Non-linear dimensionality reduction and sparse representation models for facial analysis

Yuyao Zhang

Numéro d'ordre: 2014ISAL0019 Année 2014

Institut National Des Sciences Appliquées de Lyon

Laboratoire d'InfoRmatique en Image et Systèmes d'information

Ecole Doctorale Informatique et Mathématique de Lyon

Thèse de l'Université de Lyon

Présentée en vue d'obtenir le grade de Docteur,
spécialité Informatique

par

Yuyao ZHANG

Non-linear Dimensionality Reduction and Sparse Representation Models for Facial Analysis

Thèse soutenue le 20 Févier 2014 devant le jury composé de:

M.	Thierry Chateau	Professeur, Université de Clermont Ferrand 2	Rapporteur
M.	Denis Pellerin	Professeur, Université Joseph Fourier	Rapporteur
M.	Atilla Baskurt	Professeur, INSA Lyon	Examinateur
M.	Hubert Konik	Maître de conférences, Laboratoire Hubert Curien	Examinateur
M.	William Puech	Professeur, Université Montpellier 2	Président
M.	Christophe Garcia	Professeur, INSA Lyon	Directeur
M.	Khalid Idrissi	Maître de conférences, INSA Lyon	Directeur

Laboratoire d'InfoRmatique en Image Systèmes d'information
UMR 5205 CNRS - INSA de Lyon - Bât. Jules Verne
69621 Villeurbanne cedex - France
Tel: +33(0)4 72 43 60 97 - Fax: +33(0)4 72 43 71 17

Abstract

Face analysis techniques commonly require a proper representation of images by means of dimensionality reduction leading to embedded manifolds, which aims at capturing relevant characteristics of the signals.

In this thesis, we first provide a comprehensive survey on the state of the art of embedded manifold models.

Then, we introduce a novel non-linear embedding method, the Kernel Similarity Principal Component Analysis (KS-PCA), into Active Appearance Models, in order to model face appearances under variable illumination. The proposed algorithm successfully outperforms the traditional linear PCA transform to capture the salient features generated by different illuminations, and reconstruct the illuminated faces with high accuracy.

We also consider the problem of automatically classifying human face poses from face views with varying illumination, as well as occlusion and noise. Based on the sparse representation methods, we propose two dictionary-learning frameworks for this pose classification problem.

The first framework is the Adaptive Sparse Representation pose Classification (ASRC). It trains the dictionary via a linear model called Incremental Principal Component Analysis (Incremental PCA), tending to decrease the intra-class redundancy which may affect the classification performance, while keeping the extra-class redundancy which is critical for sparse representation.

The other proposed work is the Dictionary-Learning Sparse Representation model (DLSR) that learns the dictionary with the aim of coinciding with the classification criterion. This training goal is achieved by the K-SVD algorithm.

In a series of experiments, we show the performance of the two dictionary-learning methods which are respectively based on a linear transform and a sparse representation model.

Besides, we propose a novel Dictionary Learning framework for Illumination Normalization (DL-IN). DL-IN based on sparse representation in terms of coupled dictionaries. The dictionary pairs are jointly optimized from normally illuminated and irregularly illuminated face image pairs. We further utilize a Gaussian Mixture Model (GMM) to enhance the framework's capability of modeling data under complex distribution. The GMM adapt each model to a part of the samples and then fuse them together. Experimental results demonstrate the effectiveness of the sparsity as a prior for patch-based illumination normalization for face images.

Keywords: Face Analysis, Active Appearance Models (AAMs), Kernel Similarity PCA (KS-PCA), Sparse Representation, Dictionary-learning, Face Pose Classification, Illumination Normalization

Résumé

Les techniques d'analyse du visage nécessitent généralement une représentation pertinente des images, notamment en passant par des techniques de réduction de la dimension, intégrées dans des schémas plus globaux, et qui visent à capturer les caractéristiques discriminantes des signaux.

Dans cette thèse, nous fournissons d'abord une vue générale sur l'état de l'art de ces modèles, puis nous appliquons une nouvelle méthode intégrant une approche non-linéaire, Kernel Similarity Principle Component Analysis (KS-PCA), aux Modèles Actifs d'Apparence (AAMs), pour modéliser l'apparence d'un visage dans des conditions d'illumination variables. L'algorithme proposé améliore notablement les résultats obtenus par l'utilisation d'une transformation PCA linéaire traditionnelle, que ce soit pour la capture des caractéristiques saillantes, produites par les variations d'illumination, ou pour la reconstruction des visages.

Nous considérons aussi le problème de la classification automatiquement des poses des visages pour différentes vues et différentes illumination, avec occlusion et bruit. Basé sur les méthodes des représentations parcimonieuses, nous proposons deux cadres d'apprentissage de dictionnaire pour ce problème.

Une première méthode vise la classification de poses à l'aide d'une représentation parcimonieuse active (Active Sparse Representation ASRC). En fait, un dictionnaire est construit grâce à un modèle linéaire, l'Incremental Principle Component Analysis (Incremental PCA), qui a tendance à diminuer la redondance intra-classe qui peut affecter la performance de la classification, tout en gardant la redondance inter-classes, qui elle, est critique pour les représentations parcimonieuses.

La seconde approche proposée est un modèle des représentations parcimonieuses basé sur le Dictionary-Learning Sparse Representation (DLSR), qui cherche à intégrer la prise en compte du critère de la classification dans le processus d'apprentissage du dictionnaire. Nous faisons appel dans cette partie à l'algorithme K-SVD.

Nos résultats expérimentaux montrent la performance de ces deux méthodes d'apprentissage de dictionnaire.

Enfin, nous proposons un nouveau schéma pour l'apprentissage de dictionnaire adapté à la normalisation de l'illumination (Dictionary Learning for Illumination Normalization: DLIN). L'approche ici consiste à construire une paire de dictionnaires avec une représentation parcimonieuse. Ces dictionnaires sont construits respectivement à partir de visages illuminées normalement et irrégulièrement, puis optimisés de manière conjointe. Nous utilisons un modèle de mixture de Gaussiennes (GMM) pour augmenter la capacité à modéliser des données avec des distributions plus complexes. Les résultats expérimentaux démontrent l'efficacité de notre approche pour la normalisation d'illumination.

Mots-clés: Analyse du visage, Modèles Actifs d'Apparence (AAMs), Kernel Similarity Principal Component Analysis (KS-PCA), Représentations Parcimonieuses, Apprentissage de dictionnaire, classification des poses facials, normalisation d'illumination

Acknowledgements

I would like the readers to keep all these people in mind while reading this dissertation, without them this success would not be possible.

First of all, I would like to express my gratitude to my two thesis advisors for their consistent encouragements and support during three and a half years. I am very thankful to Pr. Christophe Garcia for his insightful suggestions and comments all along this thesis work. He is always ready to answer my questions that are expressed in my poor French; he is so quick and efficient in correcting my papers so that I can do my work at my rhythm. It is my honor to work with him. I am very grateful to Dr. Khalid Idrissi for proposing to me a very interesting Ph.D. subject; and for his guidance, understanding, patience, always believing in my capabilities. He has always encouraged me to not only grow as an instructor but also as a developer and an independent thinker.

I would like to thank Pr. Denis Pellerin, Pr. Thierry Chateau, for taking their precious time to review my thesis manuscript. I am also grateful to Dr. Hubert Konik, Pr Atilla Baskurt and Pr. William Puech for having accepted to be examiners of my thesis defence.

I would also like to pay my sincere gratitude to all members of the Imagine team at LIRIS. I can say LIRIS is a great place to do research work because of its friendly work environment, open policies, and overall culture. I thank all my colleagues, administration and for creating such a pleasant work environment and for being there for me. In particular many thanks to Dr. Christian Wolf for the discussion with him which helped to improve the quality of my work. And I thank to Yi, Cagatay, Lilei, Oya, Viencent, Peng, Jinjiang, Mingyuan and Raruca for the daily conversation as colleagues shared same office time.

I thank my friends and my family members (especially my grandparents, my parents and my husband) for their unconditional support all the years.

Finally, I would like to dedicate this thesis to my grandmother Mam. Yiqing Zhang, a great and respectable women, whom I lost during the writing of this manuscript.

Contents

List of Figures

LIST OF FIGURES

LIST OF FIGURES

List of Tables

1

Introduction

1.1 Motivation

The fast development of information and internet technology makes people possible to get huge amounts of multimedia resources, including large number of images, videos, texts and audios, etc. With the explosion of information, a common requirement is to find more meaningful and significant representation which captures low-dimensional structures hidden in raw high-dimensional data. A proper representation can reduce data redundancy and discover the essential attribute of the observations, while simplify the computational complexity of the subsequent operations and hence enhance the accuracy and robustness of data analysis. This is a great challenge to researchers in the areas of machine learning, pattern recognition and computer vision. Take the task of face analysis for example. Nowadays, for each person, it is easy to collect multiple facial images, which are captured in different scenes and times, covering different illumination conditions and pose variations. With the big amount of images, one can establish considerable large databases. To effectively get a further insight of the large databases and design stable face analysis algorithms, there exist two basic problems:

- The first one is how to find compact features or proper representation for the face images.
- The second one is how to develop reasonable and effective analysis algorithms according to the data distribution in the database.

Consider the statistical distribution of the data sets, the traditional methods often assume that the global distribution is linear, and make use of linear dimension reduction methods for data analysis. Therefore, if the distribution of data does not follow the linear distribution, the classical linear methods can hardly discover the internal structure of the observations. Therefore, a more general assumption is that the data in the original observation space presents nonlinear distribution, which requires the use of nonlinear dimensionality reduction methods and well developed dictionaries to analyze.

1.2 Objectives and Contributions

The research topic of this thesis work is the face analysis based on non-linear dimension reduction and sparse representation. Our main objective is to construct several effective algorithms which improve the face analysis works under the effect of different head poses and varying lighting conditions.

In order to accomplish the objective, we first introduce a novel non-linear embedding method, the Kernel Similarity Principal Component Analysis (KS-PCA), and combine it with Active Appearance Models, in order to follow the non-linear data distribution which is caused by the face appearances under varying illumination conditions. The proposed algorithm successfully outperforms the traditional linear PCA transform to capture the salient features generated by different illuminations, and reconstruct the illuminated faces with high accuracy.

Although this non-linear orthogonal dictionary is powerful on capturing the variations caused by the change of light, there exists the limitation on representing variety of face pose images. This lead to our study on newer over-complete dictionaries, which having more atoms than the dimension of the signal, and which promises to represent a wider range of signal phenomena.

Automatic and robust algorithms for head pose estimation can be beneficial to many real life applications. Accurately localizing the head and its orientation is either the explicit goal of systems like human-computer interfaces (e.g., reacting to the users head movements), or a necessary preprocessing step for further analysis, such as identification or facial expression recognition. Due to its relevance and to the challenges posed by the problem, there has been considerable effort in the computer vision community to develop fast and reliable algorithms for head pose estimation.

So we continue to consider the problem of automatically classifying human face poses from face views with varying illumination, as well as occlusion and noise. Based on the sparse representation methods, we propose two dictionary-learning frameworks for this pose classification problem.

The first framework is the Adaptive Sparse Representation pose Classification (ASRC). It trains the dictionary via a linear model called Incremental Principal Component Analysis (Incremental PCA), tending to decrease the intra-class redundancy which may affect the classification performance, while keeping the extra-class redundancy which is critical for sparse representation.

The other proposed work is the Dictionary-Learning Sparse Representation model (DLSR) that learns the dictionary with the aim of coinciding with the classification criterion. This training goal is achieved by the K-SVD algorithm. In a series of experiments, we show the performance of the two dictionary-learning methods which are respectively based on a linear transform and a sparse representation model.

Besides the work of automatically annotating face contours under variable illumination conditions, we would like to bring another contribution of normalizing the illumination influence on the frontal faces. The objective of this work is to eliminate the great affect of illumination condition on 2D face image identification problem. We propose a novel Dictionary Learning framework for Illumination Normalization (DL-IN). DL-IN based on sparse representation in terms of coupled dictionaries jointly optimized from normal illuminated and non-standard illuminated face image patch pairs. We further utilize a GMM model to enhance the framework's capability of modeling data under complex distribution by adapting each model to a part of samples and fuse them together. Experimental results demonstrate the effectiveness of the sparsity as a prior for patch-based illumination normalization for face images.

1.3 Outline

The remainder of this manuscript is organized as follows.

Chapter 2 provides a comprehensive survey on the research of embedding models and dictionary learning.

Chapter 3 presents the non-linear embedded model based Active Appearance Models. The strong robustness of the proposed algorithm relies on the appropriate statistical

learning of the illuminated face images.

Chapter 4 describes two efficient pose classification frameworks. The first one is based on a joint optimization of manifold learning and sparse representation. And the second one is based on an analysis dictionary learning strategy.

Chapter 5 presents a multi-model illumination normalization framework. We train the coupled dictionaries heading to search for the 'standard' illuminated image. A pre-clustering step well tracks the spatial distribution of the illuminated face images, which ensures the succeed of the normalization method.

Chapter 6 summarizes the contributions of this manifold learning based face analysis work, draws conclusion of the manuscript and propose several future working direction concerning the research on embedding models.

2

State-of-the-arts

Huge amounts of high-dimensional information are captured every second by diverse natural sensors such as the eyes or ears, as well as artificial sensors like cameras or microphones. This information is largely redundant in two main aspects: it often contains multiple correlated versions of the same physical world and each version is usually densely sampled by generic sensors. The relevant information about the underlying processes that cause the observations is generally of much reduced dimensionality compared to such recorded data sets. In this chapter, we present methods for determining the proper representation of data sets by means of the reduced dimensionality subspaces, which are adaptive to both the characteristics of the signals and the processing task hand. These representations are based on the principle that our observations can be described by a sparse subset of atoms taken from a redundant dictionary, which represents the causes of our observations of the world. We describe methods for learning dictionaries that are appropriate for the representation of given classes of signals and multi-sensor data. We further show that dimensionality reduction based on dictionary representation can be extended to address specific tasks such as data analysis or classification when the learning includes a class separability criteria in the objective function. The benefits of dictionary learning clearly show that a proper understanding of causes underlying the sensed world is the key to task-specific representation of relevant information in high-dimensional data tasks.

2.1 The objective of the Embedding Models

Dimensionality reduction functioning as a feature extraction has two objectives. One objective is to reduce the computational complexity of the subsequent classification with the minimum loss of information needed for classification. The second objective is to circumvent the generalization problem of the subsequent classification and hence enhance its accuracy and robustness. To achieve the first objective, it is straightforward that we should maximize the information carried by the data in the extracted low-dimensional subspace. Although PCA does maximize the data structure information in the principal space and hence is optimal for data reconstruction, it is the discriminative information that plays roles in pattern recognition. Thus, most researchers prefer discriminant analysis to the principal component analysis, as evidenced by the fact that the vast majority of the published approaches are based on some kind of the "most discriminative" criteria. There is no doubt that various discriminant analyses can effectively achieve the first objective. The second objective of the dimensionality reduction is, however, far from straightforward. The most discriminative subspace may not be an effective criterion for it because any dimensionality reduction causes a loss of information, including the discriminative information. Any subspace cannot contain more discriminative information than any larger one that includes the former. Why can the dimensionality reduction boost the classification accuracy if the discriminative information is the most critical for classification? Although some general phenomena, such as the curse of dimensionality, small sample size problem, noise removal effect of dimensionality reduction, and better generalization in a lower dimensional space, are well known in the pattern recognition community, they have not indicated what dimensions should be extracted or what else should be removed for a more robust classification. One cannot develop an effective dimensionality reduction technique to minimize the classification accuracy just based on these general phenomena.

It is thus necessary to study the underlying principles and insights of why and how the dimensionality reduction can enhance the generalization accuracy and robustness of the subsequent classification. This is critical because the second objective of the dimensionality reduction is more important than the first one in most applications with the rapid growth of computation power. The study will also help us find the commonalities and differences of various dimensionality reduction techniques. Without

a thorough analysis and gaining an in-depth understanding of the underlying principles, it is difficult to bring the research in this area to a significantly higher level.

2.2 A History of Embedding Models

2.2.1 The Linear Embedding Models

Signal transforms have been around for as long as signal processing has been conducted. In the 1960's, early signal processing researchers gave significant attention to linear time invariant operators, which were simple and intuitive processes for manipulating analog and digital signals. In this context, the Fourier transform naturally emerged as the basis which diagonalizes these operators, and it immediately became a central tool for analyzing and designing such operators. The transform gained tremendous popularity with the introduction of the Fast Fourier Transform (FFT) in 1965 by Cooley and Tukey [1], which provided its numerical efficiency.

The Fourier basis describes a signal in terms of its global frequency content, as a combination of orthogonal waveforms

$$F = \left\{\phi_n(x) = e^{inx}\right\}_{n\in Z} \tag{2.1}$$

A signal is approximated in this basis by projecting it onto the K lowest frequency atoms, which has a strong smoothing and noise-reducing effect. The Fourier basis is thus efficient at describing uniformly smooth signals. However, the lack of localization makes it difficult to represent discontinuities, which generate large coefficients over all frequencies. Therefore, the Fourier transform typically produces over-smooth results in practical applications. For finite signals, the Fourier transform implicitly assumes a periodic extension of the signal, which introduces a discontinuity at the boundary. The Discrete Cosine Transform (DCT) is the result of assuming an anti-symmetric extension of the signal, which results in continuous boundaries, and hence in a more efficient approximation. Since the DCT has the added advantage of producing noncomplex coefficients, it is typically preferred in applications, see Figure 2.1 for some 2-D DCT atoms.

Signal approximation in the Fourier basis was soon categorized as a specific instance of linear approximation: given a basis $\{\phi_n\}_{n=1}^{N}$ of R^N, a signal $x \in R^N$ is linearly approximated by projecting it onto a fixed subset of $K < N$ basis elements

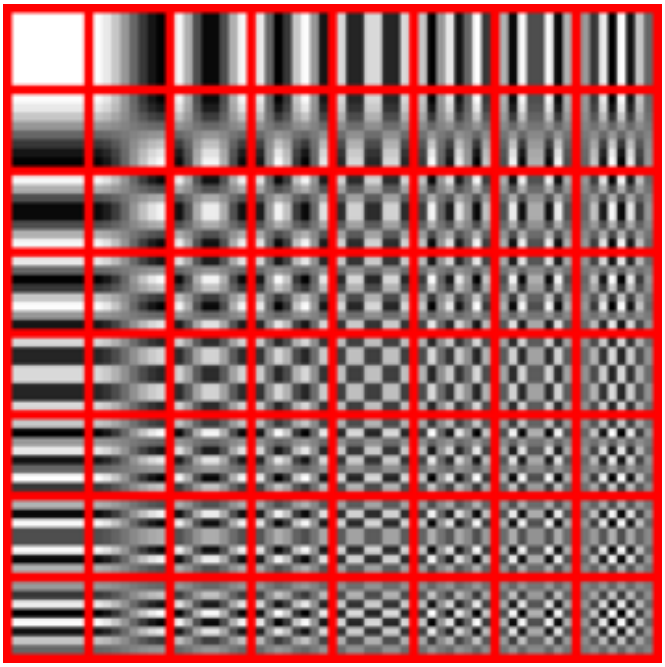

Figure 2.1: **Two-dimensional DCT basis functions** - 64 patterns of 8×8 block

$$x \approx \sum_{n \in I_K(x)} (\psi_n^T x)\phi_n \tag{2.2}$$

where $\{\psi_n\}_{n=1}^N$ is in general the bi-orthogonal basis ($\psi_n = \phi_n$ in the orthogonal case), and $I_K(x)$ is an index set adapted to each signal individually. This process is an under-complete linear transform of x, and, with the right choice of basis, can achieve compaction - the ability to capture a significant part of the signal with only a few coefficients. Indeed, this concept of compaction is later replaced with sparsity, though the two are closely related [2].

Optimizing compaction was a major driving force for the continued development of more efficient representations. During the 1970's and 1980's, a new and very appealing source of compaction was brought to light: the data itself. The focus was on a set of statistical tools developed during the first half of the century, known as the Karhunen-Loeve Transform (KLT) [3, 4], or Principal Component Analysis (PCA) [5]. The KLT is a linear transform which can be adapted to represent signals coming from a certain known distribution. The adaptation process fits a low-dimensional subspace to the data which minimizes the l^2 approximation error. Specifically, given the data covariance

matrix $\sum$ (either known or empirical), the KLT atoms are the first K eigenvectors of the eigenvalue decomposition of $\sum$,

$$\sum = U \Lambda U^T \tag{2.3}$$

From a statistical point of view, this process models the data as coming from a low-dimensional Gaussian distribution, and thus is most effective for Gaussian data. The DCT basis is regarded as a good approximation of the KLT for natural image patches when a non-adaptive transform is required. Compared to the Fourier transform, the KLT is superior (by construction) in terms of representation efficiency. However, this advantage comes at the cost of a non-structured and substantially more complex transform. As we will see, this tradeoff between efficiency and suitability continues to play a major role in modern dictionary design as well.

Other method such as Independent Component Analysis (ICA) [6] are required to understand the different processes behind the observed data. ICA is able to separate the different causes or sources by analyzing the statistical characteristics of the data set and minimizing the mutual information between the observed samples. However, ICA techniques respect some orthogonality conditions such that the maximal number of causes is often limited to the signal dimension. ICA [6] differs from PCA because it is able to separate source not only with respect to the second order correlations in a data set, but also with respect to higher order statistics.

2.2.2 The Non-Linear Embedding Models

Increasing sparsity required departure from the linear model, towards a more flexible non-linear formulation. In the nonlinear case, each signal is allowed to use a different set of atoms from the dictionary in order to achieve the best approximation. Thus, the approximation process becomes

$$x \approx \sum_{n \in I_K(x)} c_n \phi_n \tag{2.4}$$

Where $I_K(x)$ is an index set adapted to each signal individually (we refer the reader to [4, 7], for more information on this wide topic). The non-linear view paved the way to the design of newer, more efficient transforms. In the process, many of the fundamental concepts guiding modern dictionary design were formed. Following the historic time

line, we trace the emergence of the most important modern dictionary design concepts, which are mostly formed during the last two decades of the 20th century.

- Kernel-trick:

To provide a natural nonlinear extension of PCA, one can apply a nonlinear transform to the data. Let $\phi(x)$ be the nonlinear transformation mapping data from the input space X to some feature space H. Then, with the covariance matrix associated to the transformed data, the resulting principal axes take the form

$$\psi_m = \sum_{n=1}^{N} \langle \phi(x_m), \psi_m \rangle_H \, \phi(x_m) \tag{2.5}$$

where $\langle x, y \rangle$ denotes the inner product in the feature space H. In this space, each feature ${\psi_m}_{m=1}^{M}$ lies in the span of the mapped input data, with the coefficients given by the m_{th} eigenvector of the eigenproblem

$$n\lambda_m \alpha_m = K\alpha_m \tag{2.6}$$

where K is the so-called Gram matrix with entries $\langle \phi(x_m), \phi(x_n) \rangle_H$, for $m, n = 1, 2, \cdots, N$. As illustrated here, the expansion coefficients require only the evaluation of the inner products. Without the need to exhibit the mapping function, this information can be easily exploited for a large class of nonlinearities by substituting the inner product with a positive semi-definite kernel function. This argument is the kernel trick, which provides a nonlinear counterpart of the classical PCA algorithm, the so-called kernel PCA [8].

In the past 15 years or so, a novel breakthrough for artificial neural networks has been achieved in the field of pattern recognition and classification within the framework of kernel-based machine learning. They have gained wide popularity owing to the theoretical guarantees regarding performance and low computational complexity in nonlinear algorithms. Pioneered by Vapnik's support vector machines (SVMs) for classification and regression [9], kernel-based methods are nonlinear algorithms that can be adapted to an extensive class of nonlinearities. As a consequence, they have found numerous applications, including classification [10], regression [11], time-series prediction [12], novelty detection [13], image denoising [14], and bioengineering [15], etc.

- Localization:

To achieve sparsity, transforms required better localization. Atoms with concentrated supports allow more flexible representations based on the local signal characteristics, and limit the effects of irregularities, which are observed to be the main source of large coefficients. In this spirit, one of the first structures to be used was the Short Time Fourier Transform (STFT) [16], which emerges as a natural extension to the Fourier transform. In the STFT, the Fourier transform is applied locally to (possibly overlapping) portions of the signal, revealing a time-frequency (or space-frequency) description of the signal. An example of the STFT is the JPEG image compression algorithm [17], which is based on this concept. During the 1980's and 1990's, the STFT was extensively researched and generalized, becoming more known as the Gabor transform, named in homage of Dennis Gabor, who first suggested the time-frequency decomposition back in 1946 [18]. Gabor's work was independently rediscovered of

$$G = \left\{\phi_{m,n}(x) = \omega(x - \beta m)e^{i2\pi\alpha nx}\right\}_{n,m\in Z} \tag{2.7}$$

where $\omega(x)$ is a low-pass window function localized at 0 (typically a Gaussian), and α and β control the time and frequency resolution of the transform.

In higher dimensions, more complex Gabor structures were developed which add directionality, by varying the orientation of the sinusoidal waves. This structure gained substantial support from the work of Daugman [19, 20], who discovered oriented Gabor-like patterns in simple-cell receptive fields in the visual cortex. These results motivated the deployment of the transform to image processing tasks, led by works such as Daugman [21] and Porat and Zeevi [22]. Today, practical uses of the Gabor transform are mainly in analysis and detection tasks, as a collection of directional filters. Figure 2.2 shows some examples of 2-D Gabor atoms of various orientations and sizes.

- Multi-Resolution:

One of the most significant conceptual advancements achieved in the 1980's was the rise of multiscale analysis. It was realized that natural signals, and images specifically, exhibited meaningful structures over many scales, and could be analyzed and described particularly efficiently by multi-scale constructions. One of the simplest and best known of such structures is the Laplacian pyramid, introduced in 1984 by Burt and Adelson

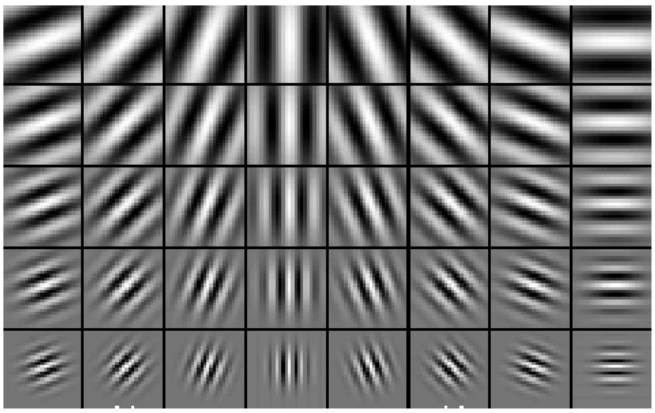

Figure 2.2: **Examples of Gabor atoms** - 12×12 blocks in different scales and orientations

[23]. The Laplacian pyramid represents an image as a series of difference images, where each one corresponds to a different scale and roughly a different frequency band.

In the second half of the 1980's, though, the signal processing community was particularly excited about the development of a new very powerful tool, known as wavelet analysis [8, 24, 25]. In a pioneering work from 1984, Grossman and Morlet [26] proposed a signal expansion over a series of translated and dilated versions of a single elementary function, taking the form

$$W = \left\{ \phi_{m,n}(x) = \alpha^{n/2} f(\alpha^n x - \beta m) \right\}_{n,m \in Z} \tag{2.8}$$

This simple idea captivated the signal processing and harmonic analysis communities, and in a series of influential works by Meyer, Daubechies, Mallat and others [27, 28], an extensive wavelet theory was formalized. The theory was formulated for both the continuous and discrete domains, and a complete mathematical framework relating the two was put forth. A significant breakthrough came from Meyer's work in 1985 [27], who found that unlike the Gabor transform (and contrary to common belief) the wavelet transform could be designed to be orthogonal while maintaining stability - an extremely appealing property to which much of the initial success of the wavelets can be attributed to.

Specifically of interest to the signal processing community was the work of Mallat and his colleagues [28, 29], which established the wavelet decomposition as a multi-resolution expansion and put forth efficient algorithms for computing it. In Mallat's

description, a multi-scale wavelet basis is constructed from a pair of localized functions referred to as the scaling function and the mother wavelet. The scaling function is a low frequency signal, and along with its translations, spans the coarse approximation of the signal. The mother wavelet is a high frequency signal, and with its various scales and translations spans the signal detail. In the orthogonal case, the wavelet basis functions at each scale are critically sampled, spanning precisely the new detail introduced by the finer level.

Non-linear approximation in the wavelet basis was shown to be optimal for piecewise-smooth 1-D signals with a finite number of discontinuities, see e.g. [30]. This was a striking finding at the time, realizing that this is achieved without prior detection of the discontinuity locations. Unfortunately, in higher dimensions the wavelet transform loses its optimality; the multi-dimensional transform is a simple separable extension of the 1-D transform, with atoms supported over rectangular regions of different sizes. This separability makes the transform simple to apply, however the resulting dictionary is only effective for signals with point singularities, while most natural signals exhibit elongated edge singularities. The JPEG2000 image compression standard, based on the wavelet transform, is indeed known for its ringing (smoothing) artefacts near edges.

- Adaptivity:

Going to the 1990's, the desire to push sparsity even further, and describe increasingly complex phenomena, was gradually revealing the limits of approximation in orthogonal bases. The weakness was mostly associated with the small and fixed number of atoms in the dictionary - dictated by the orthogonality - from which the optimal representation could be constructed. Thus, one option to obtain further sparsity was to adapt the transform atoms themselves to the signal content.

One of the first such structures to be proposed was the wavelet packet transform, introduced by Coifman, Meyer and Wickerhauser in 1992 [31]. The transform is built upon the success of the wavelet transform, adding adaptivity to allow finer tuning to the specific signal properties. The main observation of Coifman et al. was that the wavelet transform enforced a very specific time-frequency structure, with high frequency atoms having small supports and low frequency atoms having large supports. Indeed, this choice has deep connections to the behaviour of real natural signals. However, for

specific signals, better partitionings may be possible. The wavelet packet dictionary essentially unifies all dyadic time-frequency atoms which can be derived from a specific pair of scaling function and mother wavelet, so atoms of different frequencies can come in an array of time supports. Out of this large collection, the wavelet packet transform allows to efficiently select an optimized orthogonal sub-dictionary for any given signal, with the standard wavelet basis being just one of an exponential number of options. The process was thus named by the authors a Best Basis search. The wavelet packet transform is, by definition, at least as good as wavelets in terms of coding efficiency. However, we note that the multidimensional wavelet packet transform remains a separable and non-oriented transform, and thus does not generally provide a substantial improvement over wavelets for images.

- Geometric Invariance and Overcompleteness:

In 1992, Simoncelli et al. [32] published a thorough work advocating a dictionary property they termed shiftability, which describes the invariance of the dictionary under certain geometric deformations, e.g. translation, rotation or scaling. Indeed, the main weakness of the wavelet transform is its strong translation sensitivity, as well as rotation-sensitivity in higher dimensions. The authors concluded that achieving these properties required abandoning orthogonality in favor of over-completeness, since the critical number of atoms in an orthogonal transform was simply insufficient. In the same work, the authors developed an over-complete oriented wavelet transform - the steerable wavelet transform - which was based on their previous work on steerable filters and consisted of localized 2-D wavelet atoms in many orientations, translations and scales.

For the basic 1-D wavelet transform, translation-invariance can be achieved by increasing the sampling density of the atoms. The stationary wavelet transform, also known as the undecimated or non-subsampled wavelet transform, is obtained from the orthogonal transform by eliminating the sub-sampling and collecting all translations of the atoms over the signal domain. The algorithmic foundation for this was laid by Beylkin in 1992 [33], with the development of an efficient algorithm for computing the undecimated transform. The stationary wavelet transform was indeed found to substantially improve signal recovery compared to orthogonal wavelets, and its benefits

were independently demonstrated in 1995 by Nason and Silverman [34] and Coifman and Donoho [35].

