatter matrix as:

$$S_B = \sum_{i=1}^c n_i (\mu_i - \mu)(\mu_i - \mu)^T$$

$$S_W = \sum_{i=1}^c \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

where μ_i is the mean image of class X_i , and n_i is the number of images in class X_i . In LDA, the optimal projection w_{opt} is chosen so as to minimize the determinant of the projected between-class scatter matrix and maximize the determinant of resulting within-class scatter matrix:

$$w_{opt} = \arg \max_w \frac{|w^T S_B w|}{|w^T S_W w|}$$

w_{opt} is solved by taking the generalized eigenvectors of S_B and S_W corresponding to the m largest eigenvalues:

$$S_B w_i = \lambda_i S_W w_i$$

Similarly to PCA, w_{opt} is the set of m eigenvectors corresponding to the m largest eigenvalues. These m eigenvectors are called Fisherfaces that can represent any face in the database. Figure 5 shows the Eigenfaces and Fisherfaces calculated from the face images in the Yale database [29].

Chapter 2: Literature Review

Figure 5. The Eigenfaces (row 2) and Fisherfaces (row3) calculated from face images (row 1) in the Yale database.

In fact, the dimension of facial features are normally much higher than the number of images in the training set, thus using the original feature result in S_W is always singular. To overcome that problem, normally PCA is performed before LDA to project the image set to a lower dimensional space so that S_W is non-singular, while keeping as much variance in the data as possible to benefit the recognition task.

Figure 6 shows the main difference of PCA and LDA in finding the optimal linear projection. In the figure, the samples are in 2D feature space, and both PCA and LDA are used to project these points from 2D to 1D space. It can be seen that PCA actually tries to maximize the total scatter, making the samples no longer linearly separable in the projected space. On the other hand, LDA achieves greater between-class scatter, and, therefore, much simplifies the classification task.

Figure 6. A comparison of PCA and LDA for a two class problem.
(taken from [4])

Sparse based methods

Sparse method for face recognition is first proposed by Wright et al [7]. The method has been proven to be more robust to occlusions and corruptions found in real-world faces. The author argued that although images (or their features) are naturally very high dimensions, they usually exhibit degenerated structure if they belong to the same class. Thus an over-complete dictionary can be used to represent any samples and through seeking the sparsest representation automatically discriminates different classes in the training set.

Chapter 2: Literature Review

Figure 7. Overview of Sparse representation approach from [7].

Each test image is represented as a sparse linear combination of all the training images plus sparse errors.

Lets denote the set of training images as $Y = \{Y_i |_{i=1}^n\}$. The traditional state-of-the-art sparse coding has the following mathematical representation:

$$E = \min \sum_{i=1}^N \frac{1}{2} \|Y_i - BS_i\|^2 + \lambda \|S_i\|_0$$

where S_i is the code for input image Y_i ; B is the codebook used to encode the images. The first term of the equation, the reconstruction error, measures how well the information in original image space is preserved (usually it uses the Euclidean norm), while the sparseness in the second term penalizes more for the coefficients that are spread over all the space than the ones with only a few. The original problem is NP-hard, hence it is relaxed to the following formulation using l_1 -norm

$$E = \min \sum_{i=1}^N \frac{1}{2} \|Y_i - BS_i\|^2 + \lambda \|S_i\|_1$$

The above objective function can be effectively solved by actively updating B by fixing S and updating S by fixing B . Assume that the training data having class label

Chapter 2: Literature Review

$Y[j] = \{Y_{j1}, Y_{j2}, \dots, Y_{jk}\}$ encoded to values $S[j] = \{S_{j1}, S_{j2}, \dots, S_{jk}\}$ while the dictionary corresponded to class j is $D[j]$. For the new testing sample X , it will be classified to the class:

$$\text{class}(X) = \arg \min_j \|X - B[j]S[j]\|$$

Figure 8. Locality-Constraint Coding illustration.

The figure on the left represents the traditional linear sparse representation while the figure on the right enforces the reconstruction should be based on the neighbors

In [16], the author proposed the Locality-constrained Linear Coding (LLC) to improve the coding scheme by using the locality between features rather than the sparsity. It is desirable to have the locality preservation, which means the similar datum should have similar reconstruction code, and hence we can directly measure the similarity using only the coefficient. The author argued that the locality feature is more essential than the sparsity as locality lead to sparsity but not the other way around, and it also retains the local information during the reconstruction process.

With the locality is measured by distance, the following objective function is used in [16] to compute the dictionary B and reconstruction code of the training data:

Chapter 2: Literature Review

$$E = \min \sum_{i=1}^N \frac{1}{2} \|Y_i - BS_i\|^2 + \lambda \|D_i \odot S_i\|^2$$

$$s.t. \mathbf{1}^T S_i = 1$$

where D is proportional to the distance measures and $\odot$ denotes the element-wise multiplication. Solving the equation and classifying test samples is similar to the standard sparse representation.

2.3.2. Video recognition

There are 2 main directions when dealing with the video recognition. The first one considers videos as an unordered bag of frames and thus uses set-to-set matching approach. The second approach makes use of the temporal dependencies to build an appearance model.

Set-to-Set Matching

Video based face recognition can be seen as the problem of finding the set-to-set similarities if one video is represented as a set of images. Thus, recognition becomes a problem of determining the similarity between vector sets, which can be modeled as distributions [30], subspaces [31], or more general manifolds [32][33]. Different choices of similarity measures are then used to compare two sets.

One straightforward solution is to use image-based face recognition for each frame of the video. Each individual frame in the video provides a decision score regarding the similarity of the person to known subjects in the database. Appropriate fusion techniques can then be applied to give the final decision, such as majority voting. In [23], the author utilized the 2D image-recognition to find the distance matrix D between every pair of frames in two video. After that, the minimum, mean or max

Chapter 2: Literature Review

value of the distance matrix represents the distance between two input videos. In [34], Park et al. adopted three different matchers for frame-level recognition and use the sum rule with proper normalization to fuse results of individual frames.

Another popular approach in this direction tries to learn the “exemplars”, which are selected representatives from the original video. The exemplars are used to summarize the gallery video information. Following this direction, Wolf et al. [23] use the most frontal pose frame to represent the whole video since the frontal pose carries the most information about the person identity. In [35], the author used k-means clustering and selected the cluster centers as exemplars, while more sophisticated methods to learn the exemplars are presented in [36].

Following the vector set representation approach; Wang et al. [32] extended Subspace-Subspace Distance to Manifold-Manifold Distance which can be directly applied to video recognition. Most recently, Cui et al. [33] proposed an image sets alignment method to facilitate video-based face recognition. The method employs a well-aligned and pre-structured reference set so as the problem of matching two videos is converted to aligning each video to the reference set and then matching only corresponded local models.

Appearance modeling

Previous studies have showed that temporal information and video dynamics also aids in recognition of faces besides the 2D appearance at each frame [37]. Therefore, various works have extensively used the spatio-temporal information and dynamic structure of the video when modeling the video sequence. For instance, the temporal and face motion have been encoded in the learned of Hidden Markov Models [38] or joint probabilistic model [39].

Chapter 2: Literature Review

However, temporal model based approaches suffer from some drawbacks that prevent them from achieving their full potentials, Current limitations include the formulation normally use global features while ignoring local information. Besides, the lack of discriminative cues between the facial dynamics leads to difficulties in recognizing different persons.