2.2.3 From Transform to Dictionaries

By the second half of the 1990's, most of the concepts for designing effective transforms were laid out. At the same time, a conceptual change of a different sort was gradually taking place. In a seminal work from 1993, Mallat and Zhang [36] proposed a novel sparse signal expansion scheme based on the selection of a small subset of functions from a general over-complete dictionary of functions. Shortly after, Chen, Donoho and Saunders published their influential paper on the Basis Pursuit [37], and the two works signalled the beginning of a fundamental move from transforms to dictionaries for sparse signal representation. An array of works since has formed a wide mathematical and algorithmic foundation of this new field, and established it as a central tool in modern signal processing (see [38]). This seemingly minor terminological change from transforms to dictionaries enclosed the idea that a signal was allowed to have more than one description in the representation domain, and that selecting the best one depended on the task. Moreover, it de-coupled the processes of designing the dictionary and coding the signal: indeed, given the dictionary - the collection of elemental signals - different cost functions could be proposed in Eq. 2.1, and different coding methods could be applied. The first dictionaries to be used in this way were the existing transforms - such as the Fourier, wavelet, STFT, and Gabor transforms, see e.g. [36, 37]. As an immediate consequence, the move to a dictionary-based formalism provided the benefit of constructing dictionary mergers, which are the unions of several simpler dictionaries; these were proposed by Chen, Donoho and Saunders in [37], and provide a simple way to increase the variety of features representable by the dictionary.

2.2.4 Dictionary-learning

For the embedding model problems, the permanent question is: given the observed data, how to determine the subspace where the data lie? The choice of these subspaces is crucial for efficient dimensional reduction, but it is not trivial. This question commanded the new and promising research field called dictionary learning. It forced on the development of novel algorithms for building dictionaries of atoms or subspaces that provide efficient representations of classes of signals. Sparsity constraints are key

of most of the algorithms that solve the dictionary learning problems; they enforce the identification of the most important cause of the observed data and favour the accurate representation of the relevant information.

The dictionaries described so far all roughly fall under the umbrella of Harmonic Analysis, which suggests modelling interesting signal data by a simpler class of mathematical functions, and designing an efficient representation around this model. For example, the Fourier dictionary is designed around smooth functions, while the wavelet dictionary is designed around piecewise-smooth functions with point singularities. The dictionaries of this sort are characterized by an analytic formulation, and are usually supported by a set of optimality proofs and error rate bounds. An important advantage of this approach is that the resulting dictionary usually features a fast implicit implementation which does not involve multiplication by the dictionary matrix. On the other hand, the dictionary can only be as successful as its underlying model, and indeed, these models are typically over-simplistic compared to the complexity of natural phenomena.

2.2.4.1 Analytic Dictionaries

Analytic dictionaries are typically formulated as tight frames, meaning that $DD^T x = x$ for all x, and therefore the dictionary transpose can be used to obtain a representation over the dictionary. The analytic approach then proceeds by analyzing the behaviour of the filter-set $D^T x$, and establishes decay rates and error bounds.

The tight frame approach has several advantages. Analyzing the behaviour of D^T as an analysis operator seems easier than deriving sparsity bounds in a synthesis framework, and indeed, results obtained for the analysis formulation also induce upper bounds for the synthesis formulation. Another benefit is that - when formulated carefully - the algorithms for both analysis and synthesis operators become nearly reversals, simplifying algorithm design. Finally, the tight frame approach is beneficial in that it simultaneously produces a useful structure for both the analysis and synthesis frameworks, and has a meaningful interpretation in both.

- Curvelets:

The curvelet transform was introduced by Candes and Donoho in 1999 [39], and was later refined into its present form in 2003 [40]. When published, the transform astonished the harmonic analysis community by achieving what was then believed to be only possible with adaptive representations: it could represent 2-D piecewise-smooth functions with smooth curve discontinuities at an (essentially) optimal rate. The curvelet transform is formulated as a continuous transform, with discretized versions developed for both formulations [40, 41]. Each curvelet atom is associated with a specific location, orientation and scale. In the 2-D case, a curvelet atom is roughly supported over an elongated elliptical region, and is oscillatory along its width and smooth along its length. As it turns out, this property is useful for the efficient representation of smooth curves [42], and indeed several subsequent transforms follow this path. In higher dimensions, the curvelet atoms become flattened ellipsoids, oscillatory along their short direction and smooth along the other directions [40, 43].

- Contourlets:

The curvelet transform offers an impressively solid continuous construction and exhibits several useful mathematical properties. However, its discretization turns out to be challenging, and the resulting algorithms are relatively complicated. Also, current discretizations have relatively high redundancies, which make them more costly to use and less applicable for tasks like compression.

With this in mind, Do and Vetterli proposed the contourlet transform in 2002 [44, 45] as an alternative to the 2-D curvelet transform. The transform was later refined in 2006 by Lu and Do [46], and a multi-dimensional version, named surfacelets, was also recently introduced [47].

The contourlet transform implementation is based on a pyramidal band-pass decomposition of the image followed by a directional filtering stage. The main appeal of the transform is due to its simple discrete formulation, its low complexity and reduced redundancy. It should be noted, though, that while the transform is well suited for tasks such as compression, its aggressive sub-sampling has been noted to lead to artefacts in signal reconstruction, in which case a translation-invariant version of the transform is preferred [48, 49], indeed, this option significantly increases redundancy and complexity, though the simpler structure of the transform remains.

- Bandlets:

The bandelet transform was proposed in 2005 by Le Pennec and Mallat [50], with a second version introduced soon after by Peyre and Mallat [51]. The bandelet transform represents one of the most recent contributions in the area of signal-adaptive transforms, and as such it differs fundamentally from the non-adaptive curvelet and contourlet transforms.

The idea behind the bandelet construction is to exploit geometric regularity in the image - specifically edges and directional phenomena - in order to fit a specifically optimized set of atoms for the image. The original bandelet construction operates in the spatial domain, and is based on an adaptive subdivision of the image to dyadic regions according to the local complexity; in each region, a set of skewed wavelets is matched to the image flow, in such a way that the wavelet atoms essentially "wrap-around" the edges rather than cross them. This process significantly reduces the number of large wavelet coefficients, as these typically emerge from the interaction of a wavelet atom and a discontinuity.

The resulting set of atoms forms a (slightly) over-complete set, which is specifically tailored for representing the given image. In the second bandelet construction, which is formulated in the wavelet domain, the transform is further refined to produce an orthogonal set. In terms of dictionaries, the bandelet transform selects a set of atoms from a nearly infinite set, and in fact discretization is the main source for limiting the size of this set. This is as opposed to the wavelet packet transform, for instance, where the complete set of atoms is not much larger than the signal dimension.

- Other Analytic Dictionaries:

Many additional analytic transforms have been developed during the past decade, some of which we mention briefly. The complex wavelet transform [52, 53], is an oriented and near-translation-invariant high-dimensional extension of the wavelet transform, achieved through the utilization of two mother wavelets satisfying a specific relationship between them. Similar to the original wavelet transform, the complex wavelet transform is efficient and simple to implement, and the added phase information delivers orientation sensitivity and other favourable properties. The shearlet transform

[54] is a recently proposed alternative to curvelets, which utilizes structured shear operations rather than rotations to control orientation. Similar to curvelets, the shearlet transform is based on a comprehensive continuous mathematical construction, and it shares many of the properties of the curvelet transform while providing some attractive new features.

Recent adaptive dictionaries include the directionlet transform [55], which is a discrete transform which constructs oriented and anisotropic wavelets based on local image directionality, utilizing a specialized directional grouping of the grid points for its numerical implementation. Finally, the grouplet transform [56] is a multi-scale adaptive transform which essentially generalizes Haar wavelets to arbitrary supports, based on image content regularity; when applied in the wavelet domain, the transform bears some resemblance to the second-generation bandelet transform, and thus is referred to as grouped bandelets.

2.2.4.2 Dictionary Training Methods

The goal of sparse representation is to express a given signal y of dimension d as a linear combination of a small number of signal taken from a "resource" database, which is called the dictionary. Elements of the dictionary are typically unit norm functions called atoms. Let us denote the dictionary as D and the atoms as $\phi_n, n = 1, 2, \cdots, N$, where N is the size of the dictionary. The dictionary is over-complete ($d << N$) when it spans the signal can be represented as a linear combination of atoms in the dictionary

$$y = \Phi a = \sum_{n=1}^{N} a_n \phi_n \tag{2.9}$$

Because the dictionary is over-complete, a is not unique. This is where the sparsity constraint comes into play. To achieve efficient and sparse representations, we generally relax the requirement for finding the exact representation. We look for a sparse linear expansion with an approximation error of bounded energy ϵ. The objective is now to find a sparse vector a that contains a small number of significant coefficients, while the rest of the coefficients are closer or equal to zero. In other words, we want to minimize the resources (atoms) that we use to accomplish the task of signal representation. The optimization problem can be formulated as follows:

$$\min_{a} \|a\|_0 , \quad \text{subject} \quad \text{to} \quad \|y - \Phi a\|_2^2 < \epsilon \tag{2.10}$$

where $\|x\|_p$ denotes the l_p norm. Unfortunately, this problem is NP-hard. However, there exist polynomial time approximation algorithms that find a suboptimal solution for the sparse vector a. These algorithms can be classified in two main groups. The first group includes greedy algorithms such as the matching pursuit (MP) [36] and the orthogonal MP (OMP) [57], which iteratively select locally optical basis vectors. In the second group, we find algorithms based on convex relaxation methods such as the basis pursuit denoising [37] or least absolute shrinkage and selection operator (LASSO) [58], which solve the following problem:

$$\min_{a} \|y - \Phi a\|_2^2 + \lambda \|a\|_1 \tag{2.11}$$

The convex relaxation permits to replace the non-convex l_0 norm in the original problem by the convex l_1 norm. The l_0 norm of a vector is equal to the number of nonzero elements in that vector. It is called a "norm" because it is the limit of p-norm as p approaches zero. However, note that it is not a true norm, unlike the l_1 norm that has all properties of a norm. Besides pursuit algorithms, there exist other sparse approximation algorithms such as the focal underdetermined system solver (FOCUSS) [59] and sparse Bayesian learning [60], for example. A recent review of the sparse recovery algorithms can be found in [61]. The performance of these algorithms in terms of the approximation quality and the sparsity of the coefficient vector a depends not only on the signal itself, but also on the over-complete dictionary D. Once the algorithms are used on a specific class of signals y, we easily understand that not all dictionaries are able to lead to sparse solution. These are the dictionaries that include atoms explaining best the causes of the target data set. It is exactly the goal of dictionary training methods to find such optimized dictionaries.

The research in dictionary learning has followed three main directions that correspond to three categories of algorithm: 1) the probabilistic learning methods; 2) the learning methods based on clustering or vector quantization; 3) the methods for learning dictionaries with a particular construction. The construction is typically driven by priors on the structure of the data or to the target usage of the learned dictionary. This

section presents the main principles of representative algorithms in each of these three dictionary learning categories.

- Probabilistic Methods:

Representation and coding of images have always been a great challenge for researchers because of the high dimensionality and complex statistics of such signals. Thus, it is not surprising that one of the earliest works addressing the problem of learning over-complete dictionaries appears exactly for image representation. In 1997, Olshausen and Field [62] developed a maximum likelihood (ML) dictionary learning method for natural images under the sparse approximation assumption. Their method is called sparse coding. Given the linear generative image model in Eq. 2.12, the objective of the ML learning method is to maximize the likelihood that natural images have efficient, sparse representations in a redundant dictionary given by the matrix Φ. Formally, the goal of learning is to find the over-complete dictionary Φ^* such that

$$\Phi^* = arg\max_{\Phi} [logP(y|\Phi)] = arg\max_{\Phi} \left[log \int_a P(y|a,\Phi)P(a)da \right] \tag{2.12}$$

The probabilistic inference approach in over-complete dictionary learning has subsequently been adopted by researchers. The two-step optimization structure has been preserved in most of these works, and the modifications usually appeared in either the sparse approximation step, or the dictionary update step, or in both. For example, the method of optimal directions (MOD) algorithm [63] optimizes iteratively the same objective ML function as in sparse coding. However, it uses the OPM algorithm to find a sparse vector a and introduces a closed-form solution for the dictionary update step. The two modifications render the MOD approach faster compared to the method of Oshausen and Field, but still does not guarantee to find the globally optimal solution. Moreover, it is not guarantee to converge, neither to decrease the objective function at each iteration. The maximum a posterior (MAP) dictionary learning method [64] belongs also to the family of two-step iterative algorithms based on probabilistic inference. Instead of maximizing the likelihood $P(y|\Phi)$, the MAP method maximizes the posterior probability $P(\Phi, a|y)$. This essentially reduces to the same two-step algorithm, where dictionary update includes an additional constraint on the dictionary that can be for example the unit Frobenius norm of Φ or the unit l_2 norm of all atoms

in the dictionary. The sparse approximation step is here performed with FOCUSS [65]. Finally, the majorization method can also be used to minimize the objective function in both sparse approximation and dictionary update steps [66]. The sparse approximation step then reduces to the use of an iterative threshold algorithm.

Naturally, the two assumptions introduced in the sparse coding method represent constraints that can be modified or even removed to learn better dictionaries or to extend the method to other signal models. Lewicki and Sejnowski have modified the first assumption and proposed a new way to approximate the integral in eq.2.12 with a Gaussian around the posterior estimate of the coefficient vector a. This changes the update rule in the learning step [67]. They have shown that the ML dictionary learning method with the new estimate for $P(y|\Phi)$ learns dictionaries that improve the efficiency of sparse coding. The efficiency is measured here in terms of the entropy of data given the over-complete dictionary. This method actually represents a generalization of the independent component analysis (ICA) method to over-complete dictionaries. In general, convergence is not guaranteed for the l_1-constrained method, although it can be proved in some conditions [68]. One should also introduce smoother sparsity priors to obtain more stable solutions. For example, the l_1 constraint is replaced by a Kullback-Leibler (KL) divergence in [69], which shows that the sparsity is preserved, while the KL-regularization leads to efficient convex inference and stable coefficient vectors.

- Clustering-based Methods:

A slightly different family of dictionary learning techniques is based on vector quantization (VQ) achieved by K-means clustering. The VQ approach for dictionary learning has been first proposed by Schmid-Saugeon and Zakhor in MP-based video coding [70]. Their algorithm optimizes a dictionary given a set of image patches by first grouping patterns such that their distance to a given atom is minimal, and then by updating the atom such that overall distance in the group of patterns is minimal. The implicit assumption here is that each patch can be represented by a single atom with a coefficient equal to one, which reduces the learning procedure to a K-means clustering. Since each patch is represented by only one atom, the sparse approximation step becomes trivial.

A generalization of the K-means algorithm for dictionary learning, called the K-SVD algorithm, has been proposed by Aharon et al. [71]. After the sparse approximation

step with OMP, the dictionary update is performed by sequentially updating each column of Φ using a singular value decomposition (SVD) to minimize the approximation error. The update step is hence a generalized K-means algorithm since each patch can be represented by multiple atoms and with different weights. The algorithm is not guaranteed to converge in general. However, in practice, dictionaries learned with K-SVD have shown excellent performance in image denoising.

- Training Dictionaries with Specific Structures:

Many applications do not require general forms of dictionary atoms but can rather benefit from a dictionary that is a set of parametric functions. In contrary to the generic dictionaries above, the advantage of parametric dictionaries reside in the short description of the atoms. The generation function and the atom parameters are sufficient for building the dictionary functions. This is quite beneficial in terms of memory requirements, communication costs or implementation complexity in practical applications.

Such generating functions can be built on prior knowledge about the form of signal causes or the target task. For example, some perceptual criteria can drive the choice of the generating functions in building the dictionary atoms, when the objective is to reconstruct data that are eventually perceived by the human auditory or visual system. Learning in such parametric dictionaries reduces to the problem of learning the parameters for one or more generating functions. Equivalently, it consists in finding a good discrete parametrization that leads to efficient sparse signal approximations. Parametric dictionaries are usually structured, so one can enforce some desire dictionary properties during learning such as minimal dictionary coherence; for example, one can optimize a parametric dictionary such that it gets close to an equiangular tight frame (ETF). In [72], a dictionary for audio signals is learned based on a Gammatone generating function, which has been shown to have similarities with the human auditory system. The method learns a dictionary with good coherence properties, which tiles the time-frequency plane more uniformly than the original Gammatone filter bank.

Priors or models of the underlying signal causes can also lead to imposing properties such as shift-invariance [73] or multi-scale [74] characteristics of the atoms. Such constraints typically limit the search space in the dictionary optimization problem,

but lead to more accurate or task-friendly representations. Similarly, the target dictionary might present specific characteristics in particular recovery problems, such as a block-based structure [75], or orthogonality between subspace [76]. These requirements considerably affect the design of learning strategies as well as the approximation performance.

2.2.5 A summary of the state-of-the-arts for embedding models

In this section, we summary the previous content of this chapter in the table 2.1, to highlight the properties of the related works.

2.3 Conclusions

Embedding model has significantly evolved over the past decades, beginning with simple orthogonal transforms and leading to the complex over-complete analytic and trained dictionaries now defining the state-of-the-art. Substantial conceptual advancement has been made in understanding the elements of an efficient dictionary design - most notably adaptivity, multi-scale, geometric invariance, and over-completeness. However, with a wealth of tools already developed, much work remains to be done; indeed, the various components have yet to be neatly merged into a single efficient construct. Many future research directions have been mentioned in the text, and demonstrate the viability and vividness of the field as well as the large number of challenges that still awaiting.

In the following chapter of this manuscript, we present our studies which pursue the development of the embedding models: from the linear transform to the non-linear transform, from the orthogonal basis to sparse pre-learned dictionaries. We present methods for determining the proper representation of data sets by means of the reduced dimensionality subspaces, which are adaptive to both the characteristics of the signals and the processing task hand. We describe methods for learning dictionaries that are appropriate for the representation of given classes of signals and multi-sensor data. We further show that dimensionality reduction based on dictionary representation can be extended to address specific tasks such as data analysis or classification when the learning includes a class criteria in the objective function. The benefits of non-linear structures and dictionary learning clearly show that a proper understanding of the

Linear Signal Transforms	FFT	strength	Strong smoothing and noise-reducing effect, efficient at describing uniformly smooth signals.
		weakness	Lack of localization makes it difficult to represent discontinuities.
	DCT	strength	Realize continuous boundaries, more efficient on approximation than FFT.
		weakness	Not robust for localization information.
	PCA	strength	Adaptive to represent signals coming from a certain known distribution, free to applied to any kind of data.
		weakness	Process models the data as coming from a low-dimensional Gaussian distribution, and thus is most effective for Gaussian data.
	ICA	strength	Able to separate source not only with respect to the second order correlations in a data set.
		weakness	Respect some orthogonality conditions such that the maximal number of causes is often limited to the signal dimension.
Non-linear Transforms	Kernel -trick	strength	Using a kernel, the originally linear operations of PCA are done in a reproducing kernel Hilbert space with a non-linear mapping.
		weakness	Hard to reconstruct from the feature space.
	Gabor Transform / STFT	strength	Atoms with concentrated supports allow more flexible representations based on the local signal characteristics, and limit the effects of irregularities.
		weakness	High computation complexity.
	Wavelet Transform	strength	The wavelet basis was shown to be optimal for piecewise-smooth 1-D signals without prior detection of the discontinuity locations.
		weakness	For higher dimension data the wavelet transform loses itsoptimality, the dictionary is only effective for signals with point singularities.

Analytic Dictionary	Contourlets	strength	With simple discrete formulation,low complexity and reduced redundancy.
		weakness	The aggressive sub-sampling of contourlets has been noted to lead to artifacts in signal reconstruction.
	Bandlet	strength	Bandlets significantly reduces the number of large wavelet coefficients, as these typically emerge from the interaction of a wavelet atom and a discontinuity.
		weakness	Challenging on discretizations.
	Curvelets	strength	The curvelet transform is formulated as a continuous transform, with discretized versionsdeveloped for both formulations. Each curveletatom is associated with a specific location, orientationand scale.
		weakness	In higher dimensions, the curvelet atoms become flattened ellipsoids, oscillatory along their short direction and smooth along the other directions.
Dictionary Training Methods	Maximum Likelihood (ML) dictionary -learning	strength	Common technology, intuitive solution.
		weakness	Less efficient.
	Bandlet	strength	Dictionaries learned with K-SVD have shown excellent performance in image denoising.
		weakness	The algorithm is not guaranteed to converge in general.

Table 2.1: A summary of the state-of-the-arts for embedding models

sensed world is the key to task-specific representation of relevant information in high-dimensional data tasks.

3

Application of non-linear embedding models on Active Appearance Models

Illumination condition and facial pose have an explicit effect on the performance of face recognition systems, caused by the complicated non-linear variations between feature points and views. Solution of such difficulty requires an appropriate non-linear embedding model for building a powerful deformable model. The problem stated in this thesis for the facial segmentation in the complex illumination condition and with multiple face rotations, is dealt with in this chapter. It presents our contribution of an efficient embedding technique for Active Appearance Models, to make a robust face segmentation system. Based on the statistical analysis of the illuminated face data, we propose two non-linear embedding models (Probabilistic PCA and Kernel Similarity PCA) for building the appearance model.

Section 3.1 provides a brief introduction of deformable model based facial algorithms. Section 3.2 describes in detail classical AAM, followed by improvements and expansions made in this domain. Section 3.3 presents the solutions to the problem stated in this chapter. Its subsection 3.3.1 enumerates the motivations of using the non-linear embedding models, while subsection 3.3.2 and 3.3.3 concentrate on the theory of Probabilistic PCA (PPCA) and Kernel Similarity PCA (KS-PCA) respectively. Section 3.3.4 provides the experiments that performed to validate the proposed embedding model performance on two typical databases (IMM Face Database and CMU PIE

database) for evaluating the novel facial deformable models. The face segmentation results from the proposed models are compared with the results from classical AAM, to validate the efficiency, accuracy and robustness achieved.

3.1 Introduction

The deformable model gained a lot of interest in the last decade and researchers have proposed its various versions. This section provides a brief introduction to the deformable model based methods and focuses on the AAM similar deformable methods.

3.1.1 Active Contour Models

Active contour model [77], also called snakes, is a framework for delineating an object outline from a possibly noisy 2D image. This framework attempts to minimize an energy associated to the current contour as a sum of an internal and external energy:

- The *external energy* is supposed to be minimal when the snake is at the object boundary position. The most straightforward approach consists in giving low values when the regularized gradient around the contour position reaches its peak value.

- The *internal energy* is supposed to be minimal when the snake has a shape which is supposed to be relevant considering the shape of the sought object. The most straightforward approach grants high energy to elongated contours (elastic force) and to bended/high curvature contours (rigid force), considering the shape should be as regular and smooth as possible.

The snakes model is popular in computer vision, and led to several developments in 2D and 3D. In two dimensions, the active shape model [78] represents a discrete version of this approach, taking advantage of the point distribution model to restrict the shape range to an explicit domain learned from a training set.

One may visualize the snake as a rubber band of arbitrary shape that is deforming with time trying to get as close as possible to the object contour. Snakes do not solve the entire problem of finding contours in images, but rather, they depend on other mechanisms like interaction with a user, interaction with some higher level image understanding process, or information from image data adjacent in time or space. In

general, Snake is placed near the object contour. It will dynamically move towards object contour by minimizing its energy iteratively.

3.1.2 Elastic Bunch Graphe Matching

In [79], Wiskott et al. proposed an algorithm called Elastic Bunch Graph Matching for a task of recognizing persons from single images by reference to a gallery. The problem was to address image variation due to differences in facial expression, head pose, position, and size. As Wavelet embedding is robust to moderate lighting changes and small shifts and deformations, while Model graphs can easily be translated, scaled, oriented, or deformed during the matching process, thus compensating for a large part of the variance of the images.

In this work, the basic object representation is the labelled image. In the deformable grid, edges are labelled with distance information and nodes are labelled with wavelet responses locally bundled in jets, where each component of a jet is a filter response of a specific Gabor wavelet extracted at a given image point.

In segmentation a function is used to evaluate the graph similarity between an image graph and the FBG. It depends on the jet similarities and the distortion of the image grid relative to the FBG grid. The goal of this function is to find the fiducial points and thus to extract from the image a graph which maximizes the similarity with the FBG.

This Face Bunch Graphs model is effective for face tracking, but is not able to synthesize the appearance of the face being tracked.Moreover it is not able to recover from tracking errors caused by temporary occlusion of features.

3.1.3 3D Morphable Models

Blanz and Vetter [80] introduced a deformable model called 3D Morphable Model (3DMM). Learning database of 3DMM was created by the laser scans of 200 heads of young adults (100 male and 100 female). The laser scans provide head structure data in a cylindrical representation, with radii of surface points sampled at 512 equally-spaced angles, and at 512 equally spaced vertical steps. Additionally, the RGB-colour values, were recorded in the same spatial resolution and were stored in a texture map with 8 bit per channel. The resultant faces were represented by approximately 70,000 vertices

and the same number of colour values. The morphable model is based on a data set of these 3D faces.

In segmentation phase they use gradient descent algorithm for the estimation of the parameters for a given input image. The reconstructed 2D image of the model is supposed to be closest to the input image in terms of Euclidean distance.

Robustness of these 3DMM is very high compared to other deformable model based method, but its unavoidable drawback is its computational time. According to Blanz and Vetter [81] each face requires more than 4 minutes in a 2GHz Pentium 4 processor. No matter how fast the system is, it requires enormous amount of time to process 70,000 vertices. Therefore they can not be implemented in a real time scenario. Moreover the method of obtaining 3D shapes and textures by laser scanners is cumbersome and expensive due to the requirement of additional hardware.

3.1.4 Active Shape models

Cootes, Lanitis and Taylor et al. [78, 82] have shown that the 2D shape appearance of objects can be modelled using Active Shape Models (ASM). An ASM consists of a Point Distribution Model (PDM) aiming to learn the variations of valid shapes, and a set of deformable models capturing the grey-levels around a set of landmark feature points. It is closely related to the Active Appearance Model. It is also known as a Smart Snakes method, since it is an analog to an active contour model which would respect explicit shape constraints.

In ASM, a shape is described by n points, it can be represented by n landmark points for a single example as the $2n$ element vector s as

$$s = (x_1, y_1, \cdots, x_n, y_n)^T \tag{3.1}$$

All the shapes of the learning database faces are aligned and a mean shape is obtained. A well-known data embedding technique PCA (Principal Component Analysis) is applied on these shapes to obtain shape parameters b as

$$s = \bar{s} + \phi * b \tag{3.2}$$

where ϕ are the eigenvectors and the vector b defines a set of parameters of a deformable model. By varying the elements of b we can vary the shape s. An active shape

model is described by the shape parameters b combined with similarity transformation (pose parameters) defining the rotation θ, translation X_t, Y_t and scale s of the model.

$$s = T_{X_t,Y_t,s,\theta}(\bar{s} + \phi * b) \tag{3.3}$$

During training phase sampling of the k pixels on either side of the model point is done for every training image. Instead of gray level absolute values, the derivatives of these values are sampled and normalized. During segmentation phase sampling of the $m(m > k)$ pixels on either side of the predicted model point is performed. Then the quality of the fit is tested by comparing them with the gray level model obtained in the training phase. Ultimately the one which gives the best match is chosen.

ASM are robust to illumination variations since they do not involve facial textures at all. On the contrary this becomes one of their drawbacks because textures play an important role in facial analysis. While this approach can be used to model and recover some changes in the shape of an object, it can only cope with largely linear variations. Nonlinear variations caused by changes in viewpoints and self-occlusions from different hand gestures had to be captured through the use of five different models [83].

3.2 Active Appearance Models

The Active Appearance Models introduced by [84] and [85] in 1998, is the deformable models composed of both shape and texture. This section will describe the classical AAM for face analysis, followed by improvements and expansions made in this domain.

3.2.1 Classic Active Appearance Models

The classical Active Appearance Models works in three phases. In the first phase, the model is generated from examples of faces on which points are marked manually and their textures are extracted. All these points and textures are combined and their variations are learned automatically from a principal component analysis. In the second phase (also called as training or pre-computation phase) the model is trained to pre-compute a matrix which helps to find the optimum values of variations with respect to the query images in the segmentation phase. In the third phase it uses its training data for the segmentation of the objects in the query images. Following sections will present the three phases of the AAM algorithm applied on facial image.

Shape In the first phase, AAM model is generated along with the deformation parameters. On each facial image of this database, set of points are marked manually. Different researchers have used different number of points on the face. Some of them has included ear while others have surrounded features like nose, eyebrows and ears by bunch of points. The work presented in this thesis use only 58 points in order to make AAM model time and memory efficient. These 58 point are shown in the figure 3.1: 8 points for the mouth, 11 points for the nose, 16 points for both eyes, 10 points for both eyebrows, and 13 points for the chin. A total of seven point paths were used; three closed and four open. All annotations were formatted in ASF. Combination of these 58 points on each face is regarded as a shape. If there are N number of images in the database then the vector representation of these shapes is:

$$s_i = [x_{i,1}, y_{i,1}, x_{i,2}, y_{i,2}, \cdots, x_{i,58}, y_{i,58}] \, (1 \leq i \leq N) \tag{3.4}$$

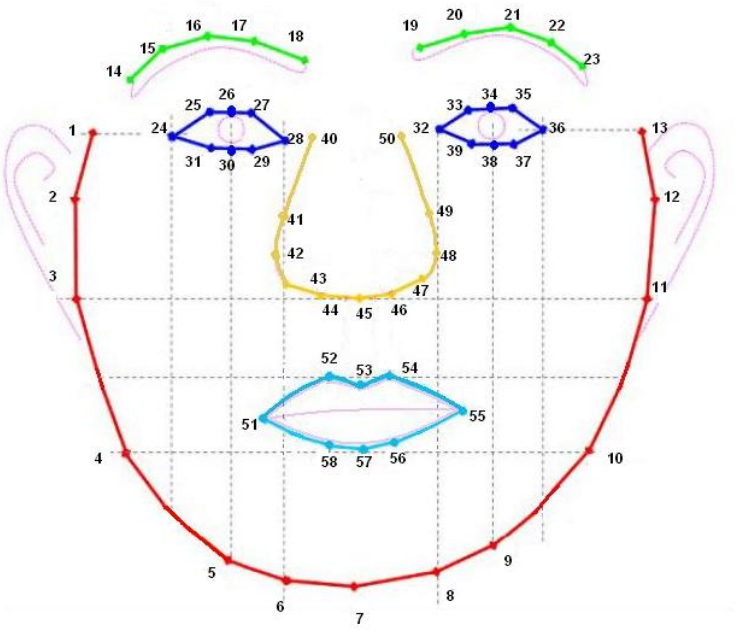

Figure 3.1: **Shape annotation image** - 58 landmarks

All the shapes obtained are rotated, resized and translated using Procrustes analysis [86]. The mean of each point is calculated to create mean shape of 58 points. The mean

shape obtained is used to extract and warp the frontal view textures of all the facial images using the Delaunay triangulation as shown in the figure 3.2. These textures undergo the procedure of photometric normalization to normalize their gray levels. Let us suppose the texture image is the texture of the learning database, then the equation for its normalization would be:

$$g_{image-normalize} = \frac{g_{image} - \beta}{\alpha} \tag{3.5}$$

$$\alpha = \sqrt{\sum_{i=1}^{m}(g_{image}(i) - \overline{g_{image}})^2} \tag{3.6}$$

$$\beta = \overline{g_{image}} \tag{3.7}$$

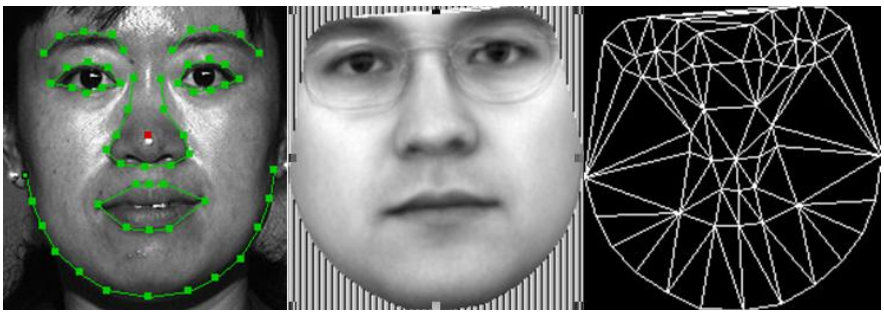

Figure 3.2: **Shape, Texture and Delaunay Triangulation** -

where α is the standard deviation and β is the mean of the pixels of the texture. Principal Component Analysis (PCA) compression is applied on the shapes and textures, to obtain shape and texture parameters with 95% of the variation retained. Each shape s_i and the texture g_i of the learning database can be synthesized by these shape and texture parameters with the help of the following equations.

$$s_i = \bar{s} + \phi_s * b_s \tag{3.8}$$

$$g_i = \bar{g} + \phi_g * b_g \tag{3.9}$$

where $\bar{s}$ and $\bar{g}$ are the mean shape and mean texture; ϕ_s and ϕ_g are the shape and texture eigenvectors obtained during PCA; b_s and b_g are the shape and texture parameters respectively.

Both of the above parameters are combined by concatenation of b_s and b_g. And a final PCA is performed to obtain the appearance parameters C.

$$b = [b_s b_g]^T, b = \phi_C * C \tag{3.10}$$

where ϕ_C are the eigenvectors obtained by retaining 95% of the variation and C is the matrix of the appearance parameters, which are used to obtain shape and texture of each face of the database.

AAM model can be translated as well as rotated with the help of pose vector P.

$$P = [\theta, t_x, t_y, Scale]^T \tag{3.11}$$

where θ corresponds to the face rotation and t_x, t_y are the offset values from the supposed origin and Scale is the magnification of the model.

AAM training The AAM model obtained in the previous section can be used directly for the face search or the search can be directed with the help of the matrix pre-computed by training images. Although direct application has some advantages over the trained AAM but for that an efficient optimization technique (e.g. gradient descent, genetic algorithm, Nelder Mead simplex etc.) is required. In the classical method of AAM, model is trained by applying it on the training images while introducing variations in all the parameters one by one. Residual images, which correspond to the difference between the model and the training image, are obtained for each parameter variation. As explained in previous, each image of the learning base can be synthesized by a particular value of parameters C and P. Let C_i be the value of appearance parameters of the image i of the learning database and P_i be the value of pose parameters. By changing the parameters C_i and P_i respectively, with δC and δP ($C = C_i + \delta C \quad and \quad P = P_i + \delta P$), a new shape s_m and a new texture g_m (eq. 3.10) are synthesized.

Let's consider the texture g_i as the texture of the original image i then pixel difference or the residual image is given as $\delta g = g_i - g_m$. By varying each parameter at a time and generating its residual images one can create a linear relation between them.

This relation is created through principal component regression technique by [85], in order to keep its dimensionality to a feasible size. The relation R_C between δc and δg and relation R_P between δc and δg are given as

$$\delta C = R_C * \delta g \tag{3.12}$$

$$\delta P = R_P * \delta g \tag{3.13}$$

where regression matrices R_C and R_P are of the size Number of C $\times$ Number of Pixels and Number of P $\times$ Number of Pixels respectively.

In later publication by Cootes et al. [84] and Cootes and Taylor [87] this principal component regression is superseded by a simpler approach. They calculated the partial differential of residual images with respect to each parameter and taking arithmetic mean of these partial differentials for all the training images and k variations in each parameter. This learning approach is denoted as Jacobian and is given as

$$\frac{\partial \delta g}{\partial P_j} = \frac{1}{M} \sum_{r}^{M} \sum_{k} \frac{\delta g_r(P_j + \delta P_{jk}) - \delta g_r(P_j + \delta P_{jk})}{2\delta P_{jk}} \tag{3.14}$$

where M is the number of training images, k is the variation on the j_{th} pose parameters P. Similarly Jacobian are calculated for each C parameters. Now R_P and R_C are calculated as

$$R_p = \left(\frac{\partial \delta g}{\partial P}\right)^{-1} \tag{3.15}$$

$$R_C = \left(\frac{\partial \delta g}{\partial C}\right)^{-1} \tag{3.16}$$

To obtain numerical stability, a singular value decomposition (SVD) of the Jacobian matrices ($\frac{\partial \delta g}{\partial P}$ and $\frac{\partial \delta g}{\partial C}$) are preferred in order to obtain their respective pseudo-inverse R_P and R_C. However due to the size this is not feasible, therefore a normal matrix inversion is carried out.

This approach is no doubt efficient as far as training is concerned especially when the number of training images and/or number of pixels of residual images becomes large enough such that it become impossible to store them in order to apply principal component regression method. Thus this method of training is easier to implement,

faster to calculate and requires far less memory to execute. Stegmann [88] observed that this training scheme do not differ significantly, from linear regression, in performance during segmentation. In fact Jacobian matrices are lightly better due to smaller computational and memory demands.

Segmentation In segmentation phase the deformed, rotated and translated shape model obtained by varying C and P parameters, is placed on the query image I to warp the face to mean frontal shape. After this shape normalization photometric texture normalization is applied to overcome illumination variations. The objective is to minimize pixel error

$$e = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[I_i(C,P) - M_i(C)]^2} \tag{3.17}$$

where $I(C,P)$ is the segmented image and $M(C)$ is the model obtained by C parameters and N is the number of pixels of the model. C and P parameters are calculated by using the relations of equations 3.15 and 3.16.

The searching algorithm is described below.

1. Generate g_m and s from the values of parameters C and P (initially set to 0).

2. Compute g_i, which is obtained by placing the shape s and warping the query image segment to mean frontal shape followed by texture normalization.

3. Calculate the residual image $\delta g_0 = g_i - g_m$, and the residual error $E_0 = |\delta g_0|$.

4. Predict $\delta C_0 = R_C * \delta g_0$ and $\delta P_0 = R_P * \delta g_0$.

5. Find new value of residual error $E_j < E_{j-1}$ with the variations predicted in appearance $C_j = C - k * \delta C_0$ and pose $P_j = P - k * \delta P_0$ parameters. Where k represents the discrete step sizes of 0.25, 0.5, 0.75 and 1.0.

6. Repeat steps from 1 to 5 while $E_{j-1} > E_j$, where E_{j-1} is the residual error of the previous iteration.

When the convergence of the error E_j is reached, i.e. when $E_{j+1} \geq E_j$, parameters C_j and P_j corresponds to the best parameters for the representations of the texture and shape of the face in a query image. A brief research work on the methods presented in this section was required to search for a method which would be more robust to the deformations of a face, so that facial features and pose of an unknown face could be extracted more efficiently. For this kind of application AAM methods are the most suitable approaches. Although [85] reported that the results of ASM is better than

AAM for marker detection, but we feel that significant information is embedded in the texture of face e.g. skin wrinkles, identity etc. Therefore it is necessary to take into account all the texture instead of specific patch around a facial feature in the case of ASM. Moreover in our work, we address the problem of face recognition, face synthesis and face compression for irregular illuminated conditions, which requires to include texture information. As far as other methods are concerned, AAM is more rapid than 3DMM and can make the use of 2D images obtained from a camera instead of using laser scanner. It has been widely used in various applications of lip-reading, cloning and expression detection etc. This section focuses on the method of Active Appearance Models (AAM) and next section presents the state of the art for AAM.

3.2.2 Advancement of AAMs

AAM algorithm has proved to be a successful method for matching the model to the query images. Since the classical AAM was described there have been a number of modifications and improvements proposed by several researchers, claiming to be superior than classical AAM. This section and its subsections give a detailed description of these improvements.

In this section we present a number of proposed improvements and alternatives to the original classical AAM, including different training methods, non-linear models and different methods of updating the model during the search. The performances of Shape-AAM, Nonlinear AAM, TB-AAM and compositional AAM are compared with that of the classical AAM.

Subsequent sections will focus on the methods adapted in AAM, which may be used for the solutions of the problems stated in this thesis i.e. pose estimation, features extraction of an unknown oriented face. These approaches can be divided into the three following subsections. Subsection 3.2.2.1 presents the work done, for the facial analysis of the oriented face, by the extension of the AAM model and its appearance parameters. Subsequent subsection 3.2.2.2 describes the facial analysis by fitting AAM on temporally synchronized multiple images acquired from two, three or multiple cameras. Subsection 3.2.2.3 presents the first approach to estimate the facial pose and features by the extensions of the AAM model and its appearance parameters.

3.2.2.1 AAMs Variants

The classical optimization of AAM by linear regression presents some drawbacks. It needs to store the matrices RC and RP in memory for the segmentation phase. Moreover the pre-computations from the training set is only an approximation for any given target image, and may be a poor one if the target image is significantly different (unknown or unseen faces) from the training images, as discussed and tackled by Cootes and Taylor [89]. A number of improvements and alternatives to the original classical AAM proposed by the research teams are discussed in this section.

Shape-AAM The Shape-AAM has been proposed by Cootes et al. [90]. It's an alternative approach to use residual image to compute the shape and pose parameters, while the texture parameters are computed directly from the image with the help of the shape model. The statistical model is created on the shape and texture, without the concatenation of two parameters (3rd PCA). Instead of appearance regression matrix of R_C, shape regression matrix R_s is calculated along with pose regression matrix R_P. This method is more time-consuming [91] than the classical method of AAM, and may be useful when there are few shape modes and many texture modes. **DAM** The direct appearance model, introduced by Hou et al. [92], also removes the 3rd PCA in classical AAM modelization, they do not combine shape and texture parameters of AAM. Unlike the Shape-AAM of Cootes et al. [90], they use information of the texture instead of the shape. They considered that the texture and shape are sufficiently correlated and the shape can be obtained from the texture of the model through a relationship built during the learning phase of DAM:

$$b_x = Sb_g \tag{3.18}$$

Therefore texture regression matrix R_g and pose regression matrix R_P are calculated in the training phase. The segmentation procedure is similar to classical AAM except they evaluate shape from the texture by the equation 3.18. Although it gives better results than the classical AAM method of [91], but it requires prior information of the correlation of texture and shape. Therefore this texture and shape linearity makes it difficult to analyze unknown oriented faces.

TB-AAM Lee and Kim [93] gave a new concept of Tensor-Based AAM. This concept is based on the tensor which is also known as n-way array or multidimensional matrix or n-mode matrix. It is a higher order generalization of a vector (first order tensor)

and a matrix (second order tensor). They incorporated a series of learning databases images with different identities, expression, pose and illumination variations etc. To include these variations by specific basis vectors they make the use of multi-linear algebra of Alex et al. [94] for the multi-linear analysis of the images with variations defined above. For the fitting phase they estimate the pose, expression and illumination condition and construct the respective AAM model for fast fitting. Although their fitting process is similar to conventional AAM but their model enables them to converge more rapidly and efficiently.

Active Wavelet Networks Hu et al. [95] proposed a method for face alignment called active wavelet networks (AWN), which replaces the AAM texture model by a wavelet network representation. They proposed that since PCA-based texture model of AAM causes the reconstruction error to be globally spread over the image and their model considers spatially localized wavelets for modelling texture, therefore the alignment of the face by AWN is more robust to occlusions and variations in illumination. This method is not suitable for classical AAM but can be adapted for the method of Shape-AAM. In each iteration they need to calculate texture parameters by orthogonally projecting the normalized face image into the learned wavelet subspace of the training phase. Texture reconstructed from these textures parameters is used to calculate residual image. Thus the use of Gabor wavelet filters makes this algorithm very complex.

The main drawback of this technique is the computations to select the model for the current query image, therefore takes more time despite of the fact that it converges more rapidly and efficiently.

3.2.2.2 Compositional Approach and Direct Search Methods for AAM Fitting

The main difference between gradient based search and direct search is their capability of error function exploration. Gradient based methods always tends towards the better solutions while exploiting the current solution and converging towards the gradient of the function. Whereas direct search methods, unlike gradient based methods, also explore other solutions of the function. Genetic algorithm and simplex are some of the known direct search methods.

AAM Fitting by Compositional Approach Matthews and Baker [96] and then Mercier et al. [97] used compositional approach for active appearance model. Their fitting algorithm is based on gradient descent algorithm called Inverse Compositional Lucas-Kanade algorithm (IC-LK) proposed by Baker and Matthews [98]. Their model generation procedure is similar to AAM but without performing the concatenation of the two parameter i.e. 3rd PCA. Equation for the AAM after performing the PCA is obtained as:

$$s_i = s_{mean} + \sum_{k=1}^{NbS} \Phi_{s_k} * b_{s_k} \tag{3.19}$$

$$g_i = g_{mean} + \sum_{k=1}^{NbG} \Phi_{g_k} * b_{g_k} \tag{3.20}$$

In the segmentation phase, the texture of the query image I inside the shape of the current models is extracted and warped. The residual image is calculated as $\delta g = g_i - g_m$, where g_m is the texture of the current Model. Followed by the calculation of the steepest descent images $SD = \nabla g_m \frac{\delta W}{\delta b_s}$ with respect to the variation of each shape parameter b_s . Then the Hessian Matrix H is calculated as: $H = \sum_{pixel} SD^T SD$.

In segmentation phase the parameters b_s and b_g are updated by the equations $[\Delta b_s, \Delta b_g]^T = -H^{-1} \sum SD^T \delta g$ iteratively until the convergence is achieved.

The results obtained by this approach are equivalent to the classical method of AAM by linear regression, however, it introduces the complex calculations of the gradient descent and the Hessian. Cootes and Kittipanya-ngam [91] has shown that this approach is more time consuming than the classical method.

Simplex Nelder and Mead have proposed the Nelder Mead Simplex algorithm [99], which is an iterative direct search method and it is used to optimize both the appearance and pose parameters at the same time. In [100, 101], Y. Aidarous et al. have used Nelder Mead Simplex for the optimization in 2D AAM. Similarly Paterson and Fitzgibbon [102] also used Simplex as an optimization for model-based head tracking technique. Cristinacce and Cootes [103] also used simplex for the optimization of their Template Selection Tracker (TST) to localize the facial features.

Genetic Algorithm Genetic Algorithm is a well-known direct search method proposed by Goldberg [104]. In segmentation phase of AAM appearance C and pose parameters P are considered as genes. All the genes of C and P are concatenated to

form a chromosome. Population of these chromosomes is randomly created. Tournament selection is applied to select parent chromosomes from the population to undergo reproduction. Two points crossover and Gaussian mutation is implemented to reproduce next generation of the chromosomes. Thus, a new generation of the same size of population is created using genes of the fittest of the old chromosome. Elitism can also be implemented to preserve the best possible solution at all time. After calculating a number of generations the algorithm can be stopped according to a specified stopping criteria. McIntosh and Hamarneh [105] and Ghosh and Mitchell [106] perform segmentation of medical images using genetic algorithm.

These optimization methods are slower than the classical AAM, but they reduce the required memory space because they do not need to save the regression matrix R_C and R_P in memory. They also improve the efficiency of AAM since exploration of the search space is not restricted as in the case of gradient descent and linear regression methods. Since they do not need training or pre-computation phase, therefore one of the major advantages achieved is the generality. Generality is the capability of the model to analyze other than those faces from which it has been created.

3.2.2.3 Model Extension

This subsection presents the first approach to estimate the facial pose and features by the extensions of the AAM model and its appearance parameters. One way is to use multiple 2D AAM model each corresponding to different face orientation. Other way is to use a single 3D AAM model which can be rotated with the help of the pose parameters to compensate each facial orientation. Some researchers have also extended appearance parameters for the pose variability.

Multiple 2DAAM Model

Cootes et al. [107] showed that by using five models it is sufficient to deal with faces where the head pose vary by 180 degrees (from left profile to right profile). To adapt to the pose of a face, five models are built from different learning databases. Thus in segmentation, for each query image, five models are then used and only the model providing the lowest error convergence is considered.

Similarly Shan et al. [108] performed pose prediction by using three AAM models, one dedicated to the frontal view and two for the profile views. Sung and Kim [109]

detected pose-robust facial expression by using three 2D+3D AAM models, one dedicated to the frontal view and two for the side views. Li et al. [110] also used three DAMs (Direct Appearance Models) for face alignment.

3D AAM

Von Duhn et al. [111] used three cameras to acquire three (frontal, profile and angle) views of a face. Landmarks localized by 2D AAM on each image are correlated to build a 3D model. They used this model for the recognition of an oriented face. Paterson and Fitzgibbon [102], Ishiyama et al. [112], Malassiotis and Strintzis [113] also developed 3D models of faces, made from faces acquired by laser scanners providing cylindrical data of the face. In order to create the 3D deformable model, a morphing is performed between all these examples of 3D faces. The deformations of the model are controlled by parameters such as of AAM. This method requires learning a 3D faces of good resolution.

2D+3D AAM

Xiao et al. [114], Hu et al. [115], Koterba et al. [116], Ramnath et al. [117] used 2D+3D AAM along with a fitting algorithm, called inverse compositional image alignment algorithm, which is an extension of a gradient descent method. Their AAM model is obtained by the non-rigid structure-from-motion algorithm of Xiao et al. [118]. This algorithm requires 2D shapes by tracking the face in a video sequence by 2D AAM, followed by the computation of 3D shape modes from this 2D AAM shape. Ultimately they combined these 3D shape modes with 2D AAM to build 2D+3D AAM model. Their fitting algorithm is similar to the one of Matthews and Baker [96] with additional 3D shape mode to optimize. Sung and Kim [109] also used 2D+3D AAM of Xiao et al. [114] to detect pose-robust facial expression by using three 2D+3D AAM models, one dedicated to the frontal view and two for the side views. Those techniques of building the 3D model require structure-from-motion algorithms, by applying an efficient 2D AAM on the sequence of oriented facial images. This procedure does not provide enough accurate 3D model compared to manually labelling the landmarks on the frontal and profile views of a facial image. Additionally, a number of shape parameters for a face search optimization are increased, compared to a simple 3D AAM with increased pose parameters.

3.3 Our Contributions

The classical Active Appearance Model explains novel images by linear combination of statistic models which are built by applying Principal Component Analysis on training data. However, PCA is not designed to extract non-linear features from the shape and texture of the non-frontal or non-uniformly illuminated faces. In general, both illumination and pose variations remain difficult to handle in face recognition.

In this section, the non-linear component analysis methods are considered to be more appropriate for handling the multiple variations which are caused by the changes of the light source. In this respect, two non-linear embedding models are employed instead of classical PCA to search more efficient parameters for generating new images in complex illumination and pose conditions. The following subsections aim at presenting the proposed Probabilistic-PCA based and Kernel Similarity PCA based Active Appearance Model method. The experiment results compared with classical AAM are provided to validate the robustness and accuracy of the proposed works.

3.3.1 Motivation

3.3.1.1 Properties and limitations of PCA

Principal component analysis (PCA) [5] is a well-established technique for dimension reduction, and a chapter on the subject may be found in practically every text on multivariate analysis. Examples of numerous applications of PCA include data compression, image processing, visualization, exploratory data analysis, pattern recognition and time series prediction. The most common derivation of PCA is in terms of a standardized linear projection which maximizes the variance in the projected space [119].

Consider a set of observed D-dimensional data vectors $t_n, n \in 1, 2, \cdots, N$, the d principal axes $w_j, j \in 1, 2, \cdots, d$, are those orthogonal axes onto which the retained variance under projection is maximal. It can be shown that the vectors w_j are given by the d dominant eigenvectors (i.e. those with the largest associated eigenvalues λ_j) of the sample covariance matrix $S = E[(t-\mu)(t-\mu)^T]$ such that $S_{w_j} = \lambda_j w_j$. The d principal components of the observed vector t_n are given by the vector $x_n = W^T(t_n - \mu)$, where $W^T = (w_1, w_2, \cdots, w_d)^T$. The variables x_j are then decorrelated such that the covariance matrix $E[x \cdot x^T]$ is diagonal with elements λ_j.

A complementary property of PCA, and that most closely related to the original discussions of Pearson (1901) is that, of all orthogonal linear projections:

$$x_n = W^T(t_n - \mu) \tag{3.21}$$

the principal component projection minimizes the squared reconstruction error $\sum_n \left\|t_n - \hat{t_n}\right\|^2$, where the optimal linear reconstruction of t_n is given by $\hat{t_n} = Wx_n + \mu$.

Principal component analysis (PCA) has widespread applications because it reveals simple underlying structures in complex data sets using analytical solutions from linear algebra [120]. A primary benefit of PCA arises from quantifying the importance of each dimension for describing the variability of a data set. In particular, the measurement of the variance along each principal component provides a means for comparing the relative importance of each dimension. An implicit hope behind employing this method is that the variance along a small number of principal components (i.e. less than the number of measurement types) provides a reasonable characterization of the complete data set. PCA is completely nonparametric: any data set can be plugged in and an answer comes out, requiring no parameters to tweak and no regard for how the data was recorded. A deeper appreciation of the limits of PCA requires some consideration about the underlying assumptions and in tandem, a more rigorous description of the source of data. Generally speaking, the primary motivation behind this method is to decorrelate the data set, or said in other terms, the goal is to remove second-order dependencies in the data. However, as presented in the following section 3.3.1.2, there exists higher-order dependencies in the complex illumination data, therefore, removing second-order dependencies is insufficient at revealing all structures in this problem.

3.3.1.2 Statistic Analysis on illuminated faces

To fully understand the difference between a set of normally illuminated faces and a set of irregular illuminated faces, a statistic analysis is definitely meaningful. Consider a set of face images, as shown in Fig. 3.3, where the pose of the faces are fixed as frontal ones, and the illumination conditions vary dynamically. As defined in section 3.2.1,eq. 3.4,fig. 3.1, the contour of the faces are described in shape vectors s_i, by the coordinates of the points, and the texture of the faces are represented by vectors g_i (equation 3.5), which contains all the pixel intensities in Delaunay Triangulation. To

analyze the correlation between shape and texture of all faces, the Euclidean distances $d(a, b)$ between elements contained in each vector are computed both for shape and texture matrix of the corresponding data set as:

$$d(s_i, s_j) = \sqrt{\sum_{i=1}^{D_s} (s_i - s_j)^2} \tag{3.22}$$

$$d(g_i, g_j) = \sqrt{\sum_{i=1}^{D_g} (g_i - g_j)^2} \tag{3.23}$$

Then, the histograms of the Euclidean distances are built as demonstrated in Fig. 3.4 and 3.5, for shape vectors and texture vectors from the illuminated face data. One can notice from the histograms that there exists a strong correlation within the illuminated face image data. And we can assume that the Euclidian distances between each observed variable follows a Gaussian distribution.

Based on the statistic analysis, it is obvious that the basic assumption of PCA that the independence between latent variables, and the remove of second-order dependencies, is not suitable for this illuminated frontal face data set. The non-linear component analysis methods are considered to be more appropriate for handling the multiple variations which are caused by the changes of the light source. In the next section, we demonstrate the application of Probabilistic PCA and Kernel Similarity PCA embedding for building the AAMs model to overpass this complicated face illumination problem.

Figure 3.3: **Frontal face Image examples from the CMU PIE database** - in 20 illumination conditions

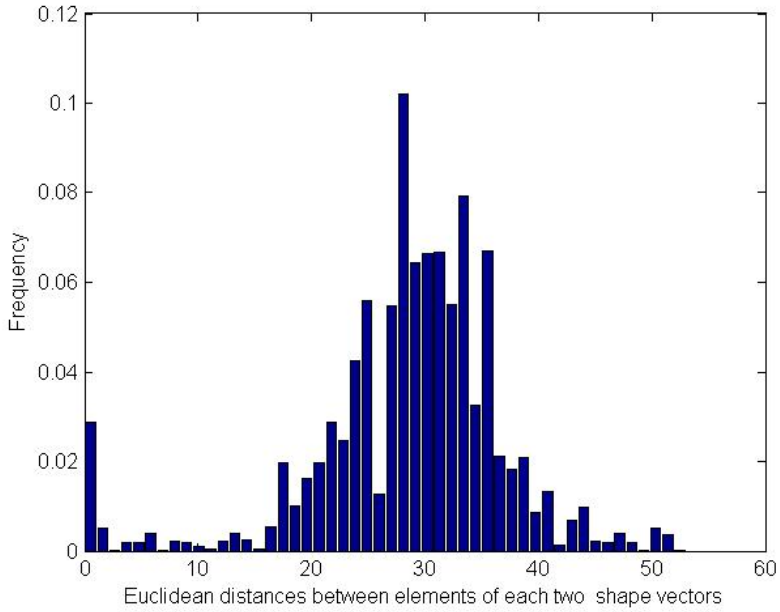

Figure 3.4: **Histogram** - Euclidean distances between each shape vector pairs

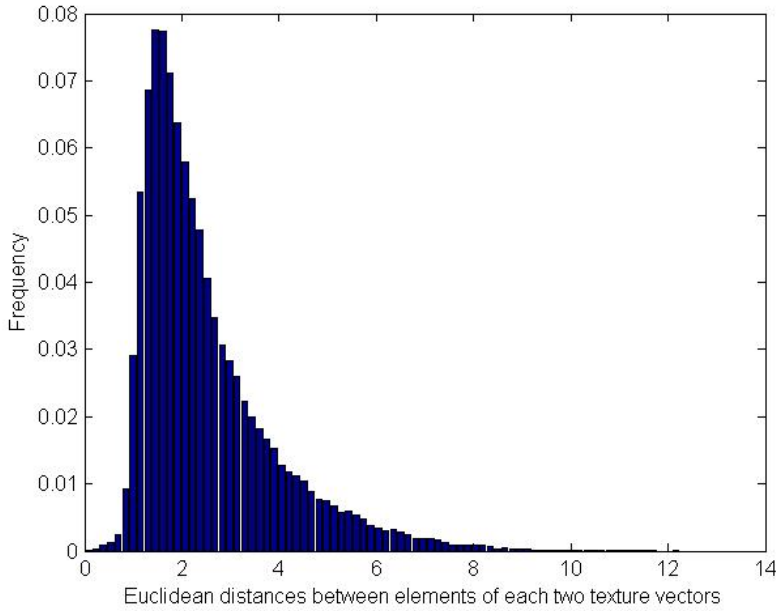

Figure 3.5: **Histogram** - Euclidean distances between each texture vector pairs

3.3.2 Probabilistic PCA (PPCA)

Principal component analysis (PCA) is a ubiquitous technique for data analysis and processing, but one which is not based upon a probabilistic model. In this section we demonstrate how the principal axes of a set of observed data vectors may be determined through maximum-likelihood estimation of parameters in a latent variable model closely related to factor analysis, as introduced by Tipping and Bishop in [121]. We consider the properties of the associated likelihood function, giving an EM algorithm for estimating the principal subspace iteratively, and discuss the advantages of AAMs which is built based on this probabilistic PCA in section 3.3.4.

3.3.2.1 The probabilistic Model

A latent variable model seeks to relate a d-dimensional observation vector t to a corresponding N-dimensional vector of latent variables x. Perhaps the most common such model is factor analysis where the relationship is linear:

$$t = Wx + \mu + \epsilon \tag{3.24}$$

The $N \times d$ matrix W relates the two sets of variables, while the parameter vector μ permits the model to have non-zero mean. The motivation is that, with $N << d$, the latent variables will offer a more parsimonious explanation of the dependencies between the observations. Conventionally, the latent variables are defined to be independent and Gaussian with unit variance. By additionally specifying error, noise, or model to be likewise Gaussian $\epsilon \sim N(0, \Psi)$, equation 3.24 induces a corresponding Gaussian distribution for the observations $t \sim N(\mu, WW^T + \Psi)$. The model parameters may thus be determined by maximum-likelihood, although because there is no closed-form analytic solution for W and Ψ, their values must be obtained via an iterative procedure.

The motivation, and indeed the key assumption for the factor analysis model is that, by constraining the error covariance Ψ to be a diagonal matrix whose elements ψ_i are usually estimated from the data, the observed variables t_i are conditionally independent given the values of the latent variables x. These latent variables are thus intended to explain the correlations between observation variables while ϵ_i represents the variable unique to a particular t_i. This is where factor analysis fundamentally differs from standard PCA, which effectively treats covariance and variance identically.

The use of the isotropic Gaussian noise model $N(0, \delta^2 I)$ for ϵ in conjunction with equation 3.24 implies that the x-conditional probabilistic distribution over t-space is given by:

$$P(t|x) \sim N(Wx + \mu, \delta^2 I) \tag{3.25}$$

With the marginal distribution over the latent variables also Gaussian and conventionally defined by $x \sim N(0, I)$, the marginal distribution for the observed data t is readily obtained by integrating out the latent variables and is likewise Gaussian:

$$P(t) \sim N(\mu, C) \tag{3.26}$$

where the observation covariance model is specified by $C = WW^T + \delta^2 I$. The corresponding log-likelihood is then

$$L = -\frac{N}{2}\left\{dln(2\pi) + ln\,|C| + tr(C^{-1}S)\right\} \tag{3.27}$$

where an exponential function of the distance is adopted as the local likelihood as:

$$S = \sum_{n=1}^{d}\sum_{m=1}^{d}(t_n - \mu_{t_n})(t_m - \mu_{t_m})P(t_n)P(t_m) \tag{3.28}$$

$$P(t_n) = \frac{1}{2\pi\delta}e^{-\frac{|t_n - \bar{t}_n|}{2\delta^2}} \tag{3.29}$$

The maximum-likelihood estimator for $\bar{t}_n$ is given by the mean of the data, in which case S is the sample covariance matrix of the observations t_n. Estimates for W and δ^2 may be obtained by iterative maximization of L, for example using the EM algorithm, which is based on the algorithm for standard factor analysis [122]. However, in contrast to factor analysis, for W and δ^2 may be obtained explicitly. Later, we will make use of the conditional distribution of the latent variables x given the observed t, which may be calculated using Bayes' rule and is again Gaussian:

$$P(x|t) \sim N(M^{-1}W^T, \delta^2 M^{-1}) \tag{3.30}$$

where we have defined $M = W^T W + \delta^2 I$. Note that M is of size $d \times d$ while C is $N \times N$.

3.3.2.2 Revisited Properties of the Maximum-Likelihood Estimators

In [121] it is shown that, with C given by $WW^T + \sigma^2 I$, the likelihood eq. 3.27 is maximized when:

$$W_{ML} = U_d(\Lambda_d - \sigma^2 I)^{1/2} R \tag{3.31}$$

where the d column vectors in the $N \times d$ matrix U_d are the principal eigenvectors of S, with corresponding eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_d$ in the $d \times d$ diagonal matrix Λ_d , and R is an arbitrary $d \times d$ orthogonal rotation matrix. Other combinations of eigenvectors (i.e. non-principal ones) correspond to saddle-points of the likelihood function. Thus, from equation 3.31, the latent variable model defined by equation 3.24 effects a mapping from the latent space into the principal subspace of the observed data. It may also be shown that for $W = W_{ML}$, the maximum-likelihood estimator for σ^2 is given by

$$\sigma^2_{ML} = \frac{1}{N-d} \sum_{j=d+1}^{N} \lambda_j \tag{3.32}$$

which has a clear interpretation as the variance 'lost' in the projection, averaged over the lost dimensions. In practice, to find the most likely model given S, we would first estimate σ^2_{ML} from 3.32, and then W_{ML} from 3.31, where for simplicity we would effectively ignore R (i.e. choose $R = I$). Alternatively, we might employ the EM algorithm, where R at convergence can be considered arbitrary.

3.3.2.3 Dimensional Reduction

The general motivation for PCA is to transform the data into some reduce-dimensionality representation, and with some minor algebraic manipulation of W_{ML}, we may indeed obtain the standard projection onto the principal axes if desired. However, it is more natural from a probabilistic perspective to consider the dimensionality-reduction process in terms of the distribution of the latent variables, conditioned on the observation. From 3.30, this distribution may be conveniently summarized by its mean:

$$\langle x_n | t_n \rangle = M^{-1} W^T_{ML} (t_n - \bar{t_n}) \tag{3.33}$$

(Note, also from 3.30, that the corresponding conditional covariance is given by $\sigma^2_{ML}M^{-1}$ and is thus independent of n.) It can be seen that when $\sigma^2 \to 0, M^{-1} \to (W^T_{ML}W_{ML})^{-1}$ and 3.33 then represents an orthogonal projection into latent space and so standard PCA is recovered. However, the density model then becomes singular, and thus undefined. In practice, with $\sigma^2 > 0$ as determined by 3.32, the latent projection becomes skewed towards the origin as a result of the Gaussian marginal distribution for x. Because of this, the reconstruction $W_{ML}\langle x_n|t_n\rangle + \bar{t_n}$ is not an orthogonal projection of t_n, and is therefore not optimal (in the squared reconstruction-error sense). Nevertheless, optimal reconstruction of the observed data from the conditional latent mean may still be obtained, in the case of $\sigma^2 > 0$, and is given by $W_{ML}(W^T_{ML}W_{ML})^{-1}M\langle x_n|t_n\rangle + \bar{t_n}$.

3.3.3 Kernel Similarity Principal Component Analysis

In most existing dimensionality reduction algorithms, the main objective is to preserve relational structure among objects of the input space in a low dimensional embedding space. This is achieved by minimizing the inconsistency between two similarity/dissimilarity measures, one for the input data and the other for the embedded data, via a separate matching objective function. Based on this idea, we propose a new dimensionality reduction method called Kernel Similarity PCA (KS-PCA).

KS-PCA addresses this problem of non-standard illuminated face data that is difficult for linear methods in practice due to the correlations between shape and texture vectors (as described in section 3.3.1.2). In the implementation, by optimizing a non-linear objective function using the gradient descent algorithm, a local minimum can be reached. The results obtained include both the optimal similarity preserving embedding and the appropriate values for the hyper-parameters of the kernel. Experimental evaluation on KS-PCA based AAMs confirmed the effectiveness of this embedding method. The results for face recognition in complex illumination conditions are shown in section 3.3.4.