2.3.3. Verification problem

Face verification can be considered as a special case of image recognition, in which we are only interested in 1:1 matching instead of 1:N matching. In face verification, the labels of training data remain unknown, however the training images or videos is divided into pairs in which the side information is known (i.e. whether the pairs come from the same person); normally there are a balanced number of same and different pairs. In the testing stage, given a pair of images or videos, the system needs to decide if both of them refer to the same identity. The lack of subject label in verification problem makes conventional methods fail to learn a discriminative similarity metric, thus affecting the performance.

The most challenging issues in both image and video verification are how to effectively utilize the side-information and how to balance the trade-off between the discriminant ability and intra-person variance due to different capture settings. To address the first problem, several researchers have specifically adjusted the formulation of popular classifiers such as Support Vector Machine [40] or Linear Discriminant Analysis [41] so that they can work well with only the weak label. Some also includes the use of a separated reference set so that the additional labels can help during learning stage [21].

To tackle the second concern regarding the difference in pose and lighting conditions in the given pair of faces, alignment methods have been used to translate

Chapter 2: Literature Review

the face appearance to a standard setting before calculating their similarities. Yin et al. [21] proposed the "associate-predict" model built on a prior memory set of multiple identities under large intra-personal variation. For a new input face, the memory set is used as a bridge to predict how the person might look like in another setting. Following a similar track, Berg and Belhumeur [42] presented a novel identity preserving alignment by detecting the fiducial points and warping them to the correct locations thus produces the frontal pose and neutral expression image. Most recently, Cui et al. [33] proposed an image sets alignment method to facilitate video-based face recognition through aligning each video keyframe to a full-pose reference set.

Chapter 3

An evaluation on Face Recognition application

In this section, we conduct extensive experiments on popular features and learning methods in both controlled and unconstrained environments. The aim is to simulate how well face recognition applications perform when finding the identity of the person in various real-world settings. Thereafter, detailed discussions on the practicability of each feature and learning method are presented.

3.1.Motivation

Although many existing methods have attempted to tackle face recognition task in constrained environment, there is little knowledge on how well these algorithms work in regular and realistic scenario; such as photographs collected from the Web, personal photos taken from Facebook profile, photographs taken using handphone, etc. That means there is still a gap between research's result in papers and real-world application. Therefore, it would be useful to understand the characteristics of each recognition method, including its accuracy, computational time and robustness on images in real world. Such topic has not been well studied in the literature.

Furthermore, though there are various attempts to simulate the person variance in the controlled environment, each dataset still has certain bias that affects the performance evaluation. MultiPIE dataset [17] is one of the successful attempts that

Chapter 3: A survey on Face Recognition application

try to capture as much variance as possible through taking photographs of cooperative users under different poses, varying expression and lighting conditions. However, the dataset also has its own drawbacks, for example the pose difference is simulated through cameras from different viewpoints thus the expression and lighting remain unchanged. In addition, the dataset contains lots of photos under extreme viewpoints (i.e. the most left and the most right profile face), extreme expression (i.e. scream or disgust) and intense lighting conditions (i.e. no ambient light), which is not normally presented in regular images. Thus the reported result in controlled environment is still far from real performance in unconstrained setting.

Besides, in contrast to the tremendous effort to build large-scale face databases, there is not a standard methodology to evaluate performance. Although there is a significant number of works in the literature, each one normally tests on only few datasets, or even subset of a dataset having one specific characteristic. Among the current datasets, FERET [11] and LFW [18] are known for their great contribution towards a standardization of testing methodology.

Therefore, in this work, we address the problem of face recognition from an application point of view. The requirements of such systems include: (a) high accuracy in normal settings where the face is clearly visible with slight head rotation and some expressions, (b) effectiveness in terms of the time taken for feature extraction and learning step in order to make the final decision. Our objective is to make a fair comparison between different methods using common standard datasets.

3.2. Implementation

3.2.1. Methods

In the first step of the pipeline, the faces are automatically detected with OpenCV [19] and aligned by fixing the eyes position extracted from a landmark localization module [20]. The size of the face region is cropped to 150×80 at the pre-processing stage.

In order to determine the effectiveness of each feature against unconstrained face application, we use three types of features as Intensity, LBP and Gabor to test the performance of the face recognition system.

- For Intensity feature, we simply concatenate all the pixel values to a single vector of the size 12000-D.
- For LBP features, the face region is first divided into non-overlap blocks of the size 10×10 in which a histogram of 59 bins is extracted for each sub-region and concatenated to form one single 7,080-D vector.
- For Gabor features, we form 40 filters using 5 scales and 8 orientations. After performing convolution operation on each of them with the original face image, we further down-sample the high dimensional resultant images using a scaling factor of 5×5 . The dimension of Gabor feature is thus equal to 19,200.

We also employ three popular learning methods as Eigenfaces (PCA), Fisherfaces (LDA) and sparse representation (in particular, LLC). The choice of these classifiers is due to their effectiveness, efficiency and simplicity. Since we are

Chapter 3: A survey on Face Recognition application

targeting the real application, the classifier does not need to be too complicated and should be able to produce the decision within seconds.

Previous works have shown that the cosine similarity is an effective metric for face recognition [43]. Thus, after the projections in Eigenfaces and Fisherfaces we use the cosine similarity to measure the likeliness of different feature vector. The cosine similarity between feature vector A and B is defined as:

$$\text{similarity}(A, B) = \frac{AB}{\|A\|\|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}$$

For the subspace methods, after the projection, each class will form a cluster in the new subspace. We calculate the cluster center and use it to represent the class. For a new test sample, we assign the candidate class labels to it by searching for closest clusters. As for the locality-constrained linear coding, the classification is discussed in Chapter 2.

Generally, only the label of the closest subject is assigned to the test sample. However, as the first choice may not be good enough, face recognition system can show the top candidates and let the users decide which one is the correct person. Hence, we adopt two performance evaluation methodologies as *Rank-1* and *Rank-5*. *Rank-1* indicates that the top candidate is the correct subject, while *Rank-5* indicates that the correct subject is among the top five candidates.

3.2.2. Datasets

To provide a thorough evaluation of our popular methods on face recognition, we carried out all experiments on four popular datasets: FERET [11], MultiPIE [17], Label Face in the Wild (LFW) [18] and Public Figure (PubFig) [15]. The first two, FERET and MultiPIE, are captured in controlled environment in the laboratory;

Chapter 3: A survey on Face Recognition application

however, the authors set up the environment change through simulating as much pose, lighting and expression changes as possible. The remaining two datasets, LFW and Public Figure, is collected from the Web by searching for celebrities' photographs. The aim of this part is to give an introduction to the design of these datasets together with their evaluation methodology, i.e., how they can be used to evaluate face recognition algorithms.

Figure 9. The appearance variations presents in each of the 4 datasets.

The left column show the face images captured in the controlled environment in FERET and MultiPIE while the right column shows the photos from LFW and PubFig.

FERET and MultiPIE are somehow similar in the sense that they are captured under the controlled environment in the lab; however the datasets also simulates a wide range of variations in pose, expression and lighting under multiple capturing sessions. Since the goal of this experiment is to test the performance of different methods on conventional case; hence, we have cut off the extreme poses and lighting condition of MultiPIE and FERET dataset. Only the poses between -25° to 25° are kept, and only neutral or smiling expression are used, which is a reasonable assumption for making the whole face visible.