3.3.3.1 PCA trick in Feature Space

Principal Component Analysis (PCA) [5] is an orthogonal basis transformation. The new basis is found by diagonalizing the centered covariance matrix of a data set

$\{x_i \in R^D, i = 1, 2, \cdots, N\}$, defined by $Cov = \langle (x_i - \bar{x}) \cdot (x_j - \bar{x})^T \rangle$. The coordinates in the Eigenvector basis are called principal components.

Clearly, one cannot assert that linear PCA will detect complex structure in a given data set. By the use of suitable nonlinear features, one can extract more information. The Kernel PCA first maps the data into some feature space Ψ via a (usually nonlinear) function Φ and then performs linear PCA on the mapped data. As the feature space Ψ might be very high dimensional (e.g. when mapping into the space of all possible $D - th$ order monomials of input space), kernel PCA employs Mercer kernels instead of carrying out the mapping Φ explicitly.

Clearly, all algorithms that can be formulated in terms of dot products, e.g. Support Vector machines [123], can be carried out in some feature space Ψ without mapping the data explicitly. All these algorithms construct their solutions as expansions in the potentially infinite-dimensional feature space

Consider a $D \times N$ observation matrix A, where each column is an observation and each row is the dimension of the observation. In our work, each column is an image and each row is the image pixels. One observation is denoted as x_k, $k = 1, 2, \ldots\ldots, N$, $x_k \in R^D$, and $\sum_{k=1}^{N} x_k = 0$, which means that the data is centered. Normally, PCA diagonalizes the covariance matrix as shown in Eqn.3.34.

$$Cov = \frac{1}{(D-1)} \sum_{i,j=1}^{D} x_i x_j^T \tag{3.34}$$

Via Singular Value Decomposition (SVD), the covariance matrix $Cov \in R^{D \times D}$ as show in eq. 3.34 can be decomposed as:

$$Cov = U \Lambda U^* \tag{3.35}$$

where U is a $D \times D$ unitary matrix, Λ is an $D \times D$ square diagonal matrix with nonnegative real numbers on the diagonal, and U^* is the conjugate transpose of U, or simply the transpose of U if U is real. The diagonal entries $\lambda_{d,d}$ of Λ are known as the eigenvalues of Cov. The columns u of matrix U are called the eigenvectors.

In some special case we have much more dimensions than faces, that $D >> N$, so finding the eigenvectors of the large $D \times D$ matrix is computationally difficult. So we apply a PCA trick: instead of 3.34, Equation 3.36 is more computationally tractable.

$$\tilde{Cov} = \frac{1}{(N-1)} \sum_{i,j=1}^{N} y_i y_j^T \tag{3.36}$$

where y_i is the vector of each element of the observation x_j, D is the dimension of observation x_j.

To diagonalize it, one has to solve the following eigenvalue equation:

$$\tilde{\lambda} \cdot v = \tilde{Cov} \cdot v. \tag{3.37}$$

where $\tilde{\lambda} = \lambda$ represent to the eigenvalues of the matrix $\tilde{Cov}$; the eigenvectors $v = A^T u = \sum_{i=1}^{N} y_i u_i$.

The previous part of this section is devoted to a straightforward translation to a non-linear scenario. We shall now describe this computation in a Hilbert space H, which is introduced via a mapping Φ.

$$\Phi : R^N \to H, x \to X. \tag{3.38}$$

In the feature space H, we assume that $\Phi(x)$ has an arbitrarily large, possibly infinite dimensionality. Again, in feature space, the data should be centered. Applying the PCA trick in feature space H,

$$\bar{Cov} = \frac{1}{(N-1)} \sum_{i=1}^{N} \Phi(y_i)\Phi(y_i)^T \tag{3.39}$$

Now one has to extract eigenvalues satisfying:

$$\bar{\lambda} \cdot V = \bar{Cov} \cdot V \tag{3.40}$$

The solutions V lies in the span of $\Phi(y_1), \Phi(y_2), \cdots, \Phi(y_N)$. This has two useful consequences: first, we can consider the equivalent equation

$$\bar{\lambda}(\Phi(y_k)^T V) = (\Phi(y_k)^T \cdot \bar{Cov} \cdot V) \tag{3.41}$$

for all $k = 1, 2, \cdots, N$, there exist coefficients α_i $(i = 1, \cdots, N)$ such that

$$V = \sum_{i=1}^{N} \alpha_i \Phi(y_i) \tag{3.42}$$

Combining eq. 3.41 and eq. 3.42, we get

$$\bar{\lambda}\sum_{i=1}^{N}\alpha_i(\Phi^T(y_k)\cdot\Phi(y_i)) = \frac{1}{M}\sum_{i=1}^{N}\alpha_i(\Phi^T(y_k)\cdot\sum_{j=1}^{N}\Phi(y_j))(\Phi^T(y_j)\cdot\Phi(y_i)) \tag{3.43}$$

where $M = N - 1$.

Defining a $N \times N$ matrix K by

$$K_{i,j} = (\Phi^T(y_i)\cdot\Phi(y_j)) \tag{3.44}$$

which lead eq. 3.42 to:

$$M\bar{\lambda}K\alpha = K^2\alpha \tag{3.45}$$

where α denotes the column vector with entries $\alpha_1, \cdots, \alpha_N$. As K is symmetric,

$$M\bar{\lambda}\alpha = K\alpha \tag{3.46}$$

Note that K is semi-positive and sym definite, which can be seen by noticing that it equals to

$$K_{i,j} = k(y_i, y_j) = e^{-\frac{\left\|y_i - y_j\right\|^2}{2\delta^2}} \tag{3.47}$$

Then, for the extraction of eigenvalues in feature space, we therefore only need to diagonalize the kernel similarity matrix $K_{i,j}$. Let $\bar{\lambda_1} \geq \bar{\lambda_2} \geq \cdots \geq \bar{\lambda_N}$ denote the eigenvalues, and $\alpha^1, \alpha^2, \cdots, \alpha^N$ the corresponding complete set of eigenvectors.

3.3.3.2 Parameter Estimation

As described previously, the Kernel Similarity method component analysis deals with nonlinear transformation via nonlinear kernel functions. There are several commonly used kernel functions, which are proven useful include: Gaussian kernel $k(x, y) = e^{-\frac{\|x-y\|^2}{2\delta^2}}$, Polynomial kernel $k(x, y) = (x \cdot y)^n$, Sigmoid kernel $k(x, y) = tan(\alpha x^T y + c)$ etc.

As assumed in section 3.3.1.2, the variables of the illuminated frontal face problem follow the Gaussian distribution. Therefore, the Gaussian kernel is the most appropriate kernel to choose. In the Gaussian kernel, there is a parameter σ that must be

predetermined, knowing that it has a significant impact on image representation in feature space. As the kernel function is defined with $k(x,y) = e^{-\frac{d(x,y)^2}{2\delta^2}}$, in which $d(x,y)$ represent the Euclidean distances between elements contained in each vector; $k(x,y)$ can be considered as a zero mean Gaussian distribution of $d(x,y)^2$. So since $d(x,y)^2$ follows Gaussian distribution, then σ represented the variance of $d(x,y)^2$. With respect to this assumption, as illustrated in Fig.3.4 and 3.5, for shape vectors and texture vectors from the illumination database, the parameters σ is estimated as the variance of $d(x,y)^2$.

3.3.4 Experimental Results

This section describes the experiments performed to validate the effectiveness of the two proposed embedding models based AAMs. Performance of the proposed face segmentation techniques were compared against the classical AAMs approach presented in 3.2. The experimental databases are the IMM Face Database and the CMU PIE database. For training the appearance models, 58 landmarks were placed on each face image: 8 points for the mouth, 11 points for the nose, 16 points for both eyes, 10 points for both eyebrows, and 13 points for the chin. A total of seven point paths were used; three closed and four open. All annotations were formatted in ASF. The warped images have approximately 7325 pixels inside the facial mask. Parameter C (defined in section 3.2.1, eq. 3.10) is constrained by 3λ, where λ are the eigenvalues obtained by applying PCA and retaining 95% of the variation in equation 3.10.

3.3.4.1 Evaluation criterion

For illustrating the experiment performance more clearly, we define the evaluation criterion in this section.

In the evaluation, the manually annotated landmarks are considered as the ground truth shape information. For each image the landmarks relabeled by the models are compared with the ground truth landmarks. A distance measure, $D(s_{gt}, s)$, gives a interpretation of the fit between two shapes, the ground truth, s_{gt} and the actual shape s. Point-to-point error E_{pt-pt} (shown in equation 3.48,) is defined as the average Euclidean distance between each corresponding landmarks from a single face.

$$E_{pt-pt} = \frac{1}{n} \sum \sqrt{(x_i - x_{gt,i})^2 + (y_i - y_{gt,i})^2} \tag{3.48}$$

where (x_i, y_i) are the coordinates of the relabeled landmarks.

To interpret a novel image, an optimization is performed in which the method minimizes the error between the pixels contained in a new image and the pixels synthesized by the appearance model. The pixel-to-pixel error $E_{pix-pix}$ can be defined as in Equation 3.49.

$$E_{pix-pix} = |\delta I|^2 = |I_i - I_m|^2 \tag{3.49}$$

where I_i is the vector of grey-level values in the image and I_m is the vector of grey-level values for the current model parameters.

To demonstrate the superiority of the proposed methods, Eqn.3.50 and 3.51 are used to compute the gain in terms of shape and texture precision of face fitting results, compared with the fitting result of classical AAMs.

$$E_{pt-pt}gain\% = \frac{E_{pt-pt}(AAMs) - E_{pt-pt}(kernel)}{E_{pt-pt}(AAMs)}\% \tag{3.50}$$

$$E_{pix-pix}gain\% = \frac{E_{pix-pix}(AAMs) - E_{pix-pix}(kernel)}{E_{pix-pix}(AAMs)}\% \tag{3.51}$$

3.3.4.2 Experiments on the IMM Face Database

We first evaluated the proposed method on the IMM Face Database [124], a commonly used database for testing deformable models. IMM Face Database comprises 240 still images of 40 different human faces, all without glasses. The gender distribution is 7 females and 33 males. The following facial structures were manually annotated using 58 landmarks. Refer to Figure 3.6 for an example of the face annotation.

The subset for training is built by the images of the first 15 individuals (number 01 to 15). Each person has 6 face images with different poses and expressions. Images of the rest 35 individuals are used for test procedure.

The face fitting results synthesized by the two proposed embedding model based AAM and classical AAM is shown in figure 3.7, for demonstrating the face segmentation performance. In the first column, faces are synthesized by the KS-PCA based

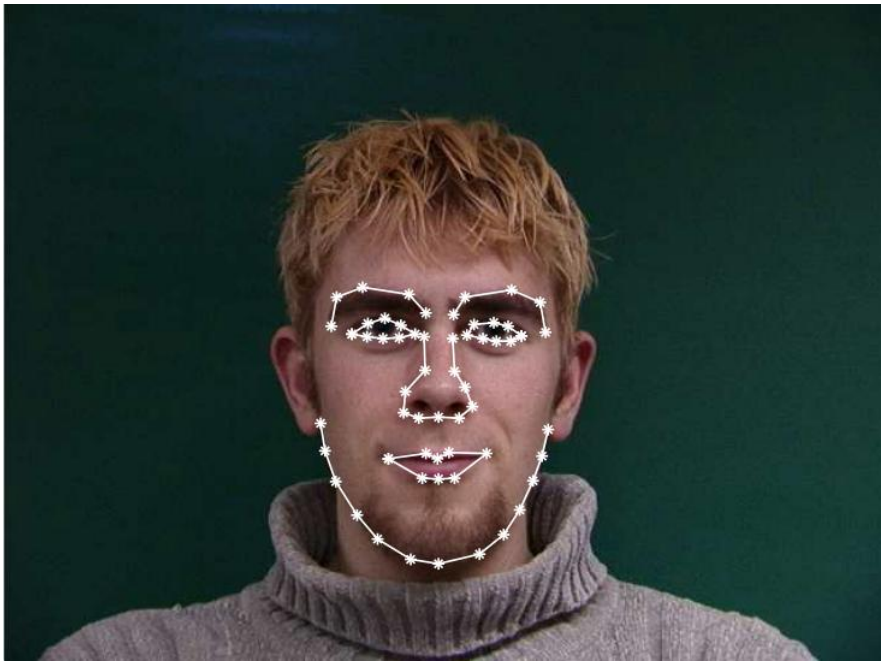

Figure 3.6: **Annotated face image** - 58 landmarks

AAMs, images in the second column represent the face segmentation result of the Probabilistic PCA based AAMs, and compared with fitting results of classical AAMs in the third column. An increased precision has been obtained due to the extraction of non-linear features. The gain in precision on point-to-point errors is reported in table 3.1, computed by eq. 3.48, eq. 3.49 and eq.3.50 respectively.

		$E_{pt-pt} \quad gain\%$	$E_{pix-pix} \quad gain\%$
Training Database	KS-PCA	25.64%	17.07 %
	PPCA	12.38%	9.46%
Test Database	KS-PCA	15.51%	10.42 %
	PPCA	11.86%	7.73%

Table 3.1: Gain in terms of fitting precision- on IMM database (computed by equation 3.50 and 3.51).

We can notice that with the proposed method, the segmentation performance and the fitting error (both for pixels and points) of the two proposed models are both reduced than classical AAMs. For this IMM database, the improvement of KS-PCA model is more than the PPCA model.

Figure 3.7: **Comparison of the fitting results** - Fitting results for KS-PCA based AAMs in the first column from left; fitting results for PPCA based AAMs in the second column; fitting results for classical AAMs in the third column; in the last column are the reference original faces.

3.3.4.3 Experiments on CMU PIE Database

In this section, the evaluation of the proposed method on the CMU Pose, Illumination, and Expression (PIE) database of human faces [125] is performed. The dataset contains 41,368 facial images of size 640×486 for 68 people across 13 poses, under 43 different illumination conditions, and with 4 different expressions For the experiments on the variation of both illumination and pose problems, the training database is built from a subset of the CMU database as illustrated in Figure 3.8. The test set is built from the images of the persons shown in Fig. 3.9. We manually labelled 1200 images of size 640×486 pixels. To train the models, 58 landmarks were placed on each face image. Refer to Figure 3.10 for an example annotation.

Figure 3.8: **Illustration of the experiments based on CMU PIE database** - Individuals in training databases

Appearance Model and Parameters

For building the statistic appearance models, we seek a parameter (the parameter C, defined in section 3.2.1, eq. 3.10) used to control variations for both shape and texture, which is extracted by the embedding models.

Figure 3.9: **Illustration of the experiments based on CMU PIE database** - Individuals in test databases

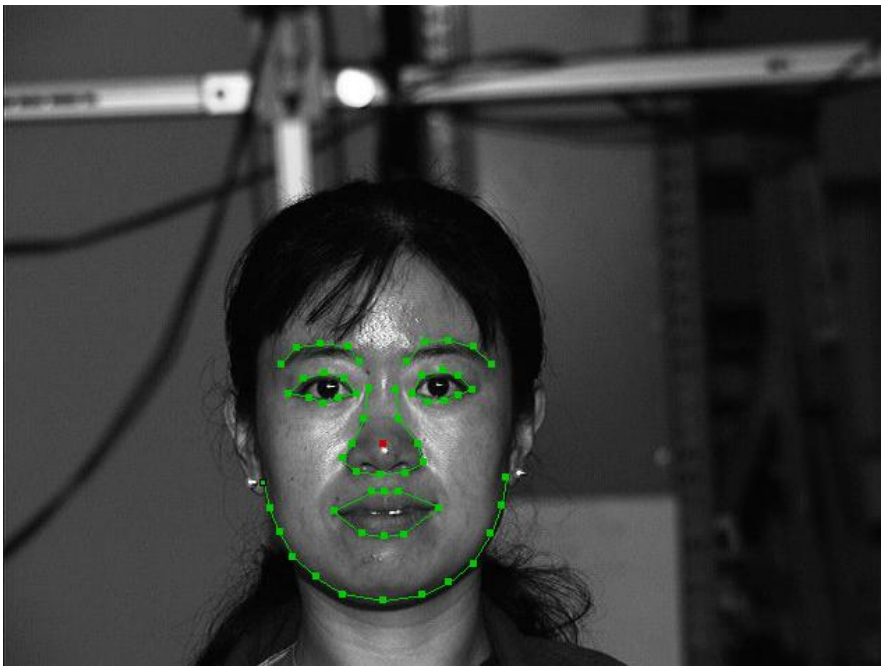

Figure 3.10: **Annotated face image** - 58 landmarks

As discussed in section 3.3.2.1 and section 3.3.3.1, in our work, the probabilistic covariance matrix $S = \sum_{n=1}^{D}\sum_{m=1}^{D}(x_i-\mu_{x_i})(t_m-\mu_{x_j})P(x_j)P(t_m)$, and the kernel similarity matrix $K_{i,j} = k(x_i, y_j) = e^{-\frac{\left\|x_i-y_j\right\|^2}{2\delta^2}}$ are employed to replace the covariance matrix for computing the standard PCA. In this section, we highlight the advantage brought by the two non-linear embedding models by analyzing the essential parameters of AAMs.

Both of the shape and texture parameters are combined by concatenation of b_s and b_g. A final embedding analysis is performed to have the appearance parameters.

$$b = [b_s b_g]^T, b = \phi_C * C \tag{3.52}$$

where ϕ_C are the eigenvectors obtained by retaining 95% of the variation, and C is the matrix, whose columns represent the appearance parameters of the faces in the training set. By disturbing the elements in one column of C, one can modify the shape and texture of the corresponding face simultaneously. The face models which synthesized by incorporating the first three appearance parameters are shown in figure 3.11. Since the appearance model C is a combination of a shape model and a texture model. Each column shows the models synthesized by varying appearance parameter from $-3\sqrt{\lambda_i}$ (left), mean shape (center) to $+3\sqrt{\lambda_i}$ (right) for the proposed PPCA model, KS-PCA model and the classical AAMs respectively. Here the parameters λ are the eigenvalues corresponding to each appearance parameter (obtained during PCA) and i is the index of the appearance parameter. One can observe that the model built by kernel similarity matrix, as shown in the second row of Figure 3.11, is able to take into account more efficiently the variations of illumination. While for the features in the first row, which belong to PPCA model, the ability to control the variation of illumination is quite limit, not as strong as the KS-PCA model. This can also be recovered in the face segmentation results which are shown in the next section.

Sensitivity to variations of Illumination

In Figure 3.12, images in the first column are synthesized by the KS-PCA based AAMs, images in the second column are synthesized by the Probabilistic-PCA based AAMs. They are compared with the fitting results of Classical AAMs in the third column. In this figure, the images in the first three rows belong to the same individual. This individual is contained by the training set. While the last three rows are the images of the same individual from the test set. These fitting results highly certify the

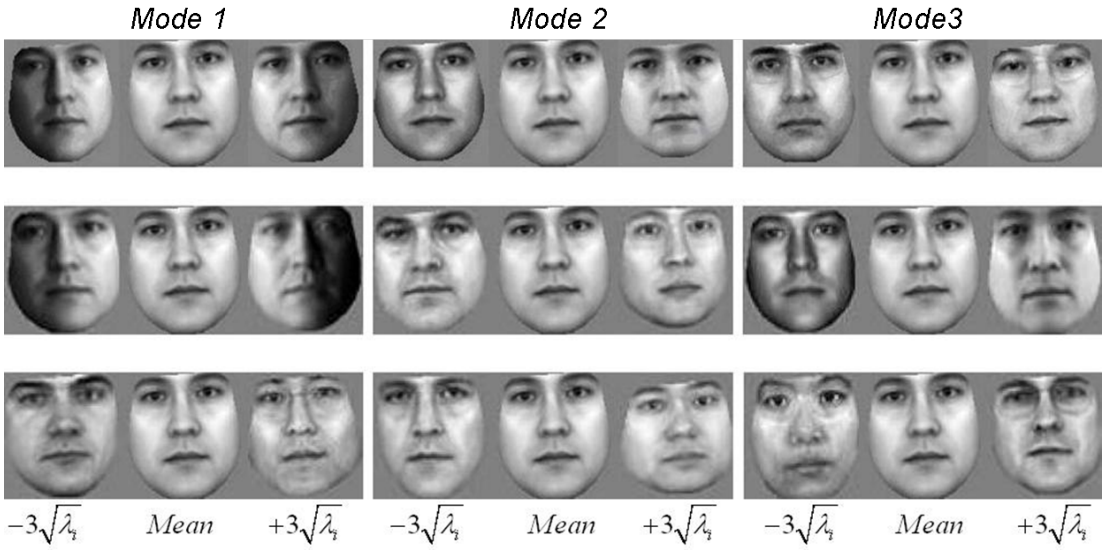

Figure 3.11: **Variations controlled by the appearance parameter C, built by KS-PCA, PPCA and PCA respectively** - The first row presents the first three modes learnt by Probabilistic-AAMs; the second row presents the first three modes learnt by KS-AAMs; the third row presents the first three modes learnt by standard AAMs)

ability of the proposed methods to synthesize unknown face appearance under variable illumination conditions. It is obvious that an increased fitting precision has been obtained due to the extraction of non-linear features. The gain in terms of accuracy on point-to-point errors and pixel-to-pixel error are reported in table 3.2.

		$E_{pt-pt} \quad gain\%$	$E_{pix-pix} \quad gain\%$
Training Database	KS-PCA	87.25%	61.36 %
	PPCA	34.58%	25.47%
Test Database	KS-PCA	76.16%	28.06 %
	PPCA	27.28%	15.06%

Table 3.2: Gain in terms of fitting precision- on CMU PIE database for poor illumimation problem (computed by equation 3.50 and 3.51).

Figures 3.13 and 3.15 show the point-to-point errors under each same illumination condition, obtained in the "classical AAMs experiment", "Probabilistic AAMs" and "Kernel Similarity AAMs experiment" on both training and test databases. The Point-to Point errors are normalized by the Euclidian distance between the eyes ($E_{pt-pt}/Deye$, where E_{pt-pt} represents point-to-point error, and $Deye$ represents the distance between the centre of the eyes of each person). This normalization is done to eliminate the effect of varying size of faces on the point-to-point error. Each curve point in Figures 3.14 and 3.16 show the mean pixel errors made by the model in the database under each same illumination condition. The number of illuminations from 1 to 20 is the 20 different illuminations contained in database. Illuminations from numbers 1 to 4 correspond to a light source from the left side of the face, illuminations from 12 to 17 correspond to a light source from the right side of the face. The other illuminations (from number 5 to 11 and number 18 to 20) correspond to different light sources in front of the face. The error curves depict the robustness of the proposed methods since it makes it possible to find non-linear facial features.

We can see that with the proposed methods, some errors are still made, but not as strong as with the classical method with a powerfully increased robustness to side illuminations. For this illuminated frontal face problem, the improvement of KS-PCA based AAMs is obviously stronger than the PPCA based AAMs. The better performance is illustrated in the parameter extraction procedure and face fitting results.

Figure 3.12: **Comparison of the face fitting results** - Fitting results for KS-PCA based AAMs are in the first column from left; fitting results for PPCA based AAMs are in the second column; fitting results for classical AAMs are in the third column; in the last column are the reference original faces.

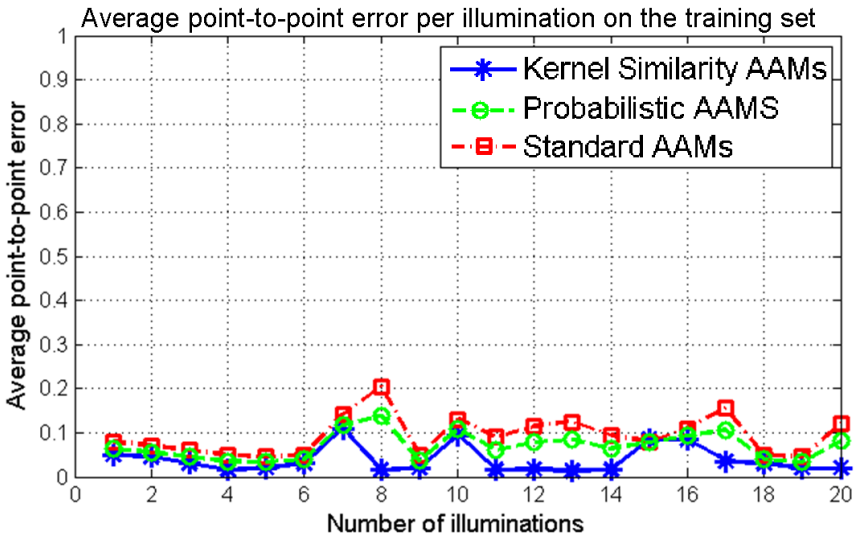

Figure 3.13: **Comparison curves between the proposed methods and classical AAMs** - Average Point-to-Point Error on the training dataset

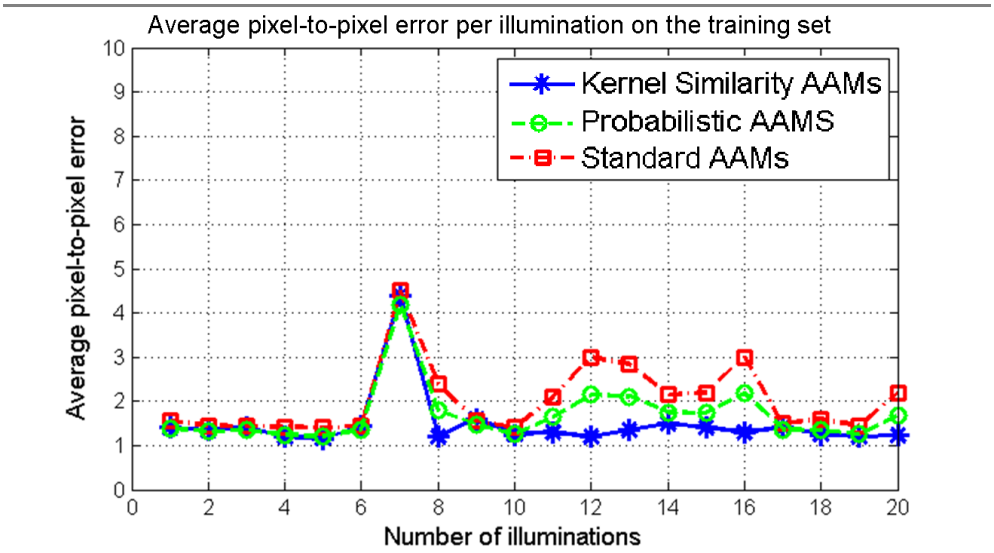

Figure 3.14: **Comparison curves between the proposed methods and classical AAMs** - Average Pixel-to-pixel Error on the training dataset

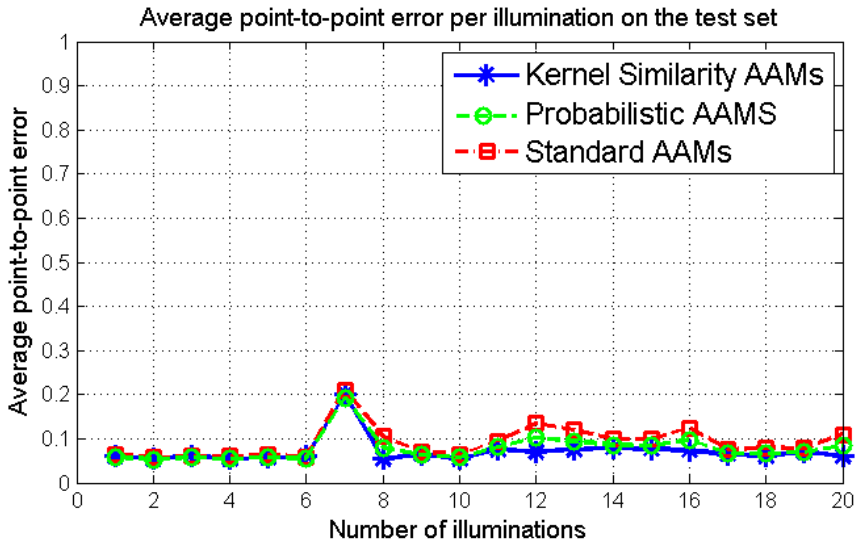

Figure 3.15: **Comparison curves between the proposed methods and classical AAMs** - Average Point-to-Point Error on the test dataset

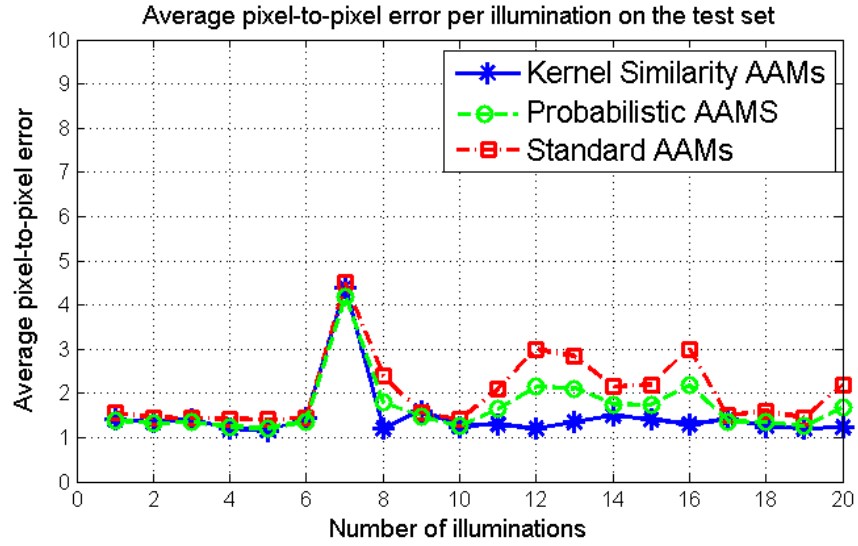

Figure 3.16: **Comparison curves between the proposed methods and classical AAMs** - Average Pixel-to-pixel Error on the test dataset

We can infer from these experiments that the KS-PCA embedding model follow more closely the non-linear variation caused by different illuminations.

Sensitivity to variations of pose

In this experiment for multiple face poses, the training set is built with the same 16 people, each person having 11 different poses captured by different cameras, the test set contains images from 10 people.

As presented in Figure 3.17, images in the first column are synthesized by the proposed KS-PCA based AAMs, while the PPCA based AAMs fitting results in the second column, compared with fitting results of classical AAMs in the third column. An increasing fitting accuracy is performed. The gain in terms of precision on point-to-point and pixel-to-pixel errors are reported in table 3.3.

		$E_{pt-pt} \quad gain\%$	$E_{pix-pix} \quad gain\%$
Training Database	KS-PCA	16.63%	14.05 %
	PPCA	10.35%	6.23%
Test Database	KS-PCA	22.44%	18.67 %
	PPCA	8.77%	11.60%

Table 3.3: Gain in terms of fitting precision - on CMU PIE database for multiple face pose problem (computed by equation 3.50 and 3.51).

Poses numbers 1, 6, 7, 10 are profile faces which are hard to synthesize, while the other poses are less complicated. The curves in Figure 3.18 to Figure 3.21 present the point-to-point errors and the pixel-to-pixel errors for each pose respectively give a consistent result. As illustrated by the curves, for the test on the training database, the proposed method is more efficient except for poses number 1, 7 and 10. On the test set, the results of poses number 1, 7 and 10 are missing, because the fitting procedures have problem to converge for both proposed methods and classical AAMs. As a consequence, the proposed non-linear methods give better fitting results in the conditions that the out-of-plane rotations of face are in a the range of $\pm 60°$. The problem of the complete profile faces is still waiting to be solved.

Figure 3.17: **Comparison of the face fitting results** - Fitting results for KS-PCA based AAMs are in the first column from left; fitting results for PPCA based AAMs are in the second column; fitting results for classical AAMs are in the third column; in the last column are the reference original faces.