Meanwhile, LFW and PubFig are collected from querying the photos of celebrities on the image search website; therefore, the condition of the photo is less extreme and no pruning is required on those dataset. However, due to the intention of

Chapter 3: A survey on Face Recognition application

collecting LFW for verification task, many persons have only a few images, we have removed these people with less than 10 photos so that all the method would have sufficient training data. The size of each dataset after pruning and the train/test split size are summarized in below table.

Dataset	No. persons	Training samples per person	No. training images	No. testing images
FERET	327	4	1308	716
LFW	155	7	1085	3167
PubFig	200	10	2000	37599
MultiPIE	337	10	3370	26097

Table 1. Summary of four datasets used in our experiments.

We rerun our experiments 10 times for each method to evaluate the performance on each dataset; each split contains the whole dataset divided into different train and test set. In each split, we randomly select 4 to 10 photos per person for training and the rest for testing the performance. FERET and LFW are relatively smaller; hence, they are used for model selection and parameter tuning.

3.3. Result

3.3.1.Result on small datasets

We conduct experiments on all methods and all feature types on the small datasets for both performance evaluation and parameter selection. The configurations of our classifiers and different parameters are as follow:

- For the Eigenfaces method, there is one parameter as the number of dimensions after the projection, which corresponds to the preserved energy level. Descriptors are typically compressed to a few hundred dimensions using the PCA technique, thus we varies this parameter from 100~1000.

Chapter 3: A survey on Face Recognition application

- In the Fisherfaces method, with the intention of reducing feature noises, descriptors are first compressed to 98% energy level using the PCA technique. LDA is then used to project the feature vectors to a lower dimensional space. Since there are around 300 subjects in each dataset, we vary the number of kept dimensions from 100~300 similar to Eigenfaces.
- There is one parameter in the Locality-constraint Linear Coding as λ . In our experiments, λ is set to values from 0.1 to 0.9.

The recognition rate and running time for each method on the FERET dataset with respect to different parameters are summarized in Table 2, more details can be found in the Appendix A.

Descriptor & Method	Parameter	Rank-1 Result	Rank-5 Result	Time/Sample
LBP + PCA	100	67.32 ± 2.39	83.94 ± 0.88	0.0090
	300	72.68 ± 1.90	86.76 ± 0.65	0.0394
	500	73.74 ± 2.01	87.32 ± 0.57	0.0926
	700	73.99 ± 1.97	87.35 ± 0.56	0.1767
	1000	73.94 ± 1.98	87.37 ± 0.61	0.8468
LBP + LDA	100	91.54 ± 0.86	98.10 ± 0.61	0.0063
	150	93.74 ± 0.98	98.32 ± 0.76	0.0200
	200	94.25 ± 0.91	98.58 ± 0.61	0.0336
	250	94.33 ± 1.08	98.58 ± 0.52	0.0436
	300	94.33 ± 0.85	98.55 ± 0.52	0.0532
LBP + LLC	0.1	93.24 ± 1.16	96.65 ± 1.10	0.3169
	0.3	94.30 ± 1.65	97.77 ± 0.72	0.3166
	0.5	94.33 ± 1.43	97.77 ± 0.54	0.4058
	0.7	94.50 ± 1.29	97.88 ± 0.51	0.6503
	0.9	94.55 ± 1.14	97.79 ± 0.43	0.7244

Table 2. Performance of different methods and descriptors with respect to different parameters on the FERET dataset.

Chapter 3: A survey on Face Recognition application

From Table 2, Eigenfaces achieve the best result when keeping 700 and above dimensions; meanwhile, Fisherfaces achieve good result with the dimension ranging from 200 to 300. Although keeping more dimensions leads to a rise of testing time, the time increment from 100 dimensions to 700 dimensions is only less than 0.2s per sample, which is insignificant compared to the preprocessing time of face detection and feature extraction. On the other hand, LLC achieves good result with λ in the range from 0.7 to 0.9.

Setting the parameters to their optimal value, we compare the result of different learning methods and features. Table 3 summarizes the result of different methods using Intensity, LBP and Gabor descriptors on the four datasets.

Descriptor & Method	FERET	LFW
Rank 1 result		
Intensity + PCA	05.8 ± 1.2	03.6 ± 0.4
Intensity + LDA	90.2 ± 0.3	41.5 ± 2.6
Intensity + LLC	91.9 ± 0.8	44.2 ± 2.7
LBP + PCA	73.9 ± 2.0	16.7 ± 2.0
LBP + LDA	94.3 ± 0.8	58.1 ± 2.0
LBP + LLC	94.6 ± 1.1	58.3 ± 1.6
Gabor + PCA	64.0 ± 1.9	27.1 ± 2.7
Gabor + LDA	97.3 ± 0.3	69.9 ± 1.8
Gabor + LLC	97.5 ± 0.9	72.0 ± 2.7
Rank 5 result		
Intensity + PCA	16.7 ± 1.0	10.2 ± 1.1
Intensity + LDA	97.8 ± 0.9	65.9 ± 2.8
Intensity + LLC	96.6 ± 0.6	66.2 ± 3.0
LBP + PCA	87.4 ± 0.6	35.5 ± 3.0
LBP + LDA	98.5 ± 0.5	77.0 ± 1.8
LBP + LLC	97.9 ± 0.4	76.1 ± 1.9
Gabor + PCA	81.1 ± 1.7	47.4 ± 1.9
Gabor + LDA	99.3 ± 0.1	85.0 ± 1.1
Gabor + LLC	99.0 ± 0.4	86.1 ± 1.8

Table 3. Rank-1 and rank-5 accuracy on FERET and LFW

Chapter 3: A survey on Face Recognition application

Descriptor & Method	FERET	LFW
Intensity + PCA	0.1836	0.0005
Intensity + LDA	0.0467	0.0059
Intensity + LLC	1.3499	0.3561
LBP + PCA	0.1767	0.0005
LBP + LDA	0.0532	0.0059
LBP + LLC	0.7244	0.3561
Gabor + PCA	0.1879	0.0206
Gabor + LDA	0.0746	0.0206
Gabor + LLC	1.7217	1.0536

Table 4. Average speed per test sample on FERET and LFW

On the feature side, the results clearly show that Gabor descriptor generally performs much better than LBP. The error rate is reduced by 20~50% using Gabor feature, however the high accuracy comes with huge memory and time trade-off. Although LBP does not give as good result as Gabor, it is noticeable that the LBP is much more lightweight and faster. Gabor takes up to 100 times longer than LBP to extract descriptor from the same face image, and even with the 5x5 down-sampling factor, Gabor feature is in 19,200-D space while the dimension of LBP is at 7,080-D.

Local Binary Patterns is significantly better than Intensity although LBP is more compact. This is easily explained since Intensity feature is highly sensitive to the intra-personal variance, while the LBP value is the result of comparison to each neighborhood pixel value. Thus it reduces the effect of lighting. Besides, the histogram operation helps to reduce the pose and expression variance.

With regard to the learning methods, the result indicates that Eigenfaces unsurprisingly yields much lower result than the other two, since PCA does not take advantage of the class label information. In unconstrained environment such as LFW, Eigenfaces only succeeds to identify the correct label in 16.7% of the time while other methods can achieve up to 70%. The LLC and LDA produce almost the same result with small differences in terms of accuracy.

Chapter 3: A survey on Face Recognition application

Among the 3 methods that we tested, PCA and LDA are much faster since they only involve a simple linear projection at the testing stage. Both methods take less than 0.1s for a test sample. LLC, on the other hand, consumes a lot more computational power. When the number of images and persons in the dataset scales up, the speed of LLC increased significantly, making it nearly impossible to test the performance of LLC on large-scale dataset.

3.3.2.Result on large-scale datasets

Because of the low accuracy of Eigenfaces and the time constraints in the real application, only the result of Fisherfaces method with different type of features is reported for large-scale datasets.

Descriptor & Method	MultiPIE	PubFig
Rank 1 result		
Intensity + LDA	83.8 ± 1.9	35.3 ± 1.3
LBP + LDA	96.8 ± 0.3	47.9 ± 1.3
Gabor + LDA	98.8 ± 0.1	64.1 ± 0.4
Rank 5 result		
Intensity + LDA	97.1 ± 0.2	55.5 ± 0.8
LBP + LDA	99.4 ± 0.1	67.2 ± 0.8
Gabor + LDA	99.7 ± 0.1	78.2 ± 0.4

Table 5. Rank-1 and rank-5 accuracy on MultiPIE and PubFig

Descriptor & Method	MultiPIE	PubFig
Intensity + LDA	0.0252	0.0229
LBP + LDA	0.0175	0.0397
Gabor + LDA	0.0243	0.0492

Table 6. Average speed per test sample on MultiPIE and PubFig

It is interesting to see that although FERET and MultiPIE have tried hard in simulating the environment including pose, expression and lighting to capture as much variance as possible, there is still a huge gap between the result on unconstrained and constrained environment. The method above in controlled environment shows almost perfect result with more than 97% of the correctly

Chapter 3: A survey on Face Recognition application

classified image returning as the top candidate. However, in the unconstrained images, the same method can only reach less than 80% accuracy. This is due to the fact that the photographs from the Internet show a greater degree of appearance changes, such as lively expression, hair and occlusions... Each photo also shows the person in a different pose, and there is hardly any repetition.

The speed of LDA on different feature types is fast. In particular, it takes less than 0.05s for each sample. In addition, the speed does not change much from small dataset to large dataset. It is because LDA method only needs to perform one linear projection and calculate the distance to the class centers.

3.4. Summary

We have presented a standard evaluation on image-based face recognition from the application point of view. In previous sections, many strategies have been analyzed, showing that almost all methods and features present acceptable recognition rates and speed when tested on the small datasets. However, on the large dataset, the combination of Fisherfaces and Gabor feature demonstrates a leading role in terms of accuracy. This comes with the tradeoff as Gabor feature takes much longer time to be extracted from the face image and consumes more storage. Therefore, Fisherfaces with LBP descriptor is the most suitable method for a real-world face recognition application. With our extensive experiments, we can see that face recognition application can achieve relatively high result using simple methods and features.

Chapter 4

Face verification in Unconstrained Videos

In the previous section, we have examined the result of various face descriptors and learning methods for image recognition task. However, in the video setting and especially in the unconstrained environment, this is somewhat different since each video normally contains one person in variable poses, expressions and lighting conditions. The videos also suffer from motion blur as well as low resolution due to storage restriction. In this section, we will investigate the problems of current video-based face recognition techniques and propose a new framework for recognizing unconstrained videos.

4.1. Motivation

With the rapid adoption of surveillance systems, digital cameras and handphones, the face recognition problem has been widely studied over the last decades. Generally, face recognition applications fall into two categories: face identification which aims to identify the subject of each face, and face verification which aims to decide whether a pair of faces are from the same subject. While many face recognition systems have shown promising results under restricted environments with cooperative users, face recognition under unconstrained environments remains challenging due to large variations in poses, lighting conditions, expressions, and so on.

Chapter 4: Face verification in unconstrained videos

Many works have studied the image-based face identification [44][26] and face verification [21][15][42][43] problems under unconstrained environments. However, for the video-based face recognition, previous works mainly focus on the face identification problem [38][34], only few works [24] have been proposed for the face verification under unconstrained environments.

Under unconstrained settings, face recognition in videos is much more challenging than that in images because videos usually contain more intra-class variations in terms of poses, expressions and lighting conditions. Among all these difficulties, pose variation is considered to be most challenging as pointed out in previous works [45][46]. Different videos may contain different poses and the number of keyframes for each may be different as well. As a result, the two videos of different persons with a overlapped set of poses may appear more similar, when compared with the videos of captured under completely different viewpoints. Hence, using traditional methods to directly compare the videos may not achieve promising results.

Therefore, we propose a novel pose-robust framework for face verification in unconstrained videos that make full use the pose information embedded in each video frame to improve the video representation and the learning step. Our contributions include: (a) a new full-pose video representation, and (b) an ensemble cross-pose classifier, both using the information of different poses to suppress the intra-person variance and increase variance between different persons. The following part will provide more details about our proposed solution.

4.2. Proposed framework

In this section, we present a new method for face verification in unconstrained videos. To explicitly handle the large pose variance in different videos, we first recover full-pose representation for each video which covers all the poses from the most left to the most right profile faces. Specifically, the keyframes are first divided into k predefined pose categories according to the detected head poses. For pose categories with no keyframes, we synthesize face frames using either a flip operation or a generative model with keyframes of known poses. After that, the keyframes in each pose category are averaged to obtain one representative frame, which both simplifies the problem and reduces the possible variances in expressions and lighting conditions.

The full-pose representation helps to capture the pose structure of each video. However, we are more interested in the specific face verification task in which face matching is performed between each pair of videos. In order to better utilize the information of different poses during the matching stage, we further propose an ensemble cross-pose classifier. In particular, we use $k \times k$ classifiers targeting at $k \times k$ different pose combinations between any two videos. For each pair of input videos, we populate the $k \times k$ pairs of keyframes between them and use the corresponding classifier to obtain the output similarity. Then, the $k \times k$ outputs from these classifiers are concatenated into a single score vector. Finally, an SVM classifier is trained on these k^2 similarity scores to perform the face verification task.

Thereafter, for the test video pair, we follow similar stages to extract the video descriptors and enumerate $k \times k$ possible pose combinations. The pairs of frames are routed to the corresponding cross-pose classifiers to obtain the classification scores. Finally, the previously trained SVM classifier will fuse all the frame-level

Chapter 4: Face verification in unconstrained videos

scores and make the final decision on whether two videos are of the same person.

Figure 10 shows the pipeline of our proposed method for a pair of videos.

Figure 10. An overview of our face verification scheme. Given the video pair, we first extract the full-pose representation for each video, using either flip operation (frame with yellow cross) or transfer operation (white cross) to fill in all missing pose categories. Using multiple cross-pose classifiers, we obtain the similarity scores between every frame-level pairs between the two representations and concatenate them together to form the similarity vector. Finally, an SVM classifier is used to learn the threshold and decided whether the two videos are of the same subject.

4.2.1. Full-pose video representation

As pointed out in the previous part, the problem when matching two videos arises because videos contain person in different poses and each pose may have different number of keyframes; thus prevent traditional recognition methods from achieving good result. To reduce such pose variance in different videos, we therefore propose to represent each video with the full-pose face representation, which covers all the poses from the most left to the most right profile. By leveraging the pose information in each keyframe, this full-representation helps to normalize all the

Chapter 4: Face verification in unconstrained videos

pose difference so that each video contains exactly the same number of keyframes and covers all possible viewpoints.