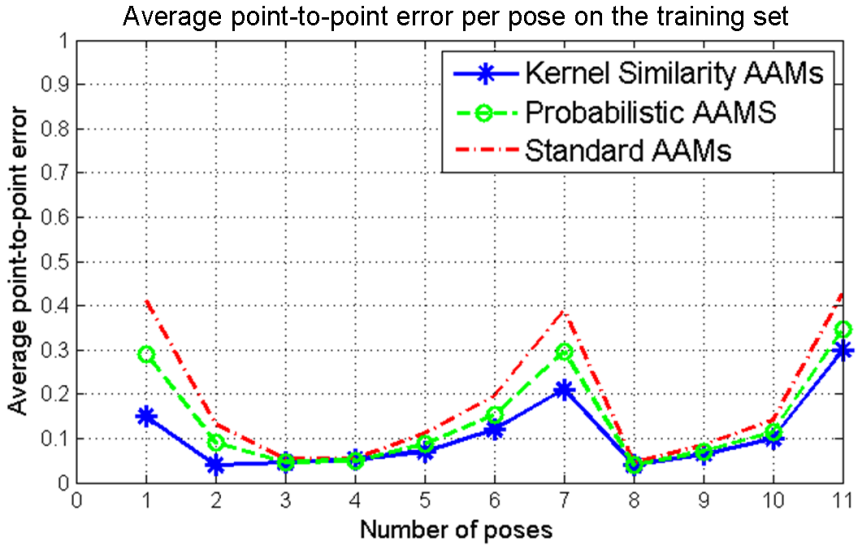

Figure 3.18: **Comparison curves between the proposed methods and classical AAMs** - Average Point-to-Point Error on the training dataset

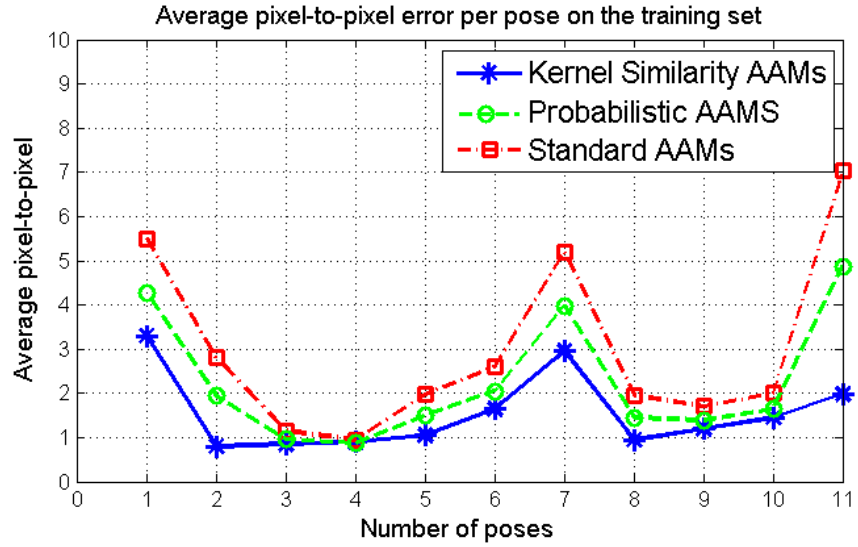

Figure 3.19: **Comparison curves between the proposed methods and classical AAMs** - Average Pixel-to-pixel Error on the training dataset

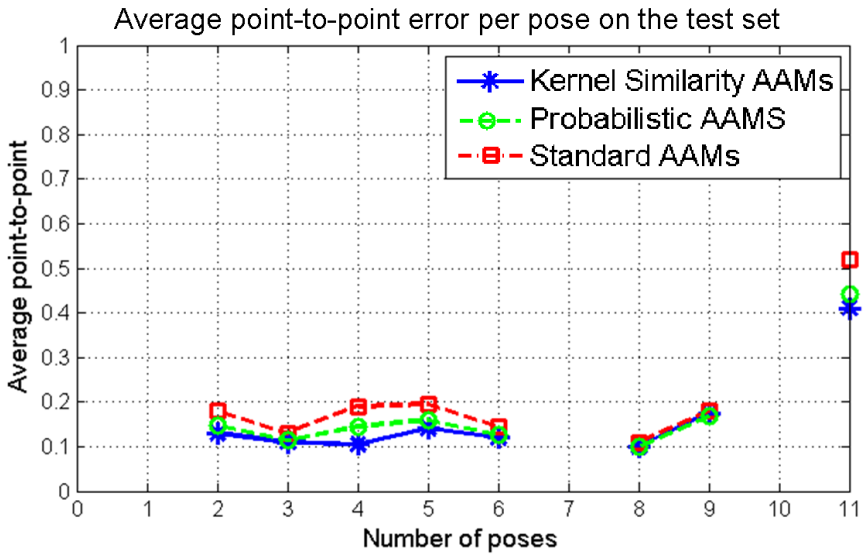

Figure 3.20: **Comparison curves between the proposed methods and classical AAMs** - Average Point-to-Point Error on the test dataset

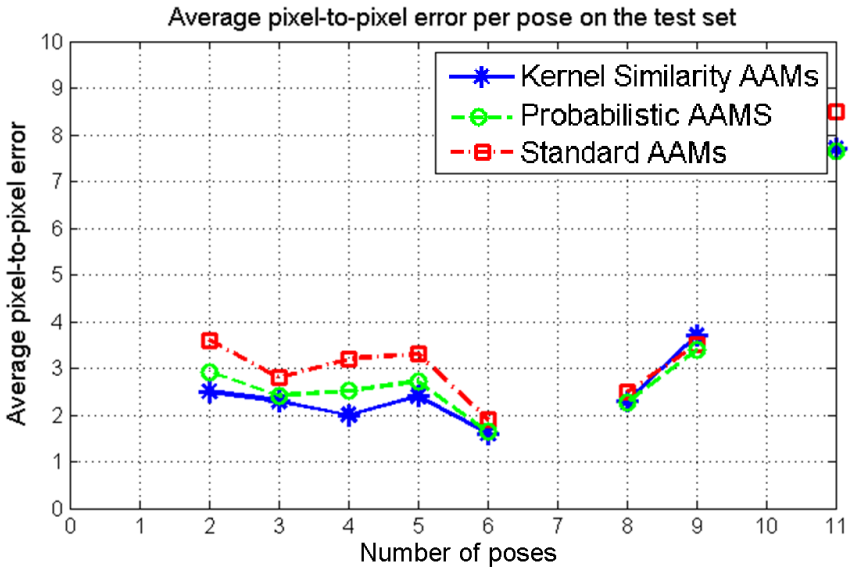

Figure 3.21: **Comparison curves between the proposed methods and classical AAMs** - Average Pixel-to-pixel Error on the test dataset

3.4 Conclusion

In this chapter, we proposed two solutions to extract the facial features by non-linear embedding model, in order to build an appearance parameter which has ability to synthesize the non-linear variations caused by complex illumination and face rotation. In this purpose, we have proposed a Kernel Similarity PCA and a Probabilistic PCA based AAMs. These two embedding model turn the classical AAMs to the non-linear deformable model.

Our algorithms have been tested on facial images from IMM database and CMU PIE database to extract the appearance parameters making non-linear illuminated variations. Results of the comparison of our proposed methods have shown that the model build by the proposed non-linear embedding methods are less sensitive to the illumination variations. The fitting performances indicate a powerful improvement on the illumination case especially on KS-PCA based AAMs. Probabilistic PCA model also brought better performance than classical AAMs but not as obvious as KS-PCA model. With these novel methods, the fitting procedure can accurately synthesize faces semi-bright-semi-dark affected by the illumination. Meanwhile, conditions with a variety of poses also benefit from the proposed algorithms; the ability of synthesizing faces with shape variations from a wide range of face poses has been improved. The problem of the synthesis of the complete profile faces is still waiting to be solved.

4

Sparse Representation embedding

Sparse and redundant representation model, as referred in chapter 2 is a novel component of the Embedding models. It assumes an ability to describe signals as linear combinations of a few atoms from a pre-specified dictionary. As such, the choice of the dictionary that sparsifies the signals is crucial for the success of this model. This chapter presents our research of two Dictionary-Learning Sparse Representation Models for dealing with a face pose estimation problem. Section 4.1 provides a brief introduction of the human face pose estimation problems. Section 4.2 describes the state-of-the-arts of the sparse representation approaches, and the applications of this embedding model in computer vision field. Our contribution is reflected in section 4.3 by two dictionary-learning algorithms for the face pose classification problem. In subsection 4.3.1, we propose to learn the dictionary via Incremental Principal Component Analysis (Incremental-PCA). By combining this linear embedding model and an updated learning strategy, we successfully eliminate the intra-class redundancy for each pose, while remain the inter-class redundancy for applying sparse representation based classification. In subsection 4.3.2 we propose a method which can unify the classification criterion and the optimism goal for learning the sparse representation dictionary. In this way, the classification can be done based on the pre-learned dictionary which insures the success of the classification. In this algorithm, the sparse representation is searched by Orthogonal Matching Pursuit (OMP) [126] and the dictionary is learned by Analysis K-SVD algorithm [127]. The OMP and Analysis K-SVD algorithms are

novel developments in embedding model domain. Experimental results in section 4.3.3 demonstrate the effectiveness of the proposed Dictionary-Learning Sparse Representation Models for treating the pose classification in dynamic illumination condition and low-resolution images, and showing a meaningful recovery of the analysis dictionary.

4.1 Introduction

Human face pose estimation from 2D images is a hot research topic, with lots of applications in expression and face recognition [128], driver monitoring [129] or human-computer interaction [130]. In a computer vision context, head pose estimation is the process of inferring the orientation of a human head from digital imagery. It requires a series of processing steps to transform a pixel-based representation of a head into a high-level concept of direction. Like other facial vision processing steps, an ideal head pose estimator must demonstrate invariance to a variety of image-changing factors. These factors include physical phenomena like camera distortion, projective geometry, multi-source non-Lambertian lighting, as well as biological appearance, facial expression, and the presence of accessories like glasses and hats. In spite of the tremendous achievements, there are still many challenges due to the large face appearance variations of expressions, illuminations, noise, occlusions etc.

Although it might seem like an explicit specification of a vision task, head pose estimation has a variety of interpretations. At the coarsest level, head pose estimation applies to algorithms that identify a head in one of a few discrete orientations, e.g., a frontal versus left/right profile view. At the fine level, a head pose estimate might be a continuous angular measurement across multiple Degrees of Freedom (DOF). A system that estimates only a single DOF, perhaps the left to right movement is still a head pose estimator, as is the more complex approach that estimates a full 3D orientation and position of a head, while incorporating additional degrees of freedom including movement of the facial muscles and jaw.

A review of related works can be found in [131]. According to the underlying implementation, the pose estimation approaches are organized in the following categories that describe the conceptual approaches.

Appearance Template Methods compare a new image of a head to a set of exemplars (each labelled with a discrete pose) in order to find the most similar view. In

the simplest implementation, the queried image is given the same pose that is assigned to the most similar of these templates. Some characteristic examples include the use of normalized cross-correlation at multiple image resolutions [132] and mean squared error (MSE) over a sliding window [133].

Appearance templates have some advantages over more complicated methods. The templates can be expanded to a larger set at any time, allowing systems to adapt to changing conditions. Furthermore, appearance templates do not require negative training examples or facial feature points. Creating a corpus of training data requires only cropping head images and providing head pose annotations. Appearance templates are also well suited for both high and low-resolution imagery.

There are many disadvantages with appearance templates. Without the use of some interpolation method, they are only capable of estimating discrete pose locations. They typically assume that the head region has already been detected and localized, whereas localization error can degrade the accuracy of the head pose estimate. They can also suffer from efficiency concerns, since as more templates are added to the exemplar set, more computationally expensive image comparisons will need to be computed. One proposed solution to these last two problems is to train a set of Support Vector Machines (SVMs) to detect and localize the face, and subsequently use the support vectors as appearance templates to estimate head pose [134, 135].

Regression based methods estimate the pose by learning a non-linear functional mapping from the image space to one or more pose directions. The motivation of these approaches is that with a set of labelled training data, a model can be built that will provide a discrete or continuous pose estimate for any new data sample. The caveat with these approaches is that it is not clear how well a specific regression tool will be able to learn the proper mapping.

The high-dimensionality of an image presents a challenge for some regression tools. Success has been demonstrated using Support Vector Regressions (SVRs) if the dimensionality of the data can be reduced, as for example with Principal Component Analysis (PCA) [136, 137], or with localized gradient orientation histograms [129] the latter giving better accuracy for head pose estimation. Alternatively, if the location of facial features are known in advance, regression tools can be used on relatively low-dimensional feature data extracted at these points [138, 139].

Among the nonlinear regression tools used for head pose estimation, neural networks have been the most widely used in the literature. An example is the multi-layer perceptron (MLP), consisting of many feed-forward cells defined in multiple layers (e.g. the output of the cells in one layer comprise the input for the subsequent layer) [140]. An MLP can also be trained for fine head pose estimation over a continuous pose range. In this configuration, the network has one output for each DOF, and the activation of the output is proportional to its corresponding orientation [141, 142].

A locally-linear map (LLM) is another popular neural network consisting of many linear maps [143]. To build the network, the input data is compared to a centroid sample for each map and used to learn a weight matrix. Head pose estimation requires a nearest-neighbor search for the closest centroid, followed by linear regression with the corresponding map. This approach can be extended with difference vectors and dimensionality reduction [144] as well as decomposition with Gabor-wavelets [145].

The advantages of neural network approaches are numerous. These systems are very fast, only require cropped labelled faces for training, work well in near-field and far-field imagery, and give some of the most accurate head pose estimates in practice.

The main disadvantage to these methods is that they are to prone to error from poor head localization. As a suggested solution, a convolutional network [146] that extends the MLP by explicitly modelling some shift, scale, and distortion invariance can be used to reduce this source of error [147].

Manifold Embedding Methods. Although an image of a head can be considered a data sample in a high-dimensional space, there are inherently many fewer dimensions in which pose can vary. With a rigid model of the head, this can be as few as three dimensions for orientation and three for position. Therefore, it is possible to consider that each high-dimensional image sample lies on a low-dimensional continuous manifold constrained by the allowable pose variations. For head pose estimation, the manifold must be modelled, and an embedding technique is required to project a new sample into the manifold. This low-dimensional embedding can then be used for head pose estimation with techniques such as regression in the embedded space or embedded template matching. Any dimensionality reduction algorithm can be considered an attempt at manifold embedding, but the challenge lies in creating an algorithm that successfully recovers head pose while ignoring other sources of image variation.

Two of the most popular dimensionality reduction techniques, principal component analysis (PCA) and its nonlinear kernelized version KPCA, discover the primary modes of variation from a set of data samples. Head pose can be estimated with PCA, by projecting an image into a PCA subspace and comparing the results to a set of embedded templates [148]. It has been shown that similarity in this low-dimensional space is more likely to correlate with pose similarity than appearance template matching with Gabor-wavelet preprocessing [149, 150]. Nevertheless, PCA and KPCA are not very efficient techniques for head pose estimation [151]. Besides the linear limitations of standard PCA that cannot adequately represent the nonlinear image variations caused by pose change, these approaches are unsupervised techniques that do not incorporate the pose labels that are usually available during training. As a result, there is no guarantee that the primary components will relate to pose variation rather than to appearance variation. Probably, they will be corresponding to both.

To alleviate these problems, the appearance information can be decoupled from the pose by splitting the training data into groups that each share the same discrete head pose. Then, PCA and KPCA can be applied to generate a separate projection matrix for each group. These pose-specific eigenspaces, or pose-eigenspaces, each represent the primary modes of appearance variation and provide a decomposition that is independent of the pose variation. Head pose can be estimated by normalizing the image and projecting it into each of the pose-eigenspaces, thus finding the pose with the highest projection energy [152]. Alternatively, the embedded samples can be used as the input to a set of classifiers, such as multi-class SVMs [153]. It has been shown that by skipping the KPCA projection altogether and using local Gabor binary patterns, one can greatly improve pose estimation with a set of multi-class SVMs [154]. Pose-eigenspaces have an unfortunate side-effect. The ability to estimate fine head pose is lost since, like detector arrays, the estimate is derived from a discrete set of measurements. If only coarse head pose estimation is desired, it is better to use multi-class linear discriminant analysis (LDA) or its kernelized version, KLDA, since these techniques can be used to find the modes of variation in the data that best account for the differences between discrete pose classes [151, 155].

Other manifold embedding approaches have shown good efficiency for head pose estimation. These include Isometric feature mapping (Isomap) [156, 157], Locally Linear Embedding (LLE) [158], and Laplacian Eigenmaps (LE) [159]. To estimate head

pose with any of these techniques, there must be a procedure to embed a new data sample into an existing manifold. Raytchev et al. [156] described such a procedure for an Isomap manifold, but for out-of-sample embedding in an LLE and LE manifold there has been no explicit solution. For these approaches, a new sample must be embedded with an approximate technique, such as a Generalized Regression Neural Network [160]. Alternatively, LLE and LE can be replaced by their linear approximations, locally embedded analysis (LEA) [161] and locality preserving projections (LPP) [162].

There are still some remaining weaknesses in the manifold embedding approaches mentioned thus far. With the exception of LDA and KLDA, each of these techniques operates in an unsupervised fashion, ignoring the pose labels that might be available during training. As a result, they have the tendency to build manifolds for identity as well as pose [160]. As one solution to this problem, identity can be separated from pose by creating a separate manifold for each subject that can be aligned together. For example, a high-dimensional ellipse can be fit to the data in a set of Isomap manifolds and then used to normalize the manifolds [163]. To map from the feature space to the embedded space, nonlinear interpolation can be performed with radial basis functions. Nevertheless, even this approach has its weaknesses, because appearance variation can be due to factors other than identity and pose, such as lighting. For a more general solution, instead of making separate manifolds for each variation, a single manifold can be created that uses a distance metric that is biased towards samples with smaller pose differences [160]. This change was shown to improve the head pose estimation performance of Isomap, LLE, and LE.

Another difficulty to consider is the heterogeneity of the training data that is common in many real-world training scenarios. To model identity, multiple people are needed to train a manifold, but it is often impossible to obtain a regular sampling of poses from each individual. Instead, the training images comprise a disjoint set of poses for each person sampled from some continuous measurement device. A proposed remedy to this problem is to create individual submanifolds for each subject, and use them to render virtual reconstructions of the discrete poses that are missing between subjects [164]. This work introduced Synchronized Submanifold Embedding (SSE), a linear embedding that creates a projection matrix that minimizes the distances between

each sample and its nearest reconstructed neighbours (based on the pose label), while maximizing the distances between samples from the same subject.

All of the manifold embedding techniques described in this section are linear or nonlinear approaches. The linear techniques have the advantage that embedding can be performed by matrix multiplication, but they lack the representational ability of the nonlinear techniques. As a middle ground between these approaches, the global head pose manifold can be approximated by a set of localized linear manifolds. This has been demonstrated for head pose estimation with PCA, LDA, and LPP [165].

Active Appearance Models (AAMs) learns the primary modes of variation in facial shape and texture from a 2D perspective.

AAMs have come a long way since their original inception. Fitting methods based on the inverse compositional image alignment algorithm overcome the linear assumption of how appearance error relates to the gradient descent search and allows more accurate, real-time convergence [96]. A tracked AAM over a video sequence can also be used to estimate the 3D shape modes, which can subsequently be reintroduced to constrain the 2D AAM fitting process [114]. Once the 3D constraint is learned, the AAM can be used to directly estimate the 3D orientation of the head. Alternatively, since the AAM shape points have a one-to-one correspondence, Structure From Motion (SFM) algorithms can be used to estimate the 3D shape of the face, as well as the relative pose difference between two video frames [166]. Further work with AAMs have introduced modifications that expand their utility to driver head pose estimation [167] and multiple cameras [115].

AAMs have good invariance to head localization error, since they adapt to the image and find the exact location of the facial features. This allows for precise and accurate head pose estimation. The main limitation of AAMs is that all of the facial features are required to be located in each image frame. In practice, these approaches are limited to head pose orientations from which the outer corners of both eyes are visible. It is also not evident that AAM fitting algorithms could successfully operate for far-field head pose estimation with low-resolution facial images.

Head pose estimation is a natural step for bridging the information between people and computers. This fundamental human ability provides rich information about the intent, motivation, and attention of people in the word. By simulating this skill, systems can be created that can better interact with people. The majority of head

pose estimation approaches assume the perspective of a rigid model, which has inherent limitations. The difficulty in creating head pose estimation systems stems from the immense variability in individual appearance coupled with differences in lighting, background, and camera geometry. In section 4.3, we present the proposed frameworks for overpass the difficulty of variable lighting conditions, as well as deal with more noisy issues.

4.2 State-of-the-arts

Techniques from sparse signal representation are beginning to see significant impact in computer vision, often on non-traditional applications where the goal is not just to obtain a compact high-fidelity representation of the observed signal, but also to extract semantic information. Sparse signal representation has been proven to be an extremely powerful tool for acquiring, representing, and compressing high-dimensional signals, and this is typically done by a pursuit algorithm that finds an approximate solution. In this section, we briefly discuss several such algorithms and their prospects for success.

4.2.1 Pursuit Algorithms

Sparse Representation relies on using an over-complete dictionary which contains prototype signal-atoms: signals are described by sparse linear combinations of these atoms. Recent activity in this field has concentrated mainly on two different ways to decompose signals with respect to a fixed or pre-learned dictionary: Convex Relaxation Techniques and Greedy Algorithms.

4.2.1.1 Convex Relaxation Techniques

Underdetermined systems of linear equations appear naturally in many important problems in science and technology, ranging from array signal processing to image processing to genomic data analysis. Such systems, with fewer equations than unknowns, may have many solutions, but often the solution of interest is the sparsest solution - the one having the fewest possible non-zeros. In "most" applications in science and technology, of course, the underlying model will not be perfectly correct and measurements will not be perfectly accurate. It is essential to use procedures which are robust against

the effects of measurement noise and modelling error. It is show in [168] that "most" matrices underlying underdetermined systems have the following property: when there exists any sufficiently sparse near-solution, the near-solution with minimal 'l_1-norm' is a good approximation to it. Here by sufficient sparsity, we mean that the number of non-zeros in the solution is only a certain fraction of the number of equations.

The use of the l_1-norm as a sparsity-promoting functional traces back several decades. A leading early application was reflection seismology, in which a sparse reflection function (indicating meaningful changes between subsurface layers) was sought from band-limited data. In 1979, Taylor, Banks and McCoy [169] proposed the use of l_1 to deconvolve seismic traces by improving on earlier ideas of Claerbout and Muir [170]. Over the next decade this idea was refined to better handle observation noise [171], and the sparsity-promoting nature of l_1-minimization was empirically confirmed. Rigorous results began to appear in the late-1980's, with Donoho and Stark [172] and Donoho and Logan [173] quantifying the ability to recover sparse reflectivity functions. The application areas for l_1-minimization began to broaden in the mid-1990's, as the LASSO algorithm [58] was proposed as a method in statistics for sparse model selection, Basis Pursuit [37] was proposed in computational harmonic analysis for extracting a sparse signal representation from highly over-complete dictionaries, and a related technique known as total variation minimization was proposed in image processing [174, 175].

Some examples of l_1-type methods for sparse design in engineering include Vandenberghe et al. [176, 177] for designing sparse interconnect wiring, and Hassibi et al. [178] for designing sparse control system feedback gains. In [179], Dahleh and DiazBobillo solve controller synthesis problems with an l_1-criterion, and observe that the optimal closed-loop responses are sparse. Lobo et al. used l_1-techniques to find sparse trades in portfolio optimization with fixed transaction costs in [180]. In [181], Ghosh and Boyd used l_1-methods to design well connected sparse graphs, in [182], Sun et al. observe that optimizing the rates of a Markov process on a graph leads to sparsity. In [183] [Section 6.5.4 and 11.4.1], Boyd and Vandenberghe describe several problems involving l_1-methods for sparse solutions, including finding small subsets of mutually infeasible inequalities, and points that violate few constraints. In a recent paper, Koh et al. used these ideas to carry out piecewise-linear trend analysis [184].

Over the last decade, the applications and understanding of l_1-minimization have continued to increase. Donoho and Huo [185] provided a more rigorous analysis of Basis

Pursuit, and this work was extended and refined in subsequent years, see [186, 187]. Much of the recent focus on l_1-minimization, however, has come in the emerging field of Compressive Sensing [188, 189]. This is a setting where one wishes to recover a signal x_0 from a small number of compressive measurements $y = \Phi x_0$. It has been shown that l_1-minimization allows recovery of sparse signals from remarkably few measurements [190, 191]: supposing Φ is chosen randomly from a suitable distribution, then with very high probability, all sparse signals x_0 can be perfectly recovered by using (P1). Moreover, it has been established [191] that Compressive Sensing is robust in the sense that l_1-minimization can deal very effectively (a) with only approximately sparse signals and (b) with measurement noise. The implications of these facts are quite far-reaching, with potential applications in data compression [188, 192], digital photography [193], medical imaging [194, 195], error correction [196, 197], sensor networks [198, 199].

4.2.1.2 Greedy Algorithms

A greedy strategy abandons exhaustive search in favor of a series of locally optimal single-term updates. It iteratively constructs a sparse approximant by maintaining a set of active vectors, initially empty, and at each stage, expanding that set by one additional vector. The active vector is chosen at each stage maximally reduces the residual l_2-error in approximating the signal from the currently active vectors. After constructing an approximant including the new vector, the residual l_2 error is evaluated; if it now falls below a specified threshold, the algorithm terminates.

In the past decade or so several efficient greedy pursuit algorithms have been proposed. The simplest ones are the matching pursuit (MP) [36] and the orthogonal matching pursuit (OMP) algorithms [200, 201]. These are greedy algorithms that select the dictionary atoms sequentially. These methods are very simple, involving the computation of inner products between the signal and dictionary atoms, and possibly deploying some least squares solvers. The underdetermined equations are easily addressed by changing the stopping rule of the algorithm.

Many variants on the above algorithm are available, offering improvements either in accuracy and/or in complexity. This family of greedy algorithms is well known and extensively used, and in fact, these algorithms have been re-invented in various fields. In the setting of statistical modelling, greedy stepwise Least-Squares is called forward stepwise regression. When used in the signal processing setting this goes by the name

of Matching-Pursuit (MP) or Orthogonal-Matching-Pursuit (OMP). Approximation theorists refer to these algorithms as Greedy Algorithms (GA), and consider several variants of them - the Orthogonal (OGA) [202], the Relaxed (RGA) [203], and the Weak Greedy Algorithm (WGA) [204]. The MP algorithm is similar to the OMP, but with an important difference that makes it simpler, and thus less accurate. In the main iteration, after the sweep and the update support stages, rather than solving a Least-Squares for re-evaluating all the coefficients in x, the coefficients of the S^{k-1} original entries remain unchanged, and the new coefficient that refers to the new member $j_0 \in S^k$ is chosen.

The Weak-MP is a further simplification of the MP algorithm, allowing for a sub-optimal choice of the next element to be added to the support. Embarking from the MP algorithm, the update support is relaxed by choosing any index that is the factor away from the optimal choice.

Both the Convex Relaxation Techniques and the Greedy Algorithms can be motivated based on maximum a posteriori (MAP) estimation, and indeed several works used this reasoning directly [205, 206]. The MAP can be used to estimate the coefficients as random variables by maximizing the posterior $P(x|y, D) \propto P(y|x, D)P(x)$. The prior distribution on the coefficient vector is assumed to be a super-Gaussian distribution that favors sparsity. For the Laplace distribution, this approach is equivalent to BP [205].

Extensive study of these algorithms in recent years has established that if the sought solution x is sparse enough, these techniques recover it well in the exact case [207, 208]. Further work considered the approximated versions and has shown stability in recovery of x [209, 210]. The recent front of activity revisits those questions within a probabilistic setting, obtaining more realistic assessments on pursuit algorithm performance and success [168]. The properties of the dictionary D set the limits on the sparsity of the coefficient vector that consequently leads to its successful evaluation.

4.2.2 Sparse Representation Applications for Computer Vision

For Computer Vision and Pattern Recognition, the efficient representations of data plays a critical role, and it has been shown again and again that learned and data adaptive dictionaries significantly outperform off-the-shelf ones such as wavelets. Current techniques for obtaining such dictionaries mostly involve their optimization in terms

of the task to be performed, denoising [71, 211], and classification [212]. Theoretical results addressing the stability and consistency of the sparse solutions (active set of selected atoms), as well as the efficiency of the coding algorithms, are related to intrinsic properties of the dictionary such as the mutual coherence, the cumulative coherence, and the Gram matrix norm of the dictionary [76, 207, 213, 214, 215]. Dictionaries can be learned by locally optimizing these and related objectives [62, 216].

4.2.2.1 Sparse Modelling for Image Restoration

Many problems in image processing and computer vision are in a dire need for prior models of the images they handle. This is especially true whenever information is missing, damaged, or modified. Armed with a good generic image prior, restoration algorithms become very effective. Numerous examples, comparisons, and applications in image demosaicing, image inpainting, and image denoising are presented in [211, 217]. It is important to note that for image denoising, over-complete dictionaries are used. Paper [211] extends the non-local means approach developed in [218, 219]. Interestingly, the two frameworks are quite related, since they both use patches as building blocks (in [211], the sparse coding is applied to all overlapping image patches), and while a dictionary is learned in [211] from a large dataset, the patches of the processed image itself are the "dictionary" in non-local means. The sparsity constraint in [211] is replaced by a proximity constraint and other processing steps in [219, 220]. The exact relationship and the combination of non-local-means with sparsity modelling has been recently exploited by the authors of [221] to further improve on these results. The authors also developed a very fast on-line dictionary learning approach.

4.2.2.2 Sparse Modelling for Image Classification

While image representation and reconstruction has been the most popular goal of sparse modelling and dictionary learning, other important image science applications are starting to be addressed by this framework, in particular, classification and detection. The work in [222] reported striking empirical results: the l_1-minimizer, has a strong tendency to separate the identity of the face from the error due to corruption or occlusion. In [223] and [224] the authors use the reconstruction/generative formulation, exploiting the quality of the representation and the coefficients for the classification tasks. This generative only formulation can be augmented by discriminative terms [221, 225, 226]

where an additional term is added to encourage the learning of dictionaries that are most relevant to the task at hand. The dictionary learning then becomes task-dependent and semi-supervised. In the case of [226] for example, a Fisher-discriminant type term is added in order to encourage signals (images) from different classes to pick different atoms from the learned dictionary. In [221], multiple dictionaries are learned, one per class, so that each class's dictionary provides a good reconstruction for its corresponding class and a poor one for the other classes (simultaneous positive and negative learning). This idea was then applied in [225] for learning to detect edges as part of an image classification system. These frameworks have been extended in [212], where a graphical model interpretation and connections with kernel methods are presented as well for the novel sparse model introduced there. Of course, adding such new terms makes the actual optimization even more challenging, and the reader is referred to those papers for details.

4.2.2.3 Learning to Sense

As we have seen, learning over-complete dictionaries that facilitate a sparse representation of the data as a linear combination of a few atoms from such dictionary leads to state-of-the-art results in image and video restoration and classification. The emerging area of compressed sensing (CS), [189, 227], and references therein, has shown that sparse signals can be recovered from far fewer samples than required by the classical Shannon-Nyquist Theorem. The samples used in CS correspond to linear projections obtained by a sensing projection matrix. It has been shown that, for example, a non-adaptive random sampling matrix satisfies the fundamental theoretical requirements of CS, enjoying the additional benefit of universality. A projection sensing matrix that is optimally designed for a certain class of signals can further improve the reconstruction accuracy or further reduce the necessary number of samples. In [216], the authors designed a framework for the joint design and optimization, from a set of training images.

4.3 Contribution

The pose estimation and classification is one of the classical problems in computer vision. Given a natural image that may contain a human face, it has been known that the

appearance of the face image can be easily affected by many image nuisances, including background illumination, pose, and facial corruption/disguise such as makeup, beard, and glasses. We contend that the problem can be solved quite effectively by a simple algorithm. The key observation is that a pre-learned, meaningful and orthogonal dictionary, combining with sparse representation algorithm is a promising classification strategy. Dictionary training is a much more recent approach to dictionary design, and as such, has been strongly influenced by the last advances in sparse representation theory and algorithms. The main advantage of trained dictionaries is that they lead to state-of-arts results in many practical signal processing applications. In this section, we propose two dictionary-learning frameworks for pose classification in variable illuminations and occluded images. In the first one, as presented in section 4.3.1, we combined a traditional transform embedding model - Incremental Principal Component Analysis and the l_1-norm sparsity measure to train a over-complete dictionary which contains only necessary variations for classification. Following this work, the second dictionary-learning idea is described in section 4.3.2. In this work, we unify the classification criterion and the optimisation goal for learning the sparse representation dictionary. Such that the classification can be done based on the pre-learned dictionary which insures the success of the classification. The experiments are shown in section 4.3.3. The classification results based on database CMU PIE [125] and LLP demonstrate the effectiveness of the proposed Dictionary-Learning Sparse Representation framework for treating the pose classification in dynamic illumination condition and low-resolution images.

4.3.1 Incremental Principal Component Analysis-based Sparse Representation for Face Pose Classification

In this section, we propose an Adaptive Sparse Representation pose Classification (ASRC) dictionary-learning algorithm which combines a traditional transform embedding model, Incremental Principal Component Analysis (referred to as Incremental-PCA) and the l_1-norm sparsity measure to train a over-complete dictionary. The key idea is a judicious choice of dictionary: representing the test signal as a sparse linear combination of the pre-learned dictionary. According to reference [222], Sparse representation is a reliable classification algorithm, since the l_1-norm sparsity measure is robust to noise and occlusions. But the way to construct the dictionary still need to be

discussed. Firstly, the sparse representation does need redundancy in the dictionary, and this kind of inter-class redundancy is provided by the pose images from different pose classes (since the dictionary is built on the combination of several pose classes). However, in each single pose class, there still exists intra-class redundancy between the samples if the dictionary is built on the dataset without selection. This kind of intra-class redundancy will affect the performance and efficiency of the representation of the target image. So we propose a conditional update learning strategy via Incremental-PCA. By applying Incremental-PCA to add only misclassified samples in the dictionary to reduce the intra-class redundancy in each pose class. A dictionary with less redundancy will improve both the classification performance and speed up the classification procedure. The final label for the test image is decided based on the over-complete dictionary built by the combination of the eigenspaces of all pose classes. Experimental results show that the proposed method is very robust when the illumination condition changes very dynamically and image resolutions are quite poor.

4.3.1.1 Incremental Principal Component Analysis

The incremental subspace learning algorithm is proposed in [228]. Let us consider a set of d-dimensional vectorized training images $I = \{I_1, I_2, \cdots, I_n\}$, where d is the dimension of each vectorized image. Classically, the initial eigenspace of the training images can be obtained by solving the singular value decomposition (SVD) of the covariance matrix:

$$C = \frac{1}{n}\sum_{i=1}^{n}(I_i - \overline{I})(I_i - \overline{I})^T \tag{4.1}$$

Where $\overline{I} = \frac{1}{n}\sum_{i=1}^{n} I_i$ is the mean image of image set I. The initial K eigenvalues $\{\lambda_1, \lambda_2, \cdots, \lambda_K\}$ lie on the diagonal of matrix $\Lambda \in R^{K\times K}$, and the corresponding eigenvectors are represented in matrix $U = \{u_1, u_2, \cdots, u_K\} \in R^{d\times K}$. This initial eigenspace defines an orthogonal linear transformation that projects the training images into a new coordinate system. The Incremental-PCA learning is then built based on the initial eigenspace. When a new training image I_{n+1} is considered, the incremental procedure updates the mean image and the eigenvectors as described in [228]. The mean image is updated as:

$$\bar{I}' = \frac{1}{n+1}(n\bar{I} + I_{n+1}) \tag{4.2}$$

Using the current eigenvectors U as the basis set, the new image I_{n+1} can be reconstructed, but with a loss represented by the residual vector v, computed as:

$$v = (U\alpha_{n+1} + \bar{I}) - I_{n+1} \tag{4.3}$$

where $\alpha_{n+1} = U^T(I_{n+1} - \bar{I})$. The vector v is then normalized:

$$\hat{v} = \frac{v}{\|v\|_2} \tag{4.4}$$

The updated eigenspace $U^{'}$ is acquired by a rotation, R, of the current eigenspace plus the residual vector:

$$U^{'} = \left[\begin{array}{cc} U & \hat{v} \end{array} \right] R \tag{4.5}$$

The rotation matrix R and updated eigenvalues $\Lambda^{'}$ can be obtained by solving the SVD of D matrix:

$$DR = R\Lambda^{'} \tag{4.6}$$

where we compose $D \in R^{(K+1)\times(K+1)}$ as:

$$D = \frac{n}{n+1}\left[\begin{array}{cc} \Lambda & 0 \\ 0^T & 0 \end{array} \right] + \frac{n}{(n+1)^2}\left[\begin{array}{cc} \alpha_{n+1}\alpha_{n+1}^T & \beta\alpha_{n+1} \\ \beta\alpha_{n+1}^T & \beta^2 \end{array} \right] \tag{4.7}$$

where $\beta = \hat{v}_{n+1}^T(I_{n+1} - \overline{I})$.

The solution to Equation 4.5 yields the new eigenvalues directly, and the new eigenvectors are then computed from Equation 4.4. The details of the whole procedure can be found in [228]. With this Incremental-PCA algorithm, the new training image I_{n+1} is taken into account by the basis eigenspace as a new type of variation.

4.3.1.2 Classification based on l_1-norm sparsity measure

In this section, we make use of the Sparse Representation Classification (referred to as SRC) method presented in [222] to generate a robust pose classification algorithm. Let us consider a classification problem with M distinct categories, then we build M eigenspaces $A_m = \left[\begin{array}{cc} \bar{I}_m & U_m \end{array} \right], m = 1, 2, \cdots, M$, on each category by Incremental-PCA learning mentioned in section 4.3.1.1, where $\bar{I}_m$ is the normalized mean image and U_m are the eigenvectors of the m^{th} pose class in the training samples. We suppose that the target can be classified with the arranged vector matrix A, where A is a combined eigenspace of M pose categories:

$$A = [A_1, A_2, \cdots, A_M] \tag{4.8}$$

With dictionary A, any new test image $y \in R^{d\times 1}$ can be approximately represented by a linear span of the training eigenspace associated with its class m:

$$y = A_m a_m \tag{4.9}$$

where $a_m = [1, a_m^1, a_m^2, \cdots, a_m^k]$ is the reconstruction coefficient vector in the m^{th} sub-eigenspace.

Practically, the test image y could be partially corrupted or occluded. In this case, the above model in equation 4.9 should be rewritten as:

$$y = \begin{bmatrix} A & E \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = Aa + Eb \tag{4.10}$$

where $a = [0, \cdots, 0, a_m, 0, \cdots, 0]$ is the sparse coefficient vector whose elements are zero except those contained in a_m, and $E = [\Omega, -\Omega] \in R^{d\times 2d}$ represents the additive trivial basis set for occluded or corrupted targets, where $\Omega \in R^{d\times d}$ is the identity matrix, and b is the coefficient vector on E. The optimal coefficients $\begin{bmatrix} a \\ b \end{bmatrix}$ are obtained by solving the following objective function:

$$l_0 = \underset{a,b}{Argmin} \left\|a\right\|_0 + \left\|b\right\|_0, \quad \text{subject to} \quad y = Aa + Eb \tag{4.11}$$

where $\left\|x\right\|_0$ counts the number of nonzero elements in x. In the case of our pose estimation, the size of the over-complete basis being larger than the dimention of images, Equation 4.11 is typically underdetermined, so that the solution is not unique. Recent development in the emerging theory of sparse representation and compressive sensing [168] reveals that if the solution $\begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$ is sparse enough, the solution of the l_0-minimization problem 4.11 is equal to the solution to the following l_1-minimization problem:

$$l_1 = \underset{a,b}{Argmin} \left\|a\right\|_1 + \left\|b\right\|_1, \quad \text{subject to} \quad y = Aa + Eb \tag{4.12}$$

This problem can be solved in polynomial time by standard linear programming method. A test sample y from one of the classes in the training set is represented with the solution $\begin{bmatrix} a_1 \\ b_1 \end{bmatrix}$ and the over-complete basis set $\begin{bmatrix} A & E \end{bmatrix}$. Ideally, the nonzero elements will all be associated with the columns of A from a single pose class m, and we can easily assign the test sample y to that class. But because of the noise in the data, there could be small nonzero elements associated with multiple object classes.

In this case, we compute a reconstruction residual RE_m for each single pose class and then classify y to the class corresponding to the minimal reconstruction error.

$$Label(m) = \underset{m}{Argmin}\, RE_m = \|y - (Aa_1^m + Eb_1)\|_2 \tag{4.13}$$

where a_1^m represents the elements from coefficient vector a_1 which is associated with the m^{th} class.

4.3.1.3 A framework of Incremental Principal Component Analysis-based Sparse Representation Classification

The framework of the Adaptive Sparse Representation pose Classification (ASRC) is explained in details in this section. The essential idea of this framework is to build an online face pose dictionary via Incremental Principal Component Analysis. However, the amount of the training images is large, and there exists a lot of intra-class redundancy variations and noise which may affect the final classification results. To overcome this situation, we devise a conditional update method that updates the training appearance basis eigenspace only with the misclassified face images. Algorithm 1 summarizes the complete classification scheme.

In order to provide a suitable training set for Incremental-PCA, for each class m, the training samples are divided into two subsets. The first one contains a small number of, n_m, randomly selected face images, and it is modelled via PCA as an initial eigenspace, and the second one is used for Incremental-PCA learning. The combination of the eigenspaces of all pose classes are used for Sparse Representation Classification as an over-complete dictionary. Afterwards, new training samples are added into the basis eigenspace by applying Incremental Principal Component Analysis in case of the Sparse Representation Classifier made a mistake. Therefore the dictionary of SR is updated with every misclassification. The incorporation of incorrectly classified pose images makes our classifier more adaptive.

The proposed conditional update of the training basis dictionary stabilizes classification accuracy and improves the classification performance especially when the image resolution is very low or the illumination condition changes dynamically.

Algorithm 1: Framework of the Adaptive Sparse Representation pose Classification (ASRC)

1:Input: a matrix of training images $I = \left\{I^1, I^2, \cdots, I^M\right\}$ from M classes, where $I^m \in R^{d \times n_m}$ represents the image set of the class m where d is the dimension of each image (in vector form), while n_m is the number of images in the class m.
2:Apply standard PCA on each image set to obtain the initial eigenspaces $A_m = \left[\begin{array}{cc} \bar{I}_m & U_m \end{array}\right] (m = 1, 2, \cdots, M)$.
3:Solve the l_1-minimization problem in Equation 4.12 for the sparse representation $\left[\begin{array}{c} a_1 \\ b_1 \end{array}\right]$ of a new image from the second training subset.
4:Label the new training image with the class m for which the reconstruction residual of the new training image is minimal in Equation 4.13.
5:Check if the new image is labelled with the correct class?
Yes → return to **step 3**;
No → continue to **step 6**.
6:Update the online eigenspace of the class to which the new training image belongs to via Incremental-PCA as in subsection 4.3.1.1.
7:Return to **step 3**.

4.3.2 A Dictionary-Learning Sparse Representation Model for Pose Classification

In this section, we present the other dictionary-learning research for dealing with the same pose classification issue as mentioned in the previous section. The key point of this novel algorithm is to search for adapting dictionaries so as to unify the classification criterion and the optimization goal for learning the sparse representation dictionary. The proposed Dictionary-Learning Sparse Representation framework is referred to as the DLSR Model in the following text. In this study, for the pose classification problem, firstly, we need to train a specific dictionary for each pose category. Then, considering a pose image to be classified, it can be represented by a linear combination of a few atoms from the pre-specified dictionaries. The class label is finally determined by comparing the reconstruction errors made by different pose classes. As such, the choice of the dictionary that sparsifies the signals is crucial for the success of this pose estimation problem. The proposed approach models the appearance of face images from the same

pose via a sparse model which learns the dictionary D from a set of image patches with the objective to minimize the reconstruction error of the target image, in order to coincide with the pose classification criterion. In this way, the classification can be done based on the pre-learned dictionary which insures the success of the classification. Then, the combination of the trained dictionaries of all pose classes are used as an over-complete dictionary for sparse representation and classification. From the experimental results in section 4.3.3, the performance of this dictionary-learning classification strategy is compared with the framework in the previous section and two state-of-arts classification results.

In Figure 4.1, we present our approach for the face pose estimation problem by using the DLSR algorithm based framework. Figure 4.1 shows both the dictionary learning and pose classification procedure, where the blocks on grey background represent the pose classification steps while the other blocks represent the dictionary learning steps.

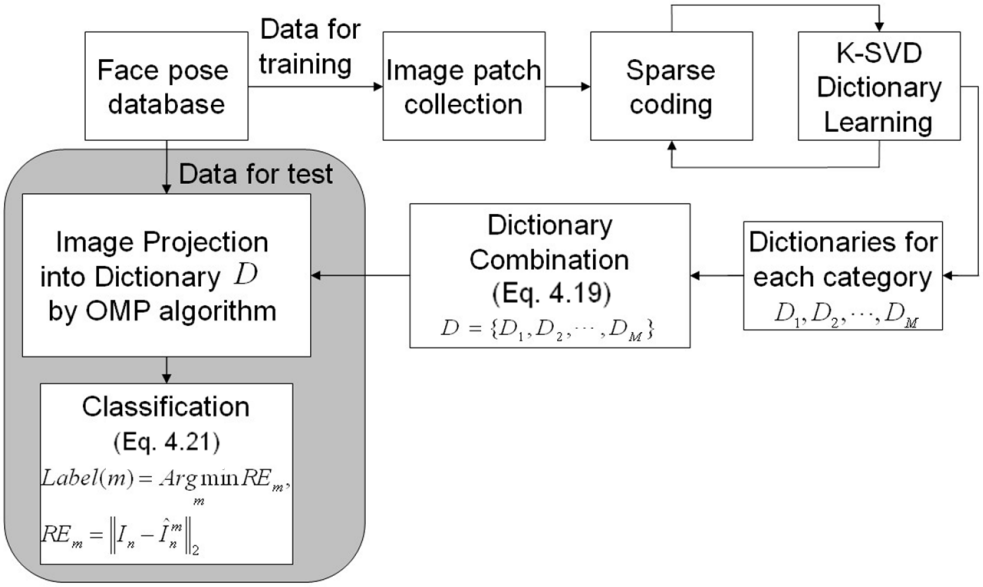

Figure 4.1: **Proposed strategy** - The Dictionary-Learning Sparse Representation framework for Pose Classification

For further explanation of the proposed work, let us consider a classification problem with M distinct categories. For each category, an initial redundant dictionary $D_0^m, m = 1, 2, \cdots, M$ is built by collecting all possible $n \times n$ image patches from the training data set. In the dictionary learning step, the Analysis K-SVD algorithm [36] is applied, for

searching the dictionary which can minimize the residual reconstruction error of the sparse representation and meanwhile keeps the representation coefficients sparse. As a result, M trained dictionaries $D_1, D_2, \cdots, D_M$ for all the M categories are gained. Then the last part of the training procedure is to combine all these M trained dictionaries, for an over-complete dictionary D, which has the ability to approach the test images from all possible categories with a minimal reconstruction error.

When we refer to the classification process, a target image is first projected into the over-complete dictionary by the Orthogonal Matching Pursuits (OMP) algorithm [126] to extract the sparse representation on dictionary D. The final decision of the classification label is made according to the minimal reconstruction error.

4.3.2.1 Sparse Representation Model

Signal processing techniques commonly require more meaningful representations which capture the useful characteristics of the signal for classification. Representing a signal involves the choice of a dictionary, which is the set of elementary signals - or atoms - used to decompose the signal. When the dictionary forms a basis, every signal is uniquely represented as the linear combination of the dictionary atoms.

For years, orthogonal and bi-orthogonal dictionaries were dominant due to their mathematical simplicity. However, the weakness of these dictionaries - namely their limited expressiveness - eventually outweighed their simplicity. This led to the development of newer over-complete dictionaries, having more atoms than the dimension of the signal, and which promised to represent a wider range of signal phenomena. Such dictionaries introduce an intriguing ambiguity in the definition of a signal representation. We consider the dictionary $D = [d_1 d_2 \cdots d_N] \in R^{d \times N}$, where the columns constitute the dictionary atoms, and $N \geq d$. Representing a signal $x \in R^d$ using this dictionary can take one of two paths: either the *analysis path* , where the signal x is represented via its inner products with the atoms,

$$\gamma_A = D^T x \tag{4.14}$$

or the *synthesis path* , where it is represented as a linear combination of the dictionary atoms,

$$x = D\gamma_S \tag{4.15}$$

The two definitions coincide in the complete case ($d = N$), when the analysis and synthesis dictionaries are bi-orthogonal. In the general case, however, the two may dramatically differ.

4.3.2.2 Analysis K-SVD Sparse Representation

We now describe the major idea of the dictionary learning algorithm introduced by R. Rubinstein et al. in [127] that we adapt to the pose classification criterion of this work. Consider a training set $X = \{x_1, x_2, \cdots, x_N\} \in R^{d \times N}$. For simplicity it is assumed that all example signals have the same sparsity level of s with respect to the dictionary D. In this work, the initial estimation D_0 is built from a patch set of the training data (described in detail in section 4.3.3.2). The followed optimization scheme is based on a two-phase block-coordinate-relaxation approach. In the first phase, X is optimized according to Equation 4.16 while keeping D fixed, and in the second phase, D is updated using the estimated signals Y. The process is repeated until the target sparsity level or a fixed number of iterations is achieved.

In this model, an emphasis is put on the zeros of $D \cdot x$, and define the co-support Λ of x as the set of index that indicates which entries in dictionary D are orthogonal to signal x. In other words, $D_\Lambda \cdot x = 0$, where D_Λ is a sub-matrix of D that contains only the rows indexed in Λ.

The K-SVD analysis is an iterative scheme that has a simple intuitive interpretation. Each iteration consists of two stages. In the first stage, we search for the set of rows in D that are "most orthogonal" to x. The second stage consists in updating each row in D to be the vector that is most orthogonal to all signals associated to it in the first stage. A detailed description is provided in Algorithm 2.

Given the dictionary D_0, optimizing Equation 4.16 for X can be done individually for each column of X by defining an ordinary sparse analysis approximation for each signal y_i,

$$\left\{y_i, \hat{\Lambda}_i\right\} = \underset{x_i, \Lambda_i}{Argmin} \left\|x_i - y_i\right\|, \quad \text{Subject to} \quad D_{\Lambda_i} \cdot x_i = 0, \quad \text{Rank}\left(D_{\Lambda_i}\right) = s \tag{4.16}$$

which may be solved by using the OMP algorithm as described in [126].

Once Y is computed, we turn to update D in the second step. For the j_{th} iteration in this step (as in step 6, Algorithm 2), the optimization is carried out sequentially for each row d_j in D. The update of d_j should be affected only by those columns of Y that are orthogonal to it, while the remaining signal examples should have no influence. Thus, letting Y_{Φ_j} denote the sub-matrix of Y containing the columns found to be orthogonal to d_j, the update step for d_j can be written as

Algorithm 2: Analysis K-SVD

1:Input: Training signals $X \in R^{d\times N}$, initial dictionary $D_0 \in R^{p\times d}$, target sparsity level s and number of iterations k

2:Output: Dictionary D and signal set Y minimizing Eq. 4.16

3:Initialization: Set $D := D_0$

4:for $n = 1 \cdots k$ **do**

5: **Analysis Pursuit:**

$$\forall i : \{y_i, \Lambda_i\} = \underset{x,\Lambda}{Argmin} \|x - y_i\|_2, \quad \text{Subject to} \quad D_\Lambda x = 0, \; Rank(D_\Lambda) = s$$

6: **for** $j = 1 \cdots p$ **do**

7: **Extract Relevant Examples:** $\Phi_j :=$ indices of the columns of Y orthogonal to d_j

8: **Compute Row:**

$$\hat{d}_j := \underset{d}{Argmin} \left\|d^T \cdot X_{\Phi_j}\right\|_2, \quad \text{Subject to } \|d\|_2 = 1$$

9: **Update Row:** $D\{\textbf{j-th row}\} := d_j^T$

10: **end for**

11: end for

$$\left\{\hat{d}_j, Y_{\Phi_j}\right\} = \underset{d_j, X_{\Phi_j}}{Argmin} \left\|X_{\Phi_j} - Y_j\right\|_2^2, \quad \text{Subject to} \quad D_{\Lambda_i} \cdot x_i = 0, \forall i \in \Phi_j, \; \text{Rank}\left(D_{\Lambda_i}\right) = s, \quad \|d_j\|_2 = 1. \tag{4.17}$$

This suggests that the "dictionary-update" stage uses only the co-supports (rather than the processed signals Y) that were found in the "sparse-coding" stage. Unfortunately, in general, solving the problem posed in Equation 4.17 is a difficult task. However, as an alternative, the following approximation is used to the above optimization goal [127]:

$$\hat{d}_j = \underset{d_j}{Argmin} \left\|d_j^T \cdot X_{\Phi_j}\right\|_2^2 \quad \text{Subject to} \quad \|d_j\|_2 = 1. \tag{4.18}$$

For this problem, the solution is the singular vector corresponding to the smallest singular value of X_{Φ_j}, which can be efficiently computed from the SVD of X_{Φ_j} , or

using some inverse power method. As we show in section 4.3.3, this simple approach turns out to be very reliable in recovering analysis dictionaries from sparse example sets.

4.3.2.3 Classification based on the trained over-complete dictionary

In this section, we explain in details the framework of the DLSR pose Classification. The essential idea of this framework is to build an over-complete dictionary, which assemble typical features from all the M categories (as we defined at the beginning of this section). Known from the previous subsection 4.3.2.1, by applying the Analysis K-SVD algorithm with the aim of minimizing reconstruction error and keeping a certain sparse level, we build M dictionaries $D_m, m = 1, 2, \cdots, M$, one for each category. Then, the over-complete dictionary D is composed as:

$$D = \begin{Bmatrix} D_1^T & D_2^T \cdots D_M^T \end{Bmatrix}^T \tag{4.19}$$

with dictionary D, any new test image $I_n \in R^{d\times 1}$ can be approximately represented by its analysis representation $\hat{I_n}$, by solving the following pursuit with the OMP algorithm [126]:

$$\left\{\hat{I_n}, \Lambda\right\} = \underset{\hat{I_n},\Lambda}{Argmin} \left\|I_n - \hat{I_n}\right\|_2, \quad \text{Subject to} \quad D_\Lambda \cdot I_n = 0, \quad \text{Rank}(D_\Lambda) = s + 1. \tag{4.20}$$

This analysis pursuit is the same one as shown in Algorithm 2 step 5, except that the sparse level is $s + 1$ instead of s, which is a relaxation of the searching constraint for ensuring that the reconstruction error can be minimized.

Ideally, for classification problem, the atoms with nonzero coefficients will all be associated with the columns of D from a single pose class m, and we can easily assign the test sample I_n to that class. But because of the noise in the data, there could be small nonzero elements associated with multiple object classes. In this case, we compute a reconstruction residual RE_m for each single pose class and then classify I_n to the class corresponding to the minimal reconstruction error, as in Equation 4.21:

$$Label(m) = \underset{m}{Argmin}\, RE_m, \quad RE_m = \left\|I_n - \hat{I}_n^m\right\|_2 \tag{4.21}$$

where $\hat{I}_n^m$ represents the estimated image which is composed by the dictionary atoms associated to the m_{th} pose class.

Given that the learning goal of dictionary D_m is to reconstruct the signals from the m_{th} initial dictionary D_0^m with minimal error, for a signal $I_n \in R^{d\times 1}$ from class m, the approximate representation by a linear span of the dictionary D_m is more reliable than the other $m-1$ dictionaries. Taking profit from the unity of the classification criterion and the dictionary learning goal, the proposed classification algorithm gains high accuracy.

4.3.3 Experimental Results

With the aim of evaluating the classification performance of the two proposed algorithms, we present the experimental results obtained from two different image databases. We first evaluate the robustness of the new methods versus illumination variations on the CMU PIE [125] database. Second, we evaluate the methods on the LIRIS Low-resolution Pose (LLP) database to demonstrate the effectiveness of the algorithms for noisy and low resolution images.

4.3.3.1 Database

- CMU PIE database

In 2000 Sim et al. collected a database, consisting of over 40,000 images of 68 subjects. (The total size of the database is about 40GB.) The database is called the CMU Pose, Illumination, and Expression (PIE) database [125]. To obtain a wide variation across pose, 13 cameras are employed in the CMU 3D Room. To obtain significant illumination variation, the 3D Room is augmented with a "flash system" similar to the one constructed by Athinodoros Georghiades, Peter Belhumeur, and David Kriegman at Yale University. Since the images are captured with, and without, background lighting, there are totally $21 \times 2 + 1 = 43$ different illumination conditions. Finally, the subjects are asked to pose with several different "expressions." In particular, the "expressions" contains a neutral expression, a smile, a blink (i.e. shut their eyes), and a talk. These are probably the four most frequently occurring "expressions" in everyday life.

In our work, we build our dictionary-learning strategies for the pose classification problem in variable illumination conditions. We select the images which are captured without background lighting, since the illumination conditions in this case are the most complex for the classification issue. As illustrated in Figure 4.2, the chosen subset consists of 12,240 images of 68 people (9 different poses as shown in the top row of

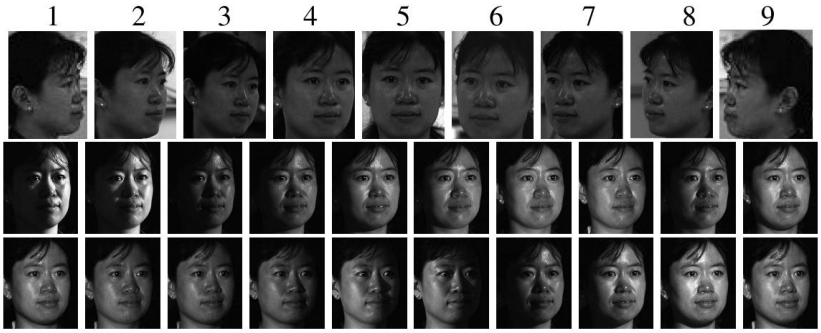

Figure 4.2: **CMU PIE database examples** - Top row: 9 different face pose images under the same illumination condition. Bottom two rows: 20 different illumination conditions.

Figure 4.2, and 20 variable illumination conditions (without background lighting) of each pose (in the bottom rows of Figure 4.2).

- LLP database

We built a large database of 248,025 face images sorted in five categories (as shown in Figure 4.3): 64770 frontal faces; 30469 left profile; 39993 right profile; 56396 quarter left; 56397 quarter right.

Figure 4.3: **LLP database examples** - Some examples of the face images from the LIRIS Low-resolution Pose database.

We collect images containing faces from the net and some other face databases. To help extracting the face images, we use the commercial face detector face.com (http://face.com). Using a script, we processed thousands of images containing faces

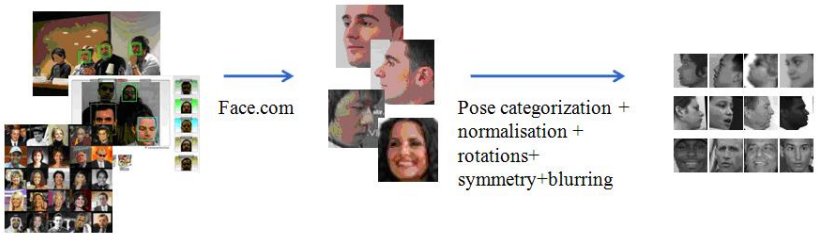

Figure 4.4: **LLP database construction procedure** - Manually face training database construction.

and retrieve their localizations and out-of-plane angle deviations. With manual selections/corrections, face images were cropped and classified into the five pose categories. Finally, several transformations were applied to the existing images to increase their number and their variability: in-plane rotations, translations, horizontal flipping and blurring. They were all resized to 36×36 images (as shown in Figure 4.4).

4.3.3.2 Parameter discussions

For the proposed ASRC and DLSR algorithms, there exist several parameters to be discussed.

- For Adaptive Sparse Representation pose Classification (ASRC)

	9×9	18×18	24×24	30×30	36×36
Class Rate	81.88%	91.03%	91.78%	88.46%	82.41%

Table 4.1: Classification rates for different image resolutions for ASRC algorithm

For Algorithm ASRC, we need to choose the best face image size for classification in the following experiments. In this experiment, we randomly select 1000 images from each pose class of LLP database, and downsampled the original image from 36×36 pixels to 30×30, 24×24, 18×18 and 9×9 pixels. The initial eigenspace for Incremental-PCA learning is built on 50 images, then we run the dictionary-learning process (as described in section 4.3.1.3, Algorithm 1) on the other 950 images. We tested the method on 5 poses from $-90°$ to $+90°$ with a step of $45°$. In each pose class, there are 1000 test images. For every resolution, the learning and test procedures are repeated 10 times to achieve a reliable result. The average classification rates for the different face image resolutions are shown in Table 4.1, where the two best classification rates of 91.03%

and 91.78% are obtained respectively for the resolution of 18×18 and 24×24. This result show that the proposed framework is very efficient on a very challenging database and therefore, in the following experiments, we subsample the original images to the resolution 24×24.

	Initial	100	500	1000	3000	6000
AVG Rate	75.40%	80.60%	87.87%	92.89%	91.13%	91.67%
Variance	6.17%	6.04%	1.47%	1.42%	1.32%	0.26%

Table 4.2: Variance and average classification rate on variable sizes of the training set (experiments are repeated 10 times) for ASRC algorithm

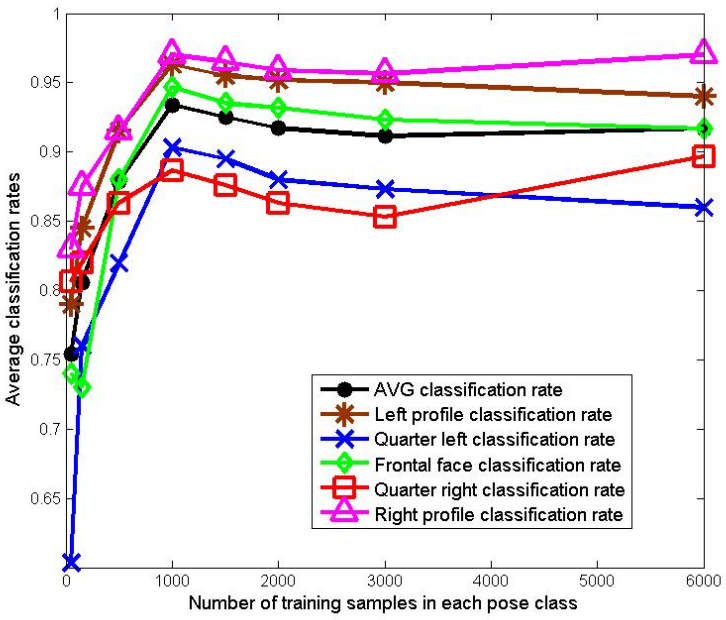

Figure 4.5: **Experiment result of parameter discussion** - Average classification performance for varying numbers of training samples for 5 pose categories, on the LLP database for ASRC algorithm.

In the second experiment, we tested the sensitivity of the face pose estimation versus the number of samples in the training set. The initial eigenspace for incremental-PCA is built on 50 randomly selected images from each pose class of LLP database. Then we add varying numbers of training samples to update the eigenspace, until the number of training samples reach 100, 500, 1000, 1500, 2000, 3000, 6000 (per class) respectively.

To achieve a reliable result, we repeat the training procedure 10 times, then apply the framework on the same test set which included 8000 test samples for each single pose class. All the training sets are constructed from a random selection of the entire LLP database. Both training and test sets are based on face images of resolution 24×24. The results of the experiments are illustrated in Figure 4.5 and Table 4.2. Figure 4.5 shows the average classification rate for each pose class. The average global classification rate for all classes is also presented. One can notice that the highest classification rate is obtained for 1000 training samples per class. Table 2 shows the mean accuracy and variance on various sizes of training set for 10 experiments. The highest classification rate of $91.03\% \pm 1.42\%$ is obtained for 1000 training samples per class. After this peak, with the number of training samples increasing, the classification rate remains stable.

- For Dictionary-Learning Sparse Representation framework (DLSR)

For the DLSR algorithm, the orthogonal dictionary is learned initially from a set of image patches which are collected form the same pose category. So the first influential parameter to discuss is the size of the patches for the initial dictionary. In this experiment, we use the CMU PIE database to discuss this patch size. For each pose class, we extract all possible $n \times n$ image patches for building the initial dictionary D_0^m, then, we apply the Analysis K-SVD algorithm on D_0^m for training a dictionary as explained in section 4.3.2.1. Some examples of patches extracted from the right profile pose (pose 1 in Figure 4.2) are shown in Figure 4.6, with the patch size 10×10 pixels. We apply 50 iterations of the Analysis K-SVD algorithm on this training set, learning an analysis dictionary. The initialization of the dictionary is the same as in Algorithm 2, and an example of the trained dictionary of the Analysis K-SVD algorithm is in Figure 4.7. The test images are represented in two-stage. First, the test images are segmented into distinct $n \times n$ patches, then each patch is represented as a linear combination of a few atoms from the pre-specified dictionary. In the second stage, the reconstructed distinct patches are recomposed for obtaining the final estimated image. We sum the reconstruction error of each patch for calculating the reconstruction error (RE_m) of the whole test image. The learned analysis dictionary is then utilized to represent each distinct patch decomposed from the target image. For representing the image, we apply the Orthogonal Matching Pursuit (OMP). The patch representation stage is followed by composing the distinct patches recovered by the learned dictionary to obtain the final estimated image. This approach is referred to as analysis patch-based image representation.