Formally, let us denote a video as $V = \{f_i\}_{i=1}^n$ where f_i a keyframe and n is the total number of keyframes. We first define k pose categories $\{P_j\}_{j=1}^k$ which covers the face pose from most left profile to the most right profile. Then, we estimate the head pose of each frame f_i in the original video and allocate this frame to the corresponding pose category according to the estimated angle. However, a real-world video usually does not cover all of the k poses but rather only several of them; for example, in the news or interview videos, the person (narrator) is often captured in frontal view with only slight head movements. As a result, some categories containing no keyframes, which means the pose variance between different videos still exists. To fill these categories, we therefore propose two operations to synthesize face frames for them.

Flip Operation

Since the face is approximately symmetric, our first solution is to horizontally flip the keyframes in one category on one side (*e.g.*, the most left profile) to fill its corresponding category on the other side (*e.g.*, the most right profile) if that is empty. In other words, we flip and allocate the keyframes in category P_j to the category P_{k-j+1} if P_{k-j+1} does not have any keyframes. With this simple operation, we can reduce the number of missing poses by more than 30%.

Transfer Operation

To fill the remaining empty categories, we propose to transfer the keyframes in the nonempty categories to synthesize virtual poses for the empty categories. Particularly, we employ the deformable model in [47]. Although the initial intention is for face alignment, their model can be applied to transfer the face appearance

Chapter 4: Face verification in unconstrained videos

from one viewpoint to another one. Similar to [47], we firstly find a set of 65 fiducial landmarks using the feature points detector in [48] and obtain the response maps for individual landmark. After that, by using the subspace constrained mean-shifts [47], we can iteratively fit the point distribution model to jointly optimize the landmark locations. Through the fitting process, we know the optimal location of the fiducial points on the original image as well as on any viewpoints. Therefore, we can synthesize the face in novel view by defining a triangulation on the set of fiducial points and performing a piecewise affine transformation that takes every triangle to the desired position in the target view.

Figure 11 illustrates the method to transfer the face appearance of one pose to another pose. Using the above warping method, we obtain a set of synthesized keyframes to fill each empty pose category.

Figure 11. (a) The original keyframe with the triangulation and
(b) its transformations to 7 different pose categories.

Chapter 4: Face verification in unconstrained videos

By using the above two operations, we have represented the video with a set of pose categories, each of which contains some keyframes with similar poses. To further reduce expression and lighting variations within each category, we propose to average all the keyframes in one pose category into a single face frame. Thus, we finally represent each video as a set of keyframes, each of which represents a distinct pose and all of which cover the poses from the most left to the most right profile, *i.e.*, $V = \left\{ r_j \right\}_{j=1}^k$ frames where r_j is the averaged frame for the j -th pose category. The steps in the face representation are summarized in the Figure 12.

Pose Representation (video V ; pose categories $P = \{P_1 \dots P_k\}$)

```

for keyframe  $f_i$  in  $V$ 
    estimate the rotation angle of  $f_i$ 
    assign  $f_i$  to the corresponding pose  $P_j$ 
end_for

for pose category  $P_j$ 
    if  $P_j$  contains no keyframe
        if  $P_{k-j+1}$  is not empty
             $P_j = \text{flip}(P_{k-j+1})$ 
        else
            find the nearest non-empty pose  $P_l$ 
             $P_j = \text{transfer}(P_l, \text{pose } j)$ 
        end_if
    end_if
     $r_j = \text{average}(\text{keyframes in } P_j)$ 
end_for

return  $k$  average frames  $\{r_1 \dots r_k\}$ 

```

Figure 12. Pseudo-code for the full-pose video representation

4.2.2. Ensemble cross-pose classifiers

In the face verification setting, one is given a set of video pairs divided into two categories, same-person and different-person pairs. Specifically, a same-person video pair $(V_1, V_2) \in S$ means that the two videos are from the same subject, while $(V_1, V_2) \in \mathcal{D}$ means that the videos are from two different subjects. The task of face verification is to predict whether two unseen test videos are from the same person or not.

Chapter 4: Face verification in unconstrained videos

After extracting the full-pose representation in previous section, each video consists of exactly k average frames in which each frame characterizes one pose category. The simplest classification algorithm is probably done by matching the corresponded keyframes in the video pair and taking the majority vote or weighted average of similarity scores. However, due to the errors of pose estimation and pose synthesis, the representative keyframes are imperfect: matching between real keyframes and the synthesized ones is not as reliable as matching real keyframes of two videos, although they may be from different categories. Therefore, each pair of keyframes between the two videos contains some information beneficial to the recognition process, and in the matching step all $k \times k$ pairs of keyframes should be considered instead of only the corresponded ones.

Recall that each keyframe represents a distinct pose, so the $k \times k$ frame-level pairs between any input videos actually represent $k \times k$ cross-pose pairs of the faces. As one universal classifier may not well model the similarity measurement for all cross-pose pairs, we propose to learn $k \times k$ cross-pose classifiers, each corresponds to one cross-pose combination.

Formally, let us denote the classifier trained on a cross-pose pair (P_i, P_j) as C_{ij} , where P_i is the i -th pose in the first video and P_j is the j -th pose in the second video with $i, j = 1, \dots, k$. To train each individual classifier C_{ij} , we employ the Side-Information Linear Discriminant analysis (SILD) [22] approach, which extends the traditional Linear Discriminant Analysis by defining the between-class and within-class scatter matrices using only the side information.

In the training process, we divide frame-level pairs from the n training video pairs into $k \times k$ non-overlapped subsets $\{G_{ij}\}_{i,j=1,\dots,k}$. Specifically, for the video pair $V_1 = \{r_{1i} \mid i=1 \dots k\}$ and $V_2 = \{r_{2j} \mid j=1 \dots k\}$, we populate all cross-pose frame-level pairs

Chapter 4: Face verification in unconstrained videos

between the two videos as $\{(r_{1i}, r_{2j}) \mid i, j=1\}^k$. The pair of keyframes (r_{1i}, r_{2j}) is allocated to the set $\mathcal{G}_{ij}$ for training the classifier C_{ij} . Therefore, each subset $\mathcal{G}_{ij}$ contains n frame-level pairs, in which the first frame in each pair belongs to pose P_i while the second frame belongs to pose P_j . Following [22], the within-class scatter matrix S_W^{ij} and between-class scatter matrix S_B^{ij} for the pose combination (i, j) are respectively defined as:

$$S_W^{ij} = \sum_{(r_{1i}, r_{2j}) \in \mathcal{S}} (r_{1i} - r_{2j})(r_{1i} - r_{2j})^T,$$

$$S_B^{ij} = \sum_{(r_{1i}, r_{2j}) \in \mathcal{D}} (r_{1i} - r_{2j})(r_{1i} - r_{2j})^T.$$

Similar to LDA, the projection matrix for pose combination (i, j) is calculated by solving this optimization problem

$$w_{ij} = \arg \max_w \frac{w^T S_B^{ij} w}{w^T S_W^{ij} w}$$

After projecting the original frame-level pair (r_{1i}, r_{2j}) to a lower dimension using w_{ij} , the classification score s_{ij} can be obtained by calculating their cosine similarity. The score determines the likeness of the two frames regardless of their pose difference. We then concatenate the scores from all $k \times k$ frame-level pairs to form a score vector of length k^2 for each pair of videos. Finally, a binary SVM classifier with RBF kernel is trained on the cross-pose score vector obtained from the training video pairs.