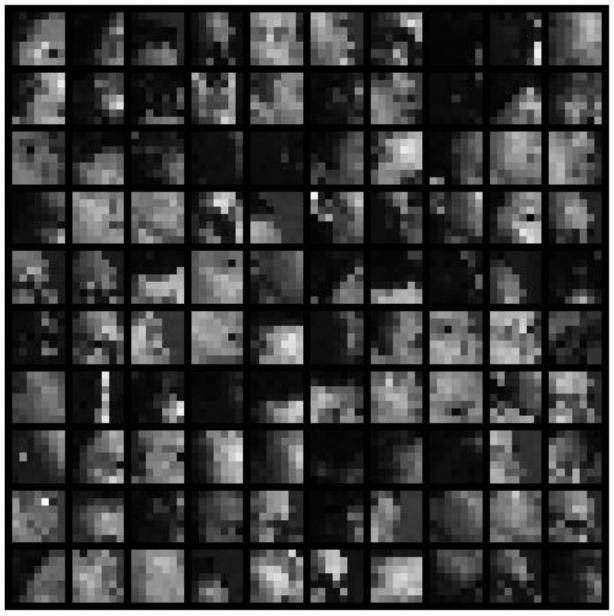

Figure 4.6: **Experiment result for parameter discussion** - Patch examples used for training the DLSR algorithm

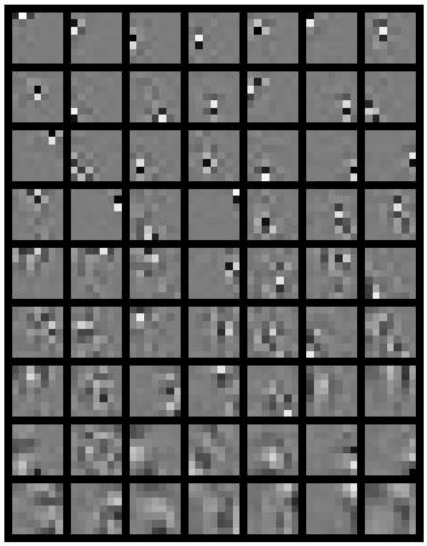

Figure 4.7: **Experiment result for parameter discussion** - Examples of the atoms from the trained DLSR Dictionaries

To choose the best patch size for classification, in the following experiments, we tested the algorithm by extracting different size of patch from differently subsampled images. The patch sizes are chosen as: 4×4, 6×6, 8×8, 10×10, 12×12, 14×14 and 16×16 pixels. The pose images are subsampled to size: 230×280, 115×140, 58×70, 29×35 and 15×18 pixels respectively. The average classification rates for different resolutions and different patch sizes are shown in Figure 4.8, where the best classification rate of 98.2% is obtained for the image resolution of 15×18, with patch size 6×6. This result shows that for this pose classification problem, the dictionary built from low resolution images, which presents more global pose information, performs better than those built from high-resolution images.

4.3.3.3 Performance on face pose classification with illumination variations

To evaluate the robustness of the two proposed methods to illumination variations, we apply the algorithms to the CMU PIE database. The database contains face images of size 640×486 pixels for 68 people across 13 poses, under 43 different illumination conditions, and with 4 different expressions. The chosen subset consists of 12,240 images of 68 people (9 different poses as shown in the top row of Figure 4.2, and 20

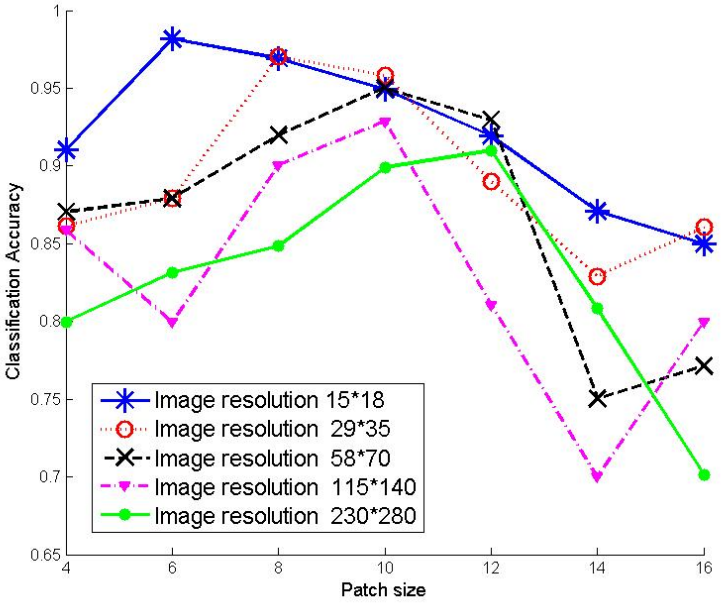

Figure 4.8: **Experiment result for parameter discussion** - Average classification rates for different resolutions and different patch sizes for DLSR algorithm.

variable illumination conditions without background lighting of each pose illustrated in the second and third rows of Figure 4.2).

- Performance of ASRC algorithm

For Algorithm ASRC, in the training procedure, we use images from 30 people (180 samples per person) of varying illuminated pose views. The faces are first located by the Viola-Jones face detector [229]. The face region is then down sampled to a 24×24 patch. The initial eigenspace for Incremental-PCA is built from 360 images (including 9 pose and 20 illumination conditions) of 2 people, and the images of the other 28 people in the training set are then added into the basis eigenspace by applying Incremental-PCA. The confusion matrix of the classification results for the 9 different poses is shown in Table 4.3.

Table 4.3 shows the confusion matrix, where the classification rate is obtained for each pose category by applying algorithm ASRC. One can notice that the classification confusions are low and generally between close categories.

- Performance of DLSR algorithm

Class	1	2	3	4	5	6	7	8	9
1	93.7%	5.7%	0.6%						
2		97.2%	2.8%						
3		0.8%	97.5%	1.7%					
4				94.9%	5.1%				
5				0.9%	98.2%	0.9%			
6						95.0%	4.2%	0.8%	
7						0.6%	95.8%	3.6%	
8						0.08%	4.2%	93.2%	2.5%
9							1.9%	5.9%	92.2%

Table 4.3: Confusion matrix of face pose classification for ASRC algorithm on the CMU PIE database (in percentage) for the 9 different poses

Class	1	2	3	4	5	6	7	8	9
1	100%								
2		97.4%	2.6%						
3		1.7%	97.2%	1.1%					
4			1.6%	97.8%	0.6%				
5				0.2%	98.8%	1.0%			
6				0.08%	0.8%	97.9%	1.2%		
7					0.13%	1.5%	97.5%	0.9%	
8							0.3%	98.0%	1.7%
9							0.3%	0.7%	99.0%

Table 4.4: Confusion matrix of face pose classification for DLSR algorithm on the CMU PIE database (in percentage) for the 9 different poses

For the DLSR algorithm, the selected subset of CMU is also divided into two parts, the first part with images of 30 people as the training set, while the second part with images of the other 38 people as the test set. The faces are first located by the Viola-Jones face detector [229]. Then, the face region is down-sampled to a 15×18 sub-image. For each pose class, we extract all possible 6×6 (according to the parameter discussion results in section 4.3.3.2, figure 4.8) image patches for building the initial dictionary D_0^m. Then, we apply the Analysis K-SVD algorithm on D_0^m for training a dictionary as explained in section 4.3.2.1. We apply 50 iterations for learning an analysis dictionary.

The confusion matrix of the classification results based on algorithm DLSR for the 9 different poses is shown in Table 4.4. Judging from this result, the proposed framework appears very efficient on a challenging database.

4.3.3.4 Performance on low-resolution, noisy face images

For testing the proposed algorithms with noisy and low-resolution images, we test the proposed methods on LLP database (as described in section 4.3.3.1, Figure 4.3).

- Performance of ASRC algorithm

For Algorithm ASRC, in training procedure, the initial eigenspace for incremental-PCA is built on 50 randomly selected images from each pose class of LLP database, sub-sampled to size 24×24. Then we add additional training samples to update the eigenspace, until the number of training samples reach 1000 (per class), to achieve the highest classification rate. The confusion matrix of the classification results for five different poses is shown in Table 4.5. It is clearly shown that the classification confusions are low.

Class	L	QL	F	QR	R
L	92.66%	6.68 %	0.66%		
QL	3.86%	91.55%	4.59%		
F	0.81%	1.2%	95.65%	1.9%	0.36%
QR			4.03 %	89.74%	6.24%
R			0.16%	8.83%	91.00%

Table 4.5: Confusion matrix of face pose classification for ASRC algorithm on the LLP database

Then we compare the ASRC framework with the standard PCA based eigenspace SRC. For standard PCA, the eigenspace dictionary is built when the number of training samples (the same training samples that we used for Incremental-PCA eigenspace) equals to 100, 500, 1000, 1500, 2000, 3000, 6000 (per class) respectively. The average classification rates for different size of the training set is shown in Figure 4.9. We can notice that when the number of training samples is below 500, the results of the two classifier are close, because when the training sets are small, the intra-class redundancy is low. And it is clear that, Incremental-PCA outperforms PCA when the number of training samples is over 500. Thanks to the online eigenspace, the effect of intra-class redundancy is eliminated by the conditional updated learning strategy.

In the last experiment of this section, we compare the ASRC framework with three classical algorithms, which are K-Nearest Neighbour (KNN), and Support Vector Machine (SVM), with linear and non-linear kernels. The Polynomial Kernel SVM learns

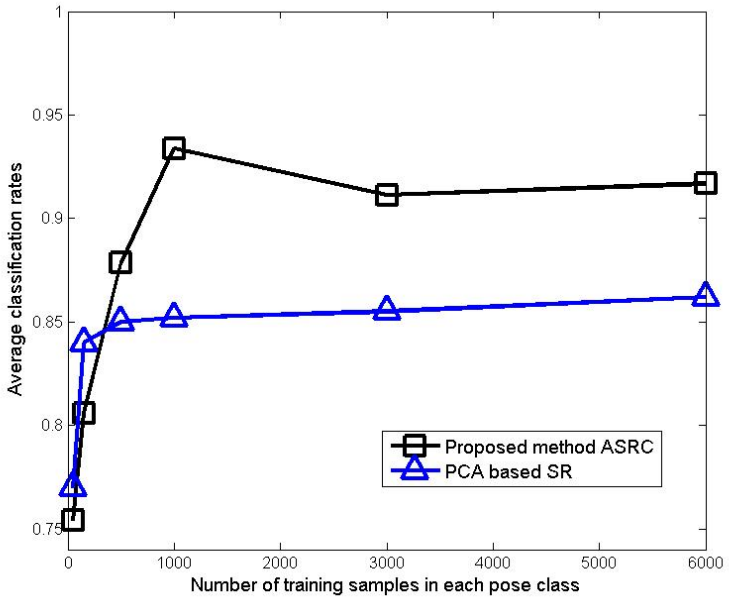

Figure 4.9: **Comparative Results 1** - Classification rate of the ASRC algorithm compared with PCA based SR classification algorithm

the hyperplanes using a polynomial kernel $K(x, y) = (x^T y + c)^d$ $(d = 2)$. And for the kNN based approach, an object is classified by a majority vote among its 10 neighbours in Euclidian space. For fair comparison, we use the same training and test sets as well as the same feature space for each experiment. As shown in Figure 4.10, the proposed algorithm outperformed the KNN (K=1, K=100) and SVM classifiers (with linear and polynomial kernels) in these low-resolution and high-noise conditions, for all sizes of the training set.

- Performance of DLSR algorithm

Table 4.6 shows the confusion matrix, where the classification rate is obtained by DLSR for each pose category. According to the result of the patch size experiment in Figure 4.8, a patch size of 8×8 is chosen for this 36×36 resolution classification problem. The classification confusions are low and generally between close categories (i.e. L and QL, R and QR). The proposed DLSR method appears to be particularly robust to deal with noisy and low-resolution images.

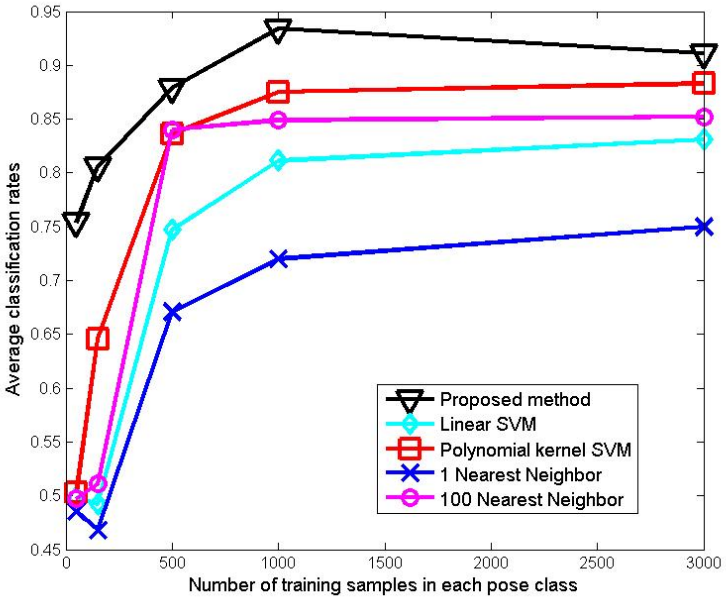

Figure 4.10: **Comparative Results 2** - Classification rate of the ASRC algorithm compared with SVM and K-Nearest Neighbours classification

Class	L	QL	F	QR	R
L	96.30%	2.68 %	1.02%		
QL	2.86%	95.72%	1.42%		
F	0.36%	1.20%	97.25%	1.19%	
QR			1.06 %	96.91%	2.03%
R				1.44%	98.56%

Table 4.6: Confusion matrix of face pose classification for DLSR algorithm on the LLP database

4.3.3.5 Comparison with other classification methods

In this section, we compare the proposed methods on database CMU PIE with two classical approaches, the first one based on Polynomial Kernel SVM [230] and the second one based on k-Nearest Neighbours [231] (k=10). For SVM and kNN, the feature space is built directly with the subsampled training images of resolution 24×24 (same size as for ASRC training). The Polynomial Kernel SVM learns the hyperplanes using a polynomial kernel $K(x, y) = (x^T y + c)^d$ $(d = 2)$. For the kNN based approach,

Class	1	2	3	4	5	6	7	8	9	AVG
DLSR	100%	97.4%	97.2%	98.0%	97.8%	98.8%	97.9%	97.5%	99.0%	98.12%
ASRC	97.2%	96.0%	97.1%	94.6%	98.3%	96.2%	97.0%	95.7%	96.8%	96.5%
SRC	91.0%	93.3%	96.1%	91.8%	93.1%	88.2%	94.3%	92.2%	86.4%	91.82%
PSFS	96.8%	78.4%	65.0%	79.5%	96.8%	93.8%	53.4%	90.0%	92.5%	84.91%
Polynomial Kernel SVM	84.4%	83.5%	87.8%	84.6%	91.5%	85.7%	86.3%	88.0%	86.2%	86.44%
kNN(k=10)	82.8%	80.9%	83.7%	79.7%	84.8%	87.1%	81.3%	85.1%	86.2%	83.51%

Table 4.7: Comparison of the classification rates between Kernel SVM, k-Nearest Neighbors, PSFS algorithm, SRC algorithm and the proposed ASRC, DLSR algorithms on the CMU PIE database (in percentage) for the 9 different poses

an object is classified by a majority vote among its 10 neighbors in Euclidian space.

The proposed ASRC and DLSR algorithms are also compared with two state-of-the-art classification methods [232] and [222]. The classification method in [232] is based on a Pose Similarity Feature Space (referred as PSFS). The authors classify different poses via an AdaBoost classifier combined with statistical procedure. The evaluation is also performed on the CMU PIE database. For fair comparison, the training and test data sets in all the approaches are unified. There are 15 people included in the training set, 4 illumination variations corresponding to flashes 01, 04, 13 and 14, and expression variations are neutral, smiling and blinking. The other images from 53 people are used for test. The approach proposed in [222] is a well-known face recognition method based on l_1-norm sparse representation minimization (referred to as SRC). In this work, the authors used downsampled images directly for feature extraction, since the classification procedure by the theory of sparse representation is robust. This framework is proved to be able to handle errors due to occlusion and corruption uniformly by exploiting the fact that these errors are often sparse with respect to the standard (pixel) basis. The theory of sparse representation helps predict how much occlusion the recognition algorithm can handle and how to choose the training images to maximize robustness to occlusion.

Table 4.7 compares the classification rates of the six approaches. The outperform of the two proposed frameworks is clear. Among ASRC and DLSR, the classification rates of the forward one are higher. This result is logical, since for ASRC, the dictionary training procedure based on Incremental-PCA can reduce the intra-class redundancy and build an orthogonal dictionary with necessary variations from each pose category. This strategy makes the classification progress avoid from the effect of intra-class redundancy. But it can not optimize the fundamental classification rule (to classify the test image according to the minimization reconstruct error) of the sparse representa-

tion. For DLSR, the optimization goal for training the pre-learned dictionary and the classification criterion are unified. The representative atoms are trained to obtain the minimal reconstruction error for each pose image. This idea eliminates the intra-class redundancy and coincides with the pose classification rule at the same time. Such that, the DLSR algorithm is more reasonable for the classification works, and provide higher classification rates.

4.4 Conclusion

In this chapter, we have presented two robust face pose estimator via dictionary-learning Sparse Representation frameworks. The first method trained the dictionary by an Incremental-PCA based updating framework, to reduce the intra-class redundancy of each pose class. The other one searched the over-complete dictionary with the optimization goal for coinciding with the pose classification rule.

In the two proposed strategies, both objectives of representing a test image are to find the sparse coefficients and to ensure that the representation error between the estimated image and original image is minimized. In the classification procedure, the test image is associated to the category which made the minimal reconstruction error.

In the experiments, some influential parameters are firstly discussed in section 4.3.3.2. The following evaluations are based on the optimal parameters. In the evaluation part, the two proposed algorithms are compared with two traditional classification methods and two state-of-the-art methods.

Comparing the two proposed frameworks, the DLSR provided better performance. This is because the optimization goal of DLSR for training the pre-learned dictionary and the classification criterion are unified. This idea eliminated the intra-class redundancy and coincided with the pose classification rule at the same time. So it is reasonable to get higher classification rates.

The ASRC and DLSR Classifiers are robust to appearance changes such as those caused by varying illumination, noise and resolutions. Experimental results show that the method improves the performance in terms of classification accuracy. In view of the classification results obtained on the CMU PIE database, the proposed algorithms exhibit a large tolerance to variations in illumination, and the experiments on the LLP database appears particularly robust for dealing with noisy and low-resolution images. The experiments have demonstrated a successful and meaningful recovery of the dictionary for the face representation.

5

Face Illumination Normalization via a jointly optimized Dictionary-Learning and Gaussian Mixture Model Clustering

For many applications, the performance of face recognition system in controlled environment has now reached a satisfactory level; however, there are still many challenges posed by uncontrolled environments. Some of these challenges are posed by the problems caused by variations in illumination, face pose, expression etc. The effect of variation in the illumination conditions in particular, which causes dramatic changes in the face appearance, is one of those challenge problems that a practical face recognition system needs to face. To be more specific, the varying direction and energy distribution of the ambient illumination, together with the 3D structure of the human face, can lead to major differences in the shading and shadows on the face. Such variations in the face appearance can be much larger than the variation caused by personal identity. The variations of both global face appearance and local facial features also cause problems for automatic face detection system.

In this chapter, we proposed a novel Dictionary Learning framework for Illumination Normalization (DL-IN). DL-IN based on sparse representation in terms of coupled dictionaries jointly optimized from normal illuminated and non-standard illuminated face image patch pairs. We further utilize a GMM model to enhance the framework's

capability of modeling data under complex distribution by adapting each model to a part of samples and fuse them together. This chapter is organized as follows, section 5.1 provides a brief introduction of the passive methods for eliminating illumination effect on face images. Our contribution is reflected in section 5.2. The effectiveness of the proposed algorithm in proved in section 5.3, and the conclusion in presented in section 5.4.

5.1 Introduction

Illumination variation is one of the most important factors which reduce significantly the performance of face recognition system. It has been proved that the variations between images of the same face due to illumination are almost larger than image variation in face identity [233]. So eliminating the effect due to illumination variations related directly to the performance and practical of face recognition system.

To handle face image variations due to changes in lighting conditions, many methods have been proposed thus far. Existing methods addressing the illumination variation problem, falls into two main classes: (1) Active methods and (2) Passive methods. Passive methods attempt to overcome this problem by studying the visible spectrum images in which face appearance has been changed by illumination variations, while active methods overcome the illumination variation by employing active imaging techniques to obtain face images captured in consistent illumination condition, or images of illumination invariant modalities. This section focuses only on passive methods, since our contribution is also a passive method.

Generally, the passive approaches can be divided into two main categories:

- Methods based on the illumination variation modeling technique. For instance, Lambertian Reflectance Model [234, 235], the 3D Morphable Model [81] and Subspace based Statistic Models [236, 237] are all well-known model based methods.

- Methods based on extracting illumination invariant features. For instance, Self-Quotient Image (SQI) model [238], Gradient method [239], Local binary patterns (LBP) [240, 241] ect. are proposed to deal with the illumination problem. In SQI model, the illumination invariant is obtained by division over a smoothed version of the image itself.

This section is organized as follows, section 5.1.1 presents the illumination variation based models. Section 5.1.2 is regarding illumination invariant feature based methods.

5.1.1 Illumination Variation Modelling

5.1.1.1 Lambertian Reflectance Model

The Lambertian model is based on Lambert's law [242]. According to Lambertâs law, if a light ray of intensity l coming from the direction u_l reaches a surface point with albedo ρ and normal direction v_r, the intensity i reflected by the point due to this light is given by

$$i = l(u_l) \cdot \rho \cdot max(u_l \cdot v_r, 0) \tag{5.1}$$

A 2-D face image based approach, which combines Eigen light field and Lambertian reflectance model, is presented by Zhou and Chellappa [242]. The generalized photometric stereo algorithm presented, combines the identity subspace with the illumination model and provides an illumination-invariant description. But assumes a fixed pose and cannot easily handle pose variations, i.e., its illumination-invariant identity description is not invariant to variations in pose. An important advantage of the approach is that it can generalize to novel illuminations. To make the recognition more robust under any situation, they assumed that the lighting conditions for the training, gallery and probe sets are completely unknown when recovering the identity signatures.

Basri and Jacobs [243] has proved that the set of all Lambertian reflectance functions obtained with arbitrary distant light sources lies close to a 9D linear subspace. In general, the set of images of a convex Lambertian object obtained under a wide variety of lighting conditions can be approximated accurately by a low-dimensional linear subspace. They also provide a simple analytic characterization of this linear space. They obtain these results by representing lighting using spherical harmonics and describing the effects of Lambertian materials as the analog of a convolution. Both the lighting function, l, and the Lambertian Kernel, k, can be written as sum of spherical harmonics. The harmonic expansion of l is defined by

$$l = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} l_{nm} Y_{nm} \tag{5.2}$$

where l_{nm} is amplitude of light at order n and Y_{nm} is an nth order harmonic. Kernel k is defined by

$$k(u) = \sum_{n=0}^{\infty} k_n Y_{n0} \tag{5.3}$$

An image of an object under certain illumination conditions can be constructed from the respective reflectance function in a simple way: each point of the object inherits its intensity from the point on the sphere whose normal is the same. This intensity is further scaled by its albedo.

Georghiades et. al. [244] present a generative appearance based method for recognizing human faces under variation in lighting and viewpoint. The method exploits the fact that the set of images of an object in fixed pose, but under all possible illumination conditions, is a convex cone in the space of images. Using a small number of training images of each face taken with different lighting directions, the shape and albedo of the face can be reconstructed. In turn, this reconstruction serves as a generative model that can be used to render or synthesize images of the face under novel poses and illumination conditions. The pose space is then sampled and, for each pose, the corresponding illumination cone is approximated by a low-dimensional linear subspace whose basis vectors are estimated using the generative model. They tested the method on 4,050 images from the Yale B Face Database; these images contain 405 viewing conditions (9 poses and 45 illumination conditions) for 10 individuals. The method performs almost without error, except on the most extreme lighting directions, and significantly outperforms popular recognition methods that do not use a generative model.

5.1.1.2 Linear Subspace

Existing methods dealing with just one of these variations are often unable to cope with the other variations. The problem is even more difficult in applications where only one gallery image per person is available. Chen, Lovell and Shan [245] describe a recognition method, Adapted Principal Component Analysis (APCA), which can simultaneously deal with large variations in both illumination and facial expressions using only a single gallery image per person. Experimental results show that APCA performs much better than other recognition methods including PCA and LDA. Gudur and Asari [236] presented Gabor Wavelet based Modular PCA (GW-MPCA) to deal with illumination and facial expression. The method divides face image into sub-images and then applies Gabor wavelet on each sub-image with different scale and orientation. Every image in the database is divided into N smaller sub-images. They found that performance of GW-MPCA is better than PCA and Modular PCA. Chen et. al. [237] addresses nonlinear feature extraction and Small Sample Size (S3) problems in face recognition. In sample feature space, the distribution of face images is nonlinear because of complex variations in illumination. The performance of classical linear method, such as Fisher Discriminant Analysis (FDA), will degrade. To overcome pose and

illumination problems, Shannon wavelet kernel is constructed and utilized for nonlinear feature extraction. Based on a modified Fisher criterion, simultaneous diagonalization method is exploited to deal with S3 problem, which often occurs in FDA based methods. Shannon Wavelet Kernel based Subspace Fisher Discriminant (SWK-SFD) method has been developed. The proposed method not only overcomes some drawbacks of existing FDA based algorithms, but is also computationally efficient. The proposed method gives superior results compared to existing FDA based methods.

5.1.1.3 3D Morphable Models

Face recognition across variations in a wide range of illuminations, including cast shadows and specular reflections is presented by Blanz and Vetter [81]. To account for these variations, the algorithm simulates the process of image formation in 3D space and it estimates 3D shape and texture of faces from single images. The estimate is achieved by fitting a statistical, Morphable Model of 3D faces to images. The model is learned from a set of textured 3D scans of heads. They describe the construction of the morphable model, an algorithm to fit the model to images, and a framework for face identification. In this framework, faces are represented by model parameters for 3D shape and texture. The morphable face model is based on a vector space representation of faces that is constructed such that any convex combination of shape and texture vectors S_i and T_i of a set of examples describes a realistic human face:

$$S = \sum_{i=1}^{m} a_i S_i \quad and \quad T = \sum_{i=1}^{m} b_i T_i \tag{5.4}$$

Continuous changes in the model parameters a_i generate a smooth transition such that each point of the initial surface moves toward a point on the final surface. Just as in morphing, artifacts in intermediate states of the morph are avoided only if the initial and final points are corresponding structures in the face, such as the tip of the nose. They tested performance of their algorithm using publicly available CMU-PIE database.

5.1.2 Illumination Invariant Features

This section presents methods for finding illumination invariant representation for illumination invariant face recognition. These methods are divided in main three sub categories named Retinex theory based methods, Gradient based method, and Local binary pattern based method.

5.1.2.1 Illumination plane subtraction with histogram equalization

The illumination plane $IP(x, y)$ of an image $I(x, y)$ corresponds to the best-fit plane from the image intensities. $IP(x, y)$ is a linear approximation of $I(x, y)$, given by

$$IP(x, y) = a \cdot x + b \cdot y + c \tag{5.5}$$

The plane parameters a, b and c can be estimated by the linear regression formula:

$$p = (N^T N)^{-1} N^T x \tag{5.6}$$

where $p \in R^3$ is a vector containing the plane parameters (a, b and c) and $x \in R^n$ is $I(x, y)$ in vector form (n is the number of pixels). $N \in R^{n \times 3}$ is a matrix containing the pixel coordinates: the first column contains the horizontal coordinates, the second column the vertical coordinates, and the third column has all its values set to 1.

After estimating $IP(x, y)$, this plane is subtracted from $I(x, y)$. This allows reducing shadows caused by extreme lighting angles. Afterward, histogram equalization is applied for compensating changes in illumination brightness, and differences in camera response curves. Plane subtraction together with histogram equalization was applied in [246] for face detection purposes, and in [247] for obtaining illumination compensation in face recognition. In this last case, histogram equalization is followed by intensity normalization of the face vectors (zero-mean and unit variance).

5.1.2.2 Self-Quotient Image

The self-quotient image (SQI) method is based on the reflectance illumination model of the human vision (Retinex theory model) [248, 249], which assumes: (1) human vision is mostly sensitive to scene reflectance and mostly insensitive to the illumination conditions, and (2) human vision responds to local changes in contrast, rather than to global brightness levels. These two assumptions are related because local contrast is a function of the reflectance. Thus, the reflectance is given by

$$I(x, y)\frac{1}{L(x, y)} = R(x, y) \tag{5.7}$$

with $I(x, y)$ the input stimulus (in this case the input image), and $L(x, y)$ the illumination or perception gain at each point. Illumination can be considered as the low-frequency component of the input stimulus, as has been proved by the spherical harmonics analysis [250]. Then, illumination can be estimated as

$$L(x,y) \approx F(x,y) * I(x,y) \tag{5.8}$$

with $F(x,y)$ a low-pass filter.

From eq.5.7 to eq.5.8 the self-quotient image $Q(x,y)$ is defined as

$$Q(x,y) = \frac{I(x,y)}{F(x,y) * I(x,y)} \approx R(x,y) \tag{5.9}$$

It should be noted that the properties of $Q(x,y)$ are dependant on the kernel size of $F(x,y)$. If too small, then $Q(x,y)$ will approximate one, and the Albedo information will be lost; if too large, there will appear halo effects near edges. In (Wang et al., 2004) this problem is solved by using a multi-scale technique that employs kernels of variable size. Another important improvement is the use of weighting kernels that divide the convolution windows in two different regions, depending on the observed pixel intensities. This technique is supposed to avoid halo effects near the edges. Thus, the final SQI computation procedure is given by (see [251] for details):

(1) Select several smoothing Gaussian kernels $G_1, \cdots, G_n$, calculate the corresponding weighting kernels $W_1, \cdots, W_n$, and the resulting multi-scale self-quotient images as

$$Q_k(x,y) = \frac{I(x,y)}{I(x,y) * \frac{1}{N} W_k G_k}, \qquad k = 1, \cdots, n \tag{5.10}$$

N is a normalization factor for obtaining normalized kernels $W_k G_k$.

(2) Summarize the multi-scale self-quotient images, after applying to each one a non-linear function T (Arctangent or Sigmoid)

$$Q(x,y) = \sum_{k=1}^{n} m_k T(Q_k(x,y)) \tag{5.11}$$

m_k are weighting factors (usually set to one).

5.1.2.3 Local Binary Patterns (LBP)

The local binary pattern (LBP), introduced originally for texture description in [240], has been used in the last years to compensate and normalize illumination in face detection and recognition contexts. LBP, also known as census transform [241], is defined as an ordered set of pixel intensity comparisons, in a local neighborhood $N(x_c, y_c)$, which represents those pixels having a lower intensity than the center pixel $I(x_c, y_c)$. In other words, LBP generates a string of bits representing which pixels in $N(x_c, y_c)$ have a lower intensity than $I(x_c, y_c)$. $I(x_c, y_c)$ is not included in $N(x_c, y_c)$. Formally

speaking let a comparison function $h(I(x_c, y_c), I(x, y))$ be 1 if $I(x_c, y_c) < I(x, y)$ and 0 otherwise. Then, the LBP is defined as

$$LBP(x, y) = \bigcup_{(x',y') \in N(x,y)} h(I(x, y), I(x', y')) \tag{5.12}$$

with $\bigcup$ represents the concatenation operator. Normally, a neighborhood of 3×3 pixels is used, although larger neighborhood can be defined [252]. In the case of a 3×3 neighborhood, $LBP(x, y)$ corresponds to the concatenation of 8 bits. LBP transforms images in illumination invariant feature representations, because all its computations are local, and they depend on the relative values between neighbor pixels (local contrast).

It should be emphasized that in this work the LBP transform is applied just as a preprocessing step, which do not alter the face recognition methods to be employed (e.g., all images are LBP transformed and then processed). In other works (e.g., [252]) spatial histograms of LBP features are computed, and then the recognition is carried out by measuring the histograms similarity using Chi square statistics, histogram intersection or Log-likelihood statistics.