Chapter 4: Face verification in unconstrained videos

For any given pair of test videos V_1 and V_2 , we also enumerate all possible cross-pose frame pairs between the two. The frame pair (r_{1i}, r_{2j}) is then dispatched to the classifier C_{ij} to produce a prediction score whether the frame r_{1i} coming from the first video and r_{2j} from the second video is of the same person. Thus, there are $k \times k$ such scores as a result of feeding $k \times k$ frame pairs to their corresponding classifiers. Finally, the pre-trained SVM classifier is used to fuse all individual classifier scores and predict whether the videos are from the same person or not.

4.3. Experimental Result

4.3.1. Dataset

Experiments are performed on the large-scale YouTube Faces (YTF) dataset [23] to assert the performance of our framework in unconstrained environment. The dataset follows the design of Label Faces in the Wild [18] protocol with videos downloaded from YouTube website by searching the celebrities' names. Many of these videos are captured by amateurs, typically under poor lighting conditions, difficult poses, and are often corrupted by motion blur, hence making the verification task on this dataset extremely challenging.

Figure 13. Samples of the videos in YouTube Faces dataset.

The recognition task is challenging due to extreme lighting, pose, occlusion, motion blur and low resolution.

Chapter 4: Face verification in unconstrained videos

The YTF dataset contains 3425 videos of 1595 different people, on average there are 181 frames per video. The author provided the standard benchmark for performance evaluation: altogether there are 5000 video pairs equally divided into 10 independent splits, with 250 same pairs and 250 different pairs in one split. We use the provided aligned version and crop each frame to the size of 150×80 pixels. We strictly follow the restricted protocol in which only the same/not-same label is known at the training time and no additional dataset is used to assist the learning stage.

The estimated head rotation angle released with the dataset is used to divide the frames into 7 pose categories in all experiments. The pose categories are defined in Table 7.

Pose category	1	2	3	4	5	6	7
Head Rotation Angle	−90°..−45°	−45°..−25°	−25°..−10°	−10°..10°	10°..25°	25°..45°	45°..90°
Frequency	12.88%	35.91%	57.96%	76.64%	61.28%	38.92%	16.70%

Table 7. Pose categories statistics from YTF dataset.

The frontal pose appears in most of the video (76.64%) since the celebrities frequently look at the camera in the video clips. As moving from the frontal to most left or right profile category, the frequency of pose category decreases significantly although the range is extended. The statistics also show that only a small fraction of videos (*i.e.* around 15%) in unconstrained environment captures the face from a diverse viewpoint of 45 degree and above. This is somehow similar to the assumption we have while analyzing the unconstrained images in the previous section.

4.3.2. Feature extraction

Our framework is tested on several well-known features for face application including Intensity, Local Binary Patterns and Gabor wavelet feature. Due to the space constraint, we apply different down-sampling factors and keep between 1000 to 5000 dimensions for each feature vector. In details:

- For Intensity feature, we keep every other pixel and concatenate all the value to a single vector of $75 \times 40 = 3000$ -D
- For LBP features, the face region is divided into non-overlap blocks of the size 30×20 and a histogram of 59 bins is used to represent each block. All the histogram values are concatenated to form a 1180-D vector.
- For Gabor features, we use 40 filters using 5 scales and 8 orientations. After convolution each of them with the original face image, we further down-sample the high dimensional resultant images using a scaling factor of 10×10 . The dimension of Gabor feature is thus equal to 4,800.

As a preprocessing step, PCA is used to further reduce the high dimensionality and suppress noise. The dimension after PCA is determined by preserving 98% energy or about 300 to 500 dimensions. For the side-information LDA projection step, we keep 30% of the columns as suggested in [22].

4.3.3. Result of individual classifier

First we examine the result of individual cross-pose classifier for our representation. Although there are 7×7 cross-pose combinations, the classifier C_{ij} to distinguish pose P_i from pose P_j should be similar to C_{ji} due to the formulation of the classifier. Taking advantage of the symmetry of the pose categories, we train only half of the classifiers; thus both increase the size of the training set for each individual classifier and make the training time almost twice as fast.

Figure 14. Confusion matrix of cross-pose classifiers

The performance of individual cross-pose classifiers is visualized in Figure 14. While frontal-to-frontal matching accomplishes up to 77% accuracy, matching between the most left and most right profile view succeeds in less than 70% of the cases. The confusion matrix shows good result around the diagonal elements as matching similar viewpoints produces higher recognition rate than matching different viewpoints. Besides, the center of the matrix is considerably brighter, proving that frontal pose is the easiest pose and matching near-frontal images are much simpler than matching profile views.

4.3.4. Impact of each component

There are two main contributions in our framework: (a) pose synthesis for video representation, and (b) the adaptive use of multiple classifiers. In order to demonstrate the contribution of each component towards our final result, we adopt four different variants of the algorithm:

- **single + no synthesis:** This is our baseline without using pose-based representation or ensemble classifiers. Instead, we use the “exemplar-based” approach to represent the videos. First, k -means is run on each video to group the frames into 7 clusters and the cluster centers are used as the video representation. Thereafter, each input video pair is transformed to 7×7 cluster-level pairs. A common classifier is trained on all pairs and the video similarity is calculated by averaging classification scores of 49 cluster pairs.
- **single + synthesis:** We employ the full pose representation, in which the keyframes are divided into 7 pose categories and pose synthesis is used to fill all missing categories. However, only one common classifier is used to make the prediction rather than using multiple cross-pose classifiers. Similar to the previous variants, the similarity between two video is calculated by averaging the classification scores of 49 frame-level pairs.
- **multiple + no synthesis:** This variant of the algorithm help to understand the contribution of multiple cross-pose classifiers towards the final result. We divide the keyframes into poses and the synthesis stage is excluded. Each classifier are then trained using only available poses. The missing of several classifier scores makes it impossible for SVM to produce an adequate result; hence, weighted average is used to fuse individual classifier

Chapter 4: Face verification in unconstrained videos

results instead. The weight of each classifier is proportional to its training error.

- **multiple + synthesis:** This is our unified framework consists of both full-pose representation and ensemble of cross-pose classifiers.

The results of the above four experiments are summarized in Table 8.

	Intensity	LBP	Gabor
single + no synthesis	76.70 $\pm$ 1.21	76.20 $\pm$ 2.06	77.48 $\pm$ 2.10
single + synthesis	77.60 $\pm$ 1.43	77.98 $\pm$ 2.16	79.26 $\pm$ 1.55
multiple + no synthesis	78.66 $\pm$ 1.24	78.76 $\pm$ 1.66	80.08 $\pm$ 1.26
multiple + synthesis	79.12 $\pm$ 1.81	79.58 $\pm$ 2.01	80.92 $\pm$ 1.10

Table 8. Results on YouTube Faces dataset

The pose representation and synthesis help to improve the results by around 1.5% from the baseline of a single classifier without pose synthesis; this is also illustrated in the Figure 15(a). The performance increment proves that our pose-based representation better captures the intra-person variance and makes comparison between videos fairer.

On the other hand, using ensemble of cross-pose classifiers help to improve the result significantly by around 2.5% from the baseline. The huge improvement shows that one classifier is not robust enough to all the intra-person variance, and considering the pose information when making the decision helps to increase the accuracy. Combining both components together, our unified framework achieves a considerable boost of 3% compare to the baseline. The ROC curve for the ensemble classifiers is shown in Figure 15(b).

Chapter 4: Face verification in unconstrained videos

For comparison, both figures also include two previously best results from [24]. In the first method (mindist), the distance matrix between every frame pairs of the two videos is calculated and the minimum distance is used to determine whether both videos show the same person. The second method, Matched Background Similarity (MBGS), is the current best result on the dataset which achieved the recognition rate of 76.4 ± 1.8 . The result is obtained by training a discriminative SVM classifier for each video sequence with the positive class includes all the keyframes of the current video and the negative class consists of keyframes belong to a different reference subject.

Chapter 4: Face verification in unconstrained videos

(a)

(b)

Figure 15. ROC curve for the YTF dataset on LBP feature.

(a) the contribution of full-pose representation and

(b) the contribution of multiple cross-pose classifiers

4.3.5. Performance of the unified framework

In the unified framework, we use another layer of SVM to combine the result from the above 3 different features, including Intensity, LBP and Gabor. This will help to incorporate all the strengths of individual feature into our final classifier. Again, we compare our result with the mindist and MBGS methods from [24]. We also include the ROC curves from two recent papers, APEM-FUSION [49] and STFRD+PMML [50], which are accepted for publication after this thesis is submitted.

Figure 16 shows the ROC curve of all methods. The minimum distance method does not perform well on this dataset, as they are quite simple and the distance between one pair of frames certainly does not provide the full picture of the video similarity. Learning-based approaches such as the MBGS or our baseline greatly improve the result, since they exploit the information from all the keyframes and try to learn a discriminative classifier to distinguish between positive and negative samples.

Figure 16. Performance comparison on the YTF dataset.

Chapter 4: Face verification in unconstrained videos

When combining all three feature types with an additional layer of SVM, we achieve the best recognition rate of 82.24 ± 1.14 . Our framework significantly outperforms the published state-of-the-art of 76.4 ± 1.8 reported in [24] by 5.8% as well as all latest results from this year [49] [50] by almost 3%.

4.4. Summary

We have introduced a novel framework for video based face verification by leveraging the viewpoint information. Our two main contributions include a simple video descriptor that helps to better represent the pose structure and the ensemble classifiers to distinguish the pose variation from the identity difference. Our framework consistently achieves higher accuracy on unconstrained videos collected from YouTube website.

The work in this chapter has been accepted to IEEE International Conference on Image Processing (ICIP), 2013.

Chapter 5

Conclusion and Future Work

5.1. Summary of current work

In this report, we have summarized different methods and features for face recognition and investigated in details their performance on various large-scale, uncontrolled datasets. It has been shown that the Fisherfaces significantly performs better than Eigenfaces and mostly ties with Locality-constrained Linear Coding, while its speed is considerably faster than LLC. On the feature side, Gabor always achieves the best performance though it comes with the price of huge computation and storage tradeoff, while LBP generally performs well despite its simplicity and compactness. Our extensive experiments also show that image-based face recognition can achieve relatively good result on unconstrained environment using only simple methods and features; hence they are highly suitable for real face recognition application.

Motivated from the image recognition result, we have presented a novel framework for video-based face verification problem. By leveraging the pose information, a new full-pose representation and an ensemble of cross-pose classifiers have been proposed for video-based recognition. When testing on a large-scale unconstrained video dataset downloaded from YouTube website, our framework achieved 82.24% in terms of verification rate, significantly outperformed the state-of-the-art of 76.4%.

Chapter 5: Conclusion and Future Work

5.2.Future work

Currently, each classifier is individually trained using subset of the training data from a specific pose category, thus making extreme poses more difficult to recognize due to limited available keyframes for training. Perhaps a joint optimization of the ensemble classifiers is more desirable. Alternatively, a reference set of videos which cover all pose categories may help in the learning stage.

Another interesting future work is to apply the framework on image recognition task assuming there are enough training samples for each person.

References

- [1] S.Z. Li and A.K. Jain. Handbook of face recognition. *Springer*, 2004.
- [2] 2003. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399-458, 2003.
- [3] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3:71-86, 1991.
- [4] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711-720, 1997.
- [5] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian face recognition. *Pattern Recognition*, 33(11):1771-1782, 2000.
- [6] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 11(4):467- 476, 2002.
- [7] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma. Robust Face Recognition via Sparse Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210-227, 2009.
- [8] F.S. Samaria and A.C. Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of IEEE Workshop on Applications of Computer Vision*, 1994.
- [9] A.M. Martinez and R. Benavente. The AR Face Database. In *CVC Technical Report #24*, 1998.
- [10] T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1615-1618, 2003.
- [11] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090-1104, 2000.
- [12] T. Ojala, M. Pietikainen, and T. Maenpaa. Multi resolution gray-scale and rotation invariant texture classification withlocal binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 2002.

- [13] D. Lowe. Distinctive image features from scale-invariant keypoints. *In International Journal of Computer Vision*, 2004.
- [14] Z. Cao, Q. Yin, X. Tang, and J. Sun. Face recognition with Learning-based Descriptor. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [15] N. Kumar, A.C. Berg, P.N. Belhumeur, and S.K. Nayar. Attribute and Simile Classifiers for Face Verification. *In IEEE International Conference on Computer Vision*, 2009.
- [16] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang and Y. Gong. Locality-constrained linear coding for image classification. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [17] R. Gross, I. Matthews, J. Cohn, T. Kanade and S. Baker. Multi-PIE. *In IEEE Conference on Automatic Face and Gesture Recognition*, 2008.
- [18] G.B. Huang, Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. *In Technical Report*, 2007.
- [19] G. R. Bradski and V. Pisarevsky. Intel's computer vision library: Applications in calibration, stereo segmentation, tracking, gesture, face and object recognition. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2000.
- [20] M. Everingham, J. Sivic and A. Zisserman. "Hello! My name is... Buffy" - Automatic naming of characters in TV video. *In British Machine Vision Conference*, 2006.
- [21] Q. Yin, X. Tang and J. Sun. An Associate-Predict Model for Face Recognition. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [22] M. Kan, S. Shan, D. Xu and X. Chen. Side-Information based Linear Discriminant Analysis for Face Recognition. *In The British Machine Vision Conference*, 2011.
- [23] L. Wolf, T. Hassner and I. Maoz. Face Recognition in Unconstrained Videos with Matched Background Similarity. *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [24] L. Wolf, T. Hassner and I. Maoz. Face Recognition in Unconstrained Videos with Matched Background Similarity. *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [25] Y. Adini, Y. Moses, and S. Ullman.. Face Recognition: The Problem of Compensating for Changes in Illumination Direction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

- [26] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2008.
- [27] N. Pinto, J.J. DiCarlo, and D.D. Cox. How far can you get with a modern face recognition test set using only simple features?. *Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [28] I. Jolliffe. Principal component analysis. *John Wiley & Sons, Ltd*, 2005.
- [29] A.S. Georgiades, P.N. Belhumeur, and D.J. Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under . *IEEE Trans. Pattern Anal. Mach. Intelligence*, 2001.
- [30] G. Shakhnarovich, J. Fisher, and T. Darrell. Face recognition from long-term observations. *European Conference on Computer Vision*, 2002.
- [31] O. Yamaguchi, K. Fukui, and K.I. Maeda. Face recognition using temporal image sequence.. *Automatic Face and Gesture Recognition*, 1998.
- [32] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. *Computer Vision and Pattern Recognition (CVPR)*, 2008.
- [33] Z. Cui, S. Shan, H. Zhang, S. Lao, and X. Chen. Image Sets Alignment for Video-based Face Recognition. *Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [34] U. Park, A.K. Jain, and A. Ross. Face recognition in video: Adaptive fusion of multiple matchers. *Computer Vision and Pattern Recognition*, 2007.
- [35] K. Toyama, and A. Blake. Probabilistic tracking in a metric space. *International Conference on Computer Vision (ICCV)*, 2001.
- [36] V. Krueger, and S. Zhou. Exemplar-based face recognition from video. *European Conference on Computer Vision*, 2002.
- [37] D.A. Roark, A.J. O'Toole, H. Abdi, and S.E. Barrett. Learning the moves: The effect of familiarity and facial motion on person recognition across large changes in viewing format. *Perceptron*, 2006.
- [38] X. Liu, and T. Chen. Video-based face recognition using adaptive hidden Markov model. *Computer Vision and Pattern Recognition*, 2003.
- [39] S. Zhou, V. Krueger, and R. Chellappa. Probabilistic recognition of human faces from video. *Computer Vision and Image Understanding*, 2003.

- [40] M.A. Hearst, S.T. Dumais, E. Osman, J. Platt, and B. Scholkopf. Support vector machines. *Intelligent Systems and their Applications*, 1998.
- [41] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.R. Mullers. Fisher discriminant analysis with kernels. *Neural Networks for Signal Processing*, 1999.
- [42] T. Berg, and P.N. Belhumeur. Tom-vs-Pete Classifiers and Identity-Preserving Alignment for Face Verification.. *British Machine Vision Conference*, 2012.
- [43] H. Nguyen, and L. Bai. Cosine similarity metric learning for face verification. *Asian Conference on Computer Vision*, 2010.
- [44] P. Li, Y. Fu, U. Mohammed, J.H. Elder, and S.J.D. Prince. Probabilistic Models for Inference About Identity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.
- [45] X. Zhang, and Y. Gao. Face recognition across pose: A review. *Pattern Recognition*, 2009.
- [46] X. Liu, and T. Chen. Pose-robust face recognition using geometry assisted probabilistic modeling. *Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [47] J.M. Saragih, S. Lucey and J.F. Cohn. Face alignment through subspace constrained mean-shifts. *International Conference on Computer Vision*, 2009.
- [48] Y.Wang, S. Lucey, and J. Cohn. Enforcing Convexity for Improved Alignment with Constrained Local Models. *Computer Vision and Pattern Recognition (CVPR)*, 2008.
- [49] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang,. Probabilistic elastic matching for pose variant face verification. *Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [50] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen,. Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. *Computer Vision and Pattern Recognition (CVPR)*, 2013.

Appendix

Appendix A. The result of each method and descriptor on FERET dataset with respect to different parameters

Descriptor & Method	Parameter	Rank-1 Result	Rank-5 Result	Time/Sample
Intensity + PCA	100	5.25 ± 1.03	15.31 ± 0.81	0.0112
	300	5.84 ± 1.20	16.51 ± 0.99	0.0431
	500	5.84 ± 1.20	16.68 ± 0.94	0.0980
	700	5.87 ± 1.24	16.73 ± 0.97	0.1836
	1000	5.87 ± 1.24	16.73 ± 0.98	0.8637
Intensity + LDA	100	83.58 ± 0.90	96.45 ± 1.21	0.0104
	150	88.02 ± 1.09	97.40 ± 0.67	0.0182
	200	89.64 ± 0.86	97.79 ± 0.71	0.0276
	250	90.17 ± 0.53	97.77 ± 0.72	0.0374
	300	90.22 ± 0.26	97.82 ± 0.90	0.0467
Intensity + LLC	0.1	91.96 ± 0.85	96.65 ± 0.57	1.3496
	0.3	90.67 ± 0.81	96.23 ± 0.76	1.3491
	0.5	89.64 ± 1.32	95.95 ± 0.62	1.3492
	0.7	88.63 ± 0.96	95.45 ± 0.58	1.3594
	0.9	87.68 ± 0.84	95.20 ± 0.48	1.3499
LBP + PCA	100	67.32 ± 2.39	83.94 ± 0.88	0.0090
	300	72.68 ± 1.90	86.76 ± 0.65	0.0394
	500	73.74 ± 2.01	87.32 ± 0.57	0.0926
	700	73.99 ± 1.97	87.35 ± 0.56	0.1767
	1000	73.94 ± 1.98	87.37 ± 0.61	0.8468
LBP + LDA	100	91.54 ± 0.86	98.10 ± 0.61	0.0063
	150	93.74 ± 0.98	98.32 ± 0.76	0.0200
	200	94.25 ± 0.91	98.58 ± 0.61	0.0336
	250	94.33 ± 1.08	98.58 ± 0.52	0.0436
	300	94.33 ± 0.85	98.55 ± 0.52	0.0532
LBP + LLC	0.1	93.24 ± 1.16	96.65 ± 1.10	0.3169
	0.3	94.30 ± 1.65	97.77 ± 0.72	0.3166
	0.5	94.33 ± 1.43	97.77 ± 0.54	0.4058
	0.7	94.50 ± 1.29	97.88 ± 0.51	0.6503
	0.9	94.55 ± 1.14	97.79 ± 0.43	0.7244

Descriptor & Method	Parameter	Rank-1 Result	Rank-5 Result	Time/Sample
Gabor + PCA	100	46.68 $\pm$ 2.24	70.39 $\pm$ 2.18	0.0116
	300	59.55 $\pm$ 2.01	78.24 $\pm$ 1.64	0.0444
	500	62.49 $\pm$ 1.75	80.25 $\pm$ 1.84	0.1008
	700	63.49 $\pm$ 1.90	80.81 $\pm$ 1.57	0.1879
	1000	64.02 $\pm$ 1.88	81.06 $\pm$ 1.67	0.8693
Gabor + LDA	100	86.65 $\pm$ 1.54	95.95 $\pm$ 1.51	0.0118
	150	92.88 $\pm$ 1.30	97.65 $\pm$ 0.75	0.0289
	200	94.78 $\pm$ 1.20	98.46 $\pm$ 0.40	0.0405
	250	96.34 $\pm$ 0.60	99.08 $\pm$ 0.32	0.0537
	300	97.35 $\pm$ 0.36	99.33 $\pm$ 0.18	0.0746
Gabor + LLC	0.1	93.24 $\pm$ 1.16	96.65 $\pm$ 1.10	1.7143
	0.3	94.30 $\pm$ 1.65	97.77 $\pm$ 0.72	1.7216
	0.5	94.33 $\pm$ 1.43	97.77 $\pm$ 0.54	1.7205
	0.7	94.50 $\pm$ 1.29	97.88 $\pm$ 0.51	1.7218
	0.9	94.55 $\pm$ 1.14	97.79 $\pm$ 0.43	1.7216