5.1.2.4 Gradient Based Methods

The gradient domain is very important to image processing. The pixel points are not completely independent of each other, there is some relationship between neighboring pixel points. The gradient domain explicitly considers such relationships between neighboring pixel points such that it is able to reveal underlying inherent structure of image data.

Zhang et. al. [239] proposed a novel method to extract illumination insensitive features for face recognition under varying lighting called the Gradientfaces. Theoretical analysis shows Gradientface is an illumination insensitive measure, and robust to different illumination, including uncontrolled, natural lighting. In addition, Gradientface is derived from the image gradient domain such that it can discover underlying inherent structure of face images since the gradient domain explicitly considers the relationships between neighboring pixel points. Therefore, Gradientface has more discriminating power than the illumination insensitive measure extracted from the pixel domain. The author has evaluated the performance of the method using recognition rate performance parameter. Evaluation of the method on CMU PIE and Yale B database shows that Gradientface method is an effective method for face recognition under varying illumination. The proposed method is compared with other three methods named multi-scale

retinex method, self quotient image method and local total variance method. The proposed method performs better than other methods. Though performance of method is evaluated on different dataset, evaluation of methods using other parameters like false acceptance rate, false rejection rate, equal error rate etc. is lacking.

5.2 Our Contributions

Different from most of the existing illumination normalization methods, we consider the normalization procedure as an image style transformation problem. We study different transform task in a unified perspective and develop a new learning framework to enhance the transform performance.

Based on a clustering learning, we found a spatial distribution of human face. Therefore, we build more than one transform models to guarantee the robustness of illumination normalization and the universality for different illumination conditions. The transform model is trained via the sparse representation algorithms.

Specifically, we learn a dictionary pair simultaneously. The pair of dictionaries aims to characterize the two structural domains of the two illuminated case: D_{Norm} for normally illuminated image patches, while D_{Illum} for non-standard illuminated image patches. The sparse representation of a non-standard illuminated patch in terms of D_{Illum} will be directly used to recover the corresponding normal image patch from D_{Norm}. In this work, we try to learn the two over-complete dictionaries by a two-step strategy similar to [71]. To enforce that the image patch pairs have the same sparse representations with respect to D_{Illum} and D_{Norm}, we learn the two dictionaries simultaneously by concatenating them with proper normalization as illustrated in Figure 5.2. The learned compact dictionaries will be applied to CMU PIE database to demonstrate their effectiveness.

5.2.1 A Dictionary Learning framework for Illumination Normalization (DL-IN)

In our work, the illumination normalization problem can be formulated as follows: given an image x which represents non-standard illuminated images, how to recover the associated normally illuminated image y of the same scene? As a patch-based method, our algorithm tries to infer the normalized image patch for non-standard illuminated image patch from the input. For this local model, we have two dictionaries D_{Norm} and D_{Illum}, which are trained to have the same sparse representations for each normal and illuminated image patch pair. We subtract the mean pixel value for each patch, so

that the dictionary represents image textures rather than absolute intensities. In the recovery process, the mean value for each normal image patch is then predicted by its illuminated version.

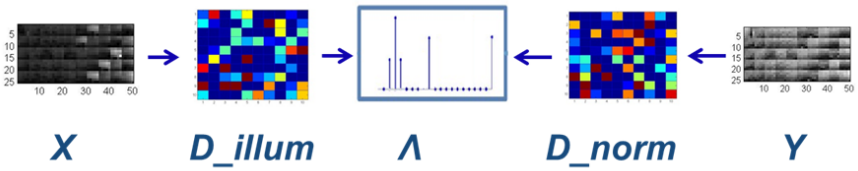

Figure 5.1: **Proposed strategy** - The Coupled Dictionary-Learning

As shown in Figure 5.1, for each input illuminated patch x, we find a sparse representation with respect to D_{Illum}. The corresponding normal patch bases D_{Norm} will be combined according to these coefficients to generate the output normalized patch y. Denote by X and Y the training datasets formed by the image patch pairs of illuminated face image and normal face image. We propose to minimize the energy function below to find the desired coupled dictionaries:

$$\begin{aligned} \min_{D,\Lambda} E_{data}(D_{Illum}, X) + E_{data}(D_{Norm}, Y) \\ +\lambda E_{reg}(\Lambda, D_{Illum}, D_{Norm}) \end{aligned} \tag{5.13}$$

where $E_{data}(a, b)$ is the data fidelity term to represent data description error, E_{reg} is the regularization term to regularize the coding coefficients. In the proposed model, the two dictionaries (D_{Illum} and D_{Norm}) will be jointly optimized. Then eq.5.13 can be turned in to the following dictionary learning and ridge regression problem:

$$\begin{aligned} \min_{D,\Lambda} \|X - D_{Illum}\Lambda\|_2^2 + \|Y - D_{Norm}\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \\ \text{s.t.} \quad \|d_{x,i}\|_{l_2} \leq 1, \quad \|d_{y,i}\|_{l_2} \leq 1, \quad \forall i \end{aligned} \tag{5.14}$$

where λ is the regularization parameter to balance the terms in the objective function, and $d_{x,i}, d_{y,i}$ are the atoms of D_{Illum} and D_{Norm}, respectively. The objective function in Eq. 5.14 is not jointly convex to D_{Illum}, D_{Norm}. However, it is convex with regard to each of them if others are fixed. Therefore, we can design an iterative algorithm to alternatively optimize the variables.

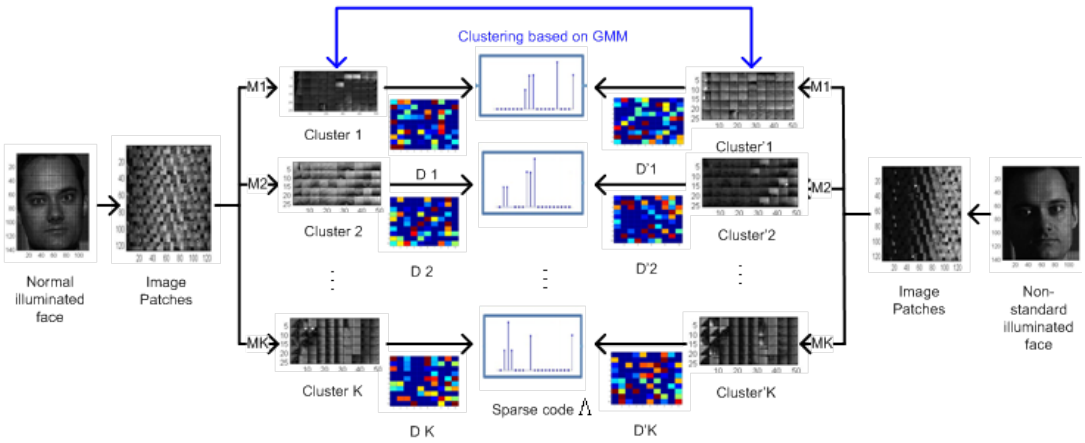

Figure 5.2: **Proposed strategy** - Dictionary Learning framework for Illumination Normalization (DL-IN)

5.2.1.1 Clustering based on the coupled image patch of normal faces and illuminated faces

Consider the complex structures in images of non-standard illumination, learning only one pair of dictionaries is not enough to cover all variations of illumination normalization. For example, in the normalized face synthesis the appearance difference between illuminated face and normalized face may vary significantly in different facial regions. Therefore multi-model should be learned to enhance the robustness. Intuitively, pre-clustering could be conducted to separate training data into several groups so that the linear mapping in each group can be more stably learned. And therefore, to normalize an illuminated face image, choosing the appropriate transform model is critical.

As illustrated in Figure 5.2, the multi-model based dictionary learning is initialized by pre-clustering the face image patch pairs. Motivated by the successful application of Gaussian Mixture Model (GMM) [253] in many practical problems, we apply it for the pre-clustering procedure. Consider that we have K models, denoted as $M_1, M_2, \cdots, M_K$ then the probability of a patch-pair (x_i, y_i) conditioning on the $k-th$ model is

$$p(x_i, y_i|M_k) = p(x_i|m_{xk}, \Sigma_{xk})p(y_i|m_{yk}, \Sigma_{yk}) \tag{5.15}$$

Here m_{xk}, Σ_{xk} and m_{yk}, Σ_{yk} are the mean vectors and covariance matrices of the patches belonging to M_k in respectively.

Optimization by EM Algorithm: With the definition of coupled conditional

probability, the GMM can be learned by Expectation-Maximization algorithm similar to that in GMM [253]. The procedure is described as follows:

Algorithm 1: Training process of GMM

1: Initialize GMM by Randomly Clustering

(a): Randomly select K pairs of patches as cluster centers, denoted as $m_{x1}^{(0)}, \cdots, m_{xK}^{(0)}$, and $m_{y1}^{(0)}, \cdots, m_{yK}^{(0)}$ which are also the initial estimation of mean vectors for GMM.

(b): For each pair of samples (x_i, y_i), categorize it to the cluster where the cluster center is closest. The distance is simply defined as $d_{ik} = \left\| x_i - m_{xk}^{(0)} \right\|^2 + \left\| y_i - m_{yk}^{(0)} \right\|^2$.

(c): Compute the covariance matrices in clusters $\Sigma_{x1}^{(0)}, \cdots, \Sigma_{xK}^{(0)}$ and $\Sigma_{y1}^{(0)}, \cdots, \Sigma_{yK}^{(0)}$ as initial estimation of covariance matrices.

(d): Initialize the prior probability for all models to be the same: $P^{(0)}(M_1) = \cdots = P^{(0)}(M_K) = \frac{1}{K}$.

2: Update the GMM by iterating the following steps:

(a): Compute the probability of every training sample-pair belonging to the $k-th$ model as

$$\omega_{ik} = \frac{p^{(t-1)}(M_k) p(x_i, y_i | M_k)}{\sum_{j=1}^{K} p^{(t-1)}(M_j) p(x_i, y_i | M_j)}$$

The calculation of conditional probability follows Eq.5.15 using the mean vectors and covariance matrices computed in the $(t-1)th$ step.

(b): Update the priori of models as

$$p(M_k) = \frac{1}{n} \sum_{i=1}^{n} \omega_{ik}$$

(c): Update the mean vectors and covariance matrices as follows:

$$m_{xK}^{(t)} = \frac{1}{np(M_k)} \sum_{i=1}^{n} \omega_{ik} x_i$$

$$m_{yK}^{(t)} = \frac{1}{np(M_k)} \sum_{i=1}^{n} \omega_{ik} y_i$$

$$\Sigma_{xK}^{(t)} = \frac{1}{np(M_k)} \sum_{i=1}^{n} \omega_{ik} (x_i - m_{xk})(x_i - m_{xk})^T$$

$$\Sigma_{yK}^{(t)} = \frac{1}{np(M_k)} \sum_{i=1}^{n} \omega_{ik} (y_i - m_{yk})(y_i - m_{yk})^T$$

After the GMM is trained, the dictionaries for illumination normalization are learned for every model, denoted as $D_{Illum}^1, \cdots, D_{Illum}^K$ and $D_{Norm}^1, \cdots, D_{Norm}^K$. So for a new patch x, then y can be computed as

$$y = \sum_{k=1}^{K} P(M_k|x)(D_{Norm}^k \Lambda^k) \tag{5.16}$$

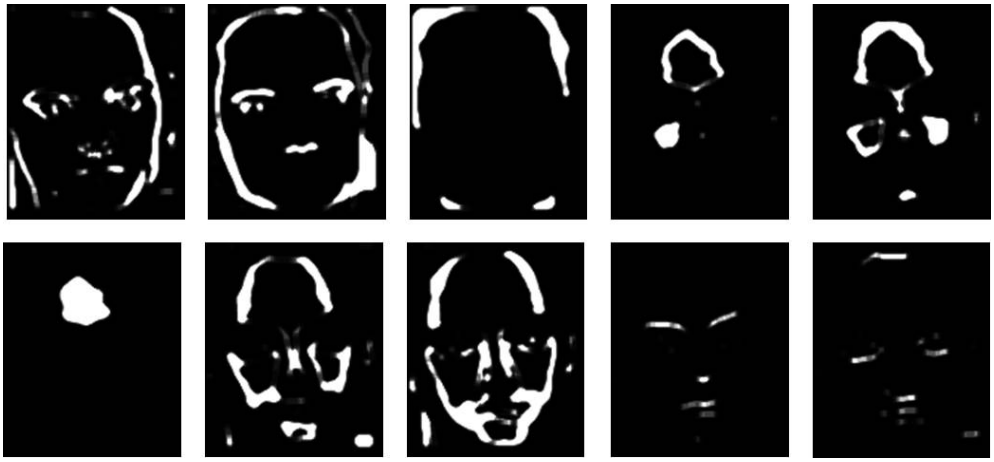

Figure 5.3: **Examples of cluster distribution** - Each sub-figure shows the distribution of a cluster

As illustrated in Figure 5.2, according to the GMM based pre-clustering, we learn the distribution of the patches consisted in each cluster, the structure of the face images shows a distinctive spatial distribution as demonstrated in Figure 5.3. One can notice that the patches in different clusters concentrate at different spatial locations.

5.2.1.2 Dictionary Learning

To solve the dictionary learning problem in Eq. 5.14, we separate the objective function into 2 sub-problems: sparse coding for training samples, and dictionary updating. Each training procedure of the multi-models are based on the coupled patch clustering results.

Given the sampled training image patch pairs $P = X, Y$, where $X = \{x_1, x_2, \cdots, x_n\}$ are the set of sampled non-standard illuminated image patches and $Y = \{y_1, y_2, \cdots, y_n\}$ are the corresponding standard illuminated image patches, our goal is to learn dictionaries for illuminated and normal image patches, so that the sparse representations of the two different case are the same. This is a difficult problem, the individual sparse coding problems in the illuminated and normal face patch spaces are:

$$\underset{D_{Illum},\Lambda}{min} \|X - D_{Illum}\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \tag{5.17}$$

$$\min_{D_{Norm},\Lambda} \|Y - D_{Norm}\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \tag{5.18}$$

respectively. We combine these objectives, forcing the illuminated and normal face patch to share the same sparse codes, instead writing

$$\min_{D_c,\Lambda} + \|X_c - D_c\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \tag{5.19}$$

where

$$X_c = \begin{bmatrix} X^T & Y^T \end{bmatrix}^T, \qquad D_c = \begin{bmatrix} D_{Illum}^T & D_{Norm}^T \end{bmatrix}^T \tag{5.20}$$

Eq. 5.19 can be optimized by a two-step strategy. When dictionary pair D_c is fixed. 5.19 is a Lasso problem. Many L1-optimization algorithms can solve it effectively, such as FISTA [254], LARS [255], etc. In this paper, we choose LARS [255] as the L1-optimization method for its efficiency and stability.

Then, with Λ fixed, dictionary pair D_c can be updated as:

$$\min_{D_c} + \|X_c - D_c\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \tag{5.21}$$

Thus we can use the same learning strategy in the single dictionary case for training the two dictionaries for our illumination normalization purpose.

With DL-IN, we can learn the dictionary pair D_{Illum} and D_{Norm} on which the sparse coding coefficients of two illumination conditions have stable bidirectional linear transformations. The stability is further enhanced by the pre-clustering procedure in section 5.2.1.1 and by exploiting the image nonlocal redundancy of patches as presented in section 5.2.2.1.

5.2.1.3 Summary Algorithm

In this section, we resume the joint dictionary learning procedure for the illumination normalization problem. The training algorithm is summarized in Algorithm 2:

5.2.2 Normalized face image synthesis

5.2.2.1 Model Selection and image initialization

With the spatial distribution observation as shown in section 5.2.1.1, we can have an empirical estimation of the spatial distribution of each cluster. For a new illuminated image patch x_i, the initial model selection can then be competed as:

Algorithm 2: Joint dictionary learning for illumination normalization

Input: Training datasets X and Y of illuminated face image patches and normal face image patches. Each corresponding pair indicates the same individual.

1: Cluster the training dataset by the GMM EM optimization as presented in section 5.2.1.1.

2: For each cluster k, initial dictionary pair D^k_{Illum} and D^k_{Norm}.

3: For each iteration until convergence:

(a) Fix dictionaries D^k_{Illum} and D^k_{Norm}, update sparse code Λ^k by optimize eq. 5.19 using LARS algorithm.

(b) Fix sparse code Λ^k, update D^k_{Illum} and D^k_{Norm} in eq. 5.21.

Output: D^k_{Illum}, $D^k_{Norm}$and Λ^k.

$$P(M_k|Loc_i, x_i) = \frac{P(Loc_i, x_i|M_k)}{\sum_{j=1}^{K} P(Loc_i, x_i|M_j)P(M_j)} \tag{5.22}$$

where $Loc_i = (row_i, col_i)^T$ are coordinates of patches x_i in spatial domain and distribution $P(Loc_i, x_i|M_k)$ is the prior probability from empirical observation on training data, which is culculated by: $P(Loc_i, x_i|M_k) = P(Loc_i|M_k)N(x_i|m_{xk}, \Sigma_{uk})$. Where $P(Loc_i|M_k)$ can be easily computed according to the clustering results in section 5.2.1.1.

5.2.2.2 Illumination normalization based on the jointly learned Dictionaries

According to the learning the dictionaries D_{Illum} and D_{Norm}, and the model selection method presented in the previous section 5.2.2.1, for a given illuminated image x, we can find a sparse representation with respect to D_{Illum}. The corresponding normal face image patch basis D_{Norm} will be combined according to these coefficients to generate the output normalized image patch y. This synthesis problem is solved by optimize:

$$\underset{\alpha^k, y^k_i}{min} \left\|x^k_i - D^k_{Illum}\alpha^k_i\right\|^2_2 + \left\|y^k_i - D^k_{Norm}\alpha^k_i\right\|^2_2 + \lambda \left\|\alpha^k_i\right\|_1 \tag{5.23}$$

where x^k_i is a patch of x and y^k_i is the corresponding patch in the intermediate estimate of y to be synthesized. D^k_{Illum}, D^k_{Norm} and α^k_i are the learned dictionaries and sparse code of the current chosen cluster k. Eq. 5.23 can be solved by alternatively updating α^k_i and y^k_i. Finally, each patch of y can be reconstructed as:

$$\hat{y}_i^k = D_{Norm}^k \hat{\alpha}_i^k \tag{5.24}$$

After all the patches are estimated, the estimation of the desired image y can then be obtained.

In our synthesis method, an initial estimation of y_i^k is needed. We can first code x_i^k on D_{Illum}^k for coding vector α_i^k, and then initialize y_i^k as:

$$y_i^k = D_{Norm}^k \alpha_i^k \tag{5.25}$$

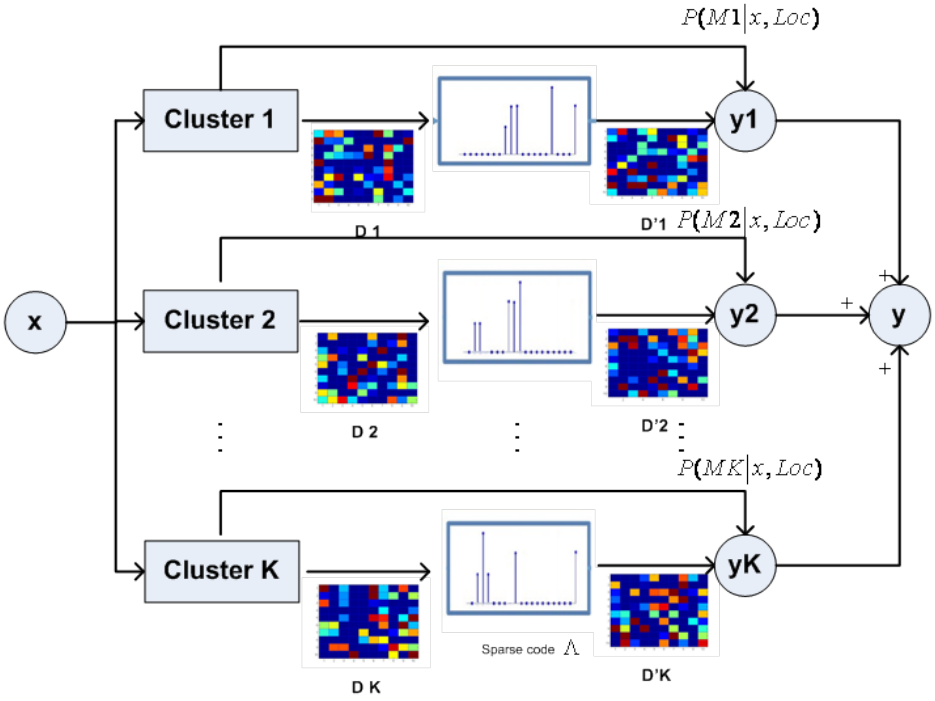

Figure 5.4: **Synthesis process** - Illustration of the GMM inference process

Then normalized image patch y_i can be computed by eq. 5.26, as illustrated in Figure 5.4:

$$y_i = \sum_{k=1}^{K} P(M_k|Loc_i, x_i) y_i^k \tag{5.26}$$

5.2.2.3 Summary Algorithm

In this section, we resume the normalized face image synthesis procedure. The synthesis algorithm is summarized in Algorithm 3:

Algorithm 3: Normalized face image synthesis

Input: The illuminated image x, trained dictionaries D_{Illum} and D_{Norm}.

1: Compute the posteriori probability by eq. 5.22 for each cluster.

2: Initialize y^k as eq. 5.25.

3: For each cluster, each iteration until convergence: Optimize eq. 5.24 by alternatively updating α_i^k and y_i^k.

4: Combine the synthesized y_i^k of each cluster according to Figure 5.4:

$$y_i = \sum_{k=1}^{K} P(M_k|Loc_i, x_i) y_i^k$$

Output: Normalized image y.

5.3 Experimental Results

In this section, experiments are conducted to justify the proposed illumination normalization framework. The performances of algorithms are evaluated in the aspects of visual results of illuminated normalized images and the appearance difference between the ground truth image and the normalized image.

CMU PIE databases [125] are selected for evaluation. The CMU PIE consists of 68 individuals. For training and testing the proposed algorithm, the frontal face images under 20 different illumination conditions with background lighting off are selected. In our experiments, all images are simply aligned and each image is cropped and resized to 115×140.

5.3.1 Evaluation criterion

To assess the performance of DL-IN algorithm for normalizing both the known and unknown illumination conditions, we separate the 20 illumination conditions into two groups as illustrated in Figure 5.5 and Figure 5.6, which represent the illuminated images for dictionary learning and the illuminated images for performance evaluation.

In our experiments, 500 thousand training patch pairs are extracted from a subset of CMU database which is built by 15 individuals and 10 illumination conditions. The patch size is 5×5. Pre-clustering is conducted and cluster number for each illumination condition is set to be 32. In Eq. 5.18, the choice of λ is empirically set by experience. We always set $\lambda = 0.1$ in our experiments, which generally yields satisfactory results. The number of atoms in the learned dictionary for each cluster is set as 512.

In addition, we propose to employ an objective measurement for the visual quality of a normalized image as follows:

Figure 5.5: **Database examples** - Illumination conditions used for training Dictionaries (Illumination number 05,08,10,12,14,15,17,18,19,20)

Figure 5.6: **Database examples** - Illumination conditions used for evaluating the proposed framework (Illumination number 02,03,04,06,07,09,13,16,21)

$$Diff = \frac{\|I_{standard} - I_{normalized}\|_2}{d} \tag{5.27}$$

where $I_{standard}$ and $I_{normalized}$ denote the corresponding image under normal illumination condition and the normalized image of the same subject respectively. And d is the number of pixels in an image. The images of illumination number 11 are treated as the standard illumination $I_{standard}$. The measurement results are reported in Table 5.1-5.3. The smaller the value $Diff$ is, the better the algorithm performs.

5.3.2 Reconstruction of normalized images

Figure 5.7: **Standard image** - Illumination number 11

In the reconstruction stage, we define 3 different conditions. Test 1: The individuals and illumination conditions are both contained by the training set; Test 2: The individuals are unknown, but illumination condition is contained in the dictionary learning procedure; Test 3: Neither the individual, nor the illumination is included by the trained dictionaries. The illumination normalization results for the 3 different cases are show in Figure 5.8, 5.9, 5.10 respectively.

Illumination number	05	08	10	12	14	15	17	18	19	20	AVG
Diff	0.179	0.163	0.175	0.195	0.230	0.256	0.190	0.1620	0.166	0.176	0.189

Table 5.1: Average $Diff$ for each illumination condition of Test 1

In the first row of Figure 5.8, one can notice that although the illumination images are well normalized, there exists a obvious block phenomenon. This is caused by the local patch combination and also the clustering step in the reconstruction procedure. To eliminate the influence of the blocks, we apply the local pixel grouping denoising method [256], which is a robust denoising and smoothen method. The denoising normalization

Figure 5.8: **Illumination normalization results** - Evaluation on Test 1. (a): Original normalized face images. (b): Denoising normalized face images. (c): Target normal illuminated face images. (d): Original non-standard illuminated face images.

results are shown in row (b) of Figure 5.8. For the following figures, we present only the denoising normalization results.

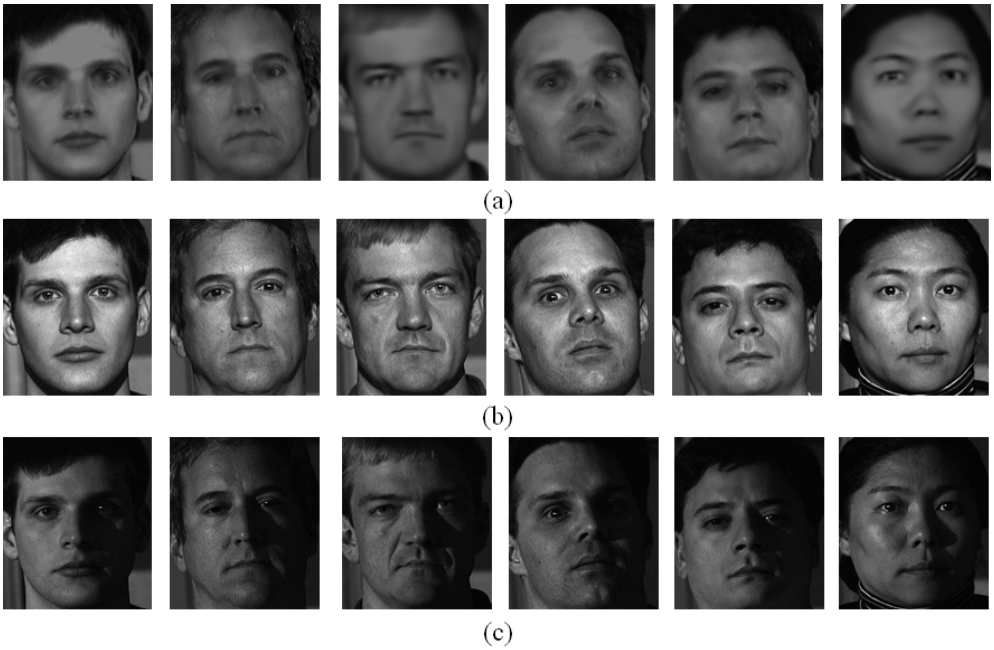

Figure 5.9: **Illumination normalization results** - Evaluation on Test 2. (a): Denoising normalized face images. (b): Target normal illuminated face images. (c): Original non-standard illuminated face images.

Illumination number	05	08	10	12	14	15	17	18	19	20	AVG
Diff	0.194	0.181	0.195	0.218	0.256	0.272	0.217	0.188	0.192	0.195	0.211

Table 5.2: Average $Diff$ for each illumination condition of Test 2

Figure 5.9 shows the illumination compensating results for the unknown individuals, but the illumination case is contained by the dictionary training process. The performance is as good as the ones shown in Figure 5.8, owning to the use of the clustering based local information. But the average $Diff$ value is larger than the average $Diff$ value of Test 1, which denotes that the normalization results are actually effected by the unknown individuals.

Figure 5.10 shows the normalization results for the images which are totally excluded by the dictionary learning. In this experiment, the shadows of the input illuminated face images are not completely eliminated. Because a new sort of illuminated images lead to

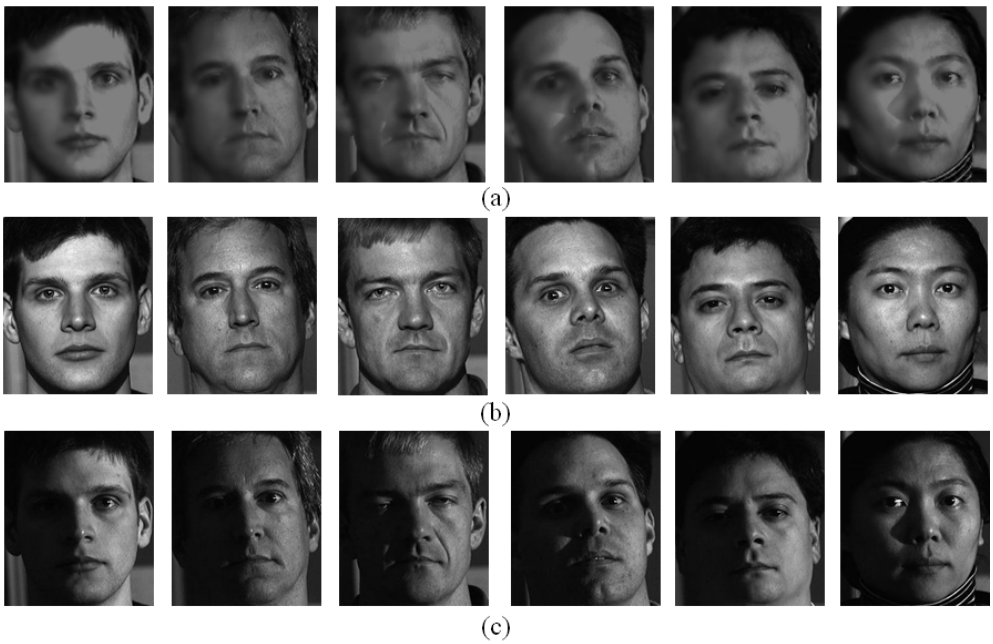

Figure 5.10: **Illumination normalization results** - Evaluation on Test 3. (a): Denoising normalized face images. (b): Target normal illuminated face images. (c): Original non-standard illuminated face images.

Illumination number	02	03	04	06	07	09	13	16	21	AVG
Diff	0.312	0.310	0.306	0.260	0.259	0.270	0.288	0.344	0.266	0.291

Table 5.3: Average $Diff$ for each illumination condition of Test 3

novel patch appearances. The patches are clustered and transformed according to the learned dictionaries which are built by the patches with the most similar appearances. So it is possible to synthesize a generally normalized face, but with the residual shadow trail.

5.3.3 Face recognition application

In this section, the face recognition results based on the proposed normalization algorithm and several well-known illumination invariant features are reported. For each individual in the CMU-PIE database, the 20 illuminated frontal face samples are randomly separated into two parts, 10 images as reference (gallery) faces and 10 images as query (probe) images.

All reference and query images were pre-processed by the same illumination-normalization algorithm and then used for recognition. Since our experiments focus on the effects of illumination normalization, the original pixel values of the normalized images were used as facial features for recognition, without further feature extraction or dimension reduction. For classification, the Sparse Representation Classifier (SRC) [222] was selected.

We compared the proposed algorithm, namely DL-NI, with Discrete Cosine Transform (DCT) [257] and Discrete Wavelet Transform (DWT) [258]. Also, with the state-of-the-art illumination invariant feature extraction algorithms, Self Quotient Image (SQI) [238], and Local Binary pattern (LBP) [259] were included for comparative purposes.

The recognition rates of face identification by different methods are tabulated in Table 5.4. As shown, the algorithm based on our method, DL-NI, produced higher recognition rates than the related algorithms.

Illumination invariant features	LBP	SQI	DCT	DWT	DL-NI
Face recognition rates	75.4%	94.0%	88.9%	87.3%	98.6%

Table 5.4: Comparisons between different methods for face recognition on CMU PIE database

5.4 Conclusion

In this chapter, we proposed a novel Dictionary Learning framework for Illumination Normalization (DL-IN). DL-IN based on sparse representation in terms of coupled dictionaries jointly optimized from normal illuminated and non-standard illuminated face image patch pairs. We further utilize a GMM model to enhance the framework's capability of modeling data under complex distribution by adapting each model to a part of samples and fuse them together. Experimental results demonstrate the effectiveness of the sparsity as a prior for patch-based illumination normalization for face images. In the future work, we will adapt the clustering and patch based jointly dictionary learning algorithm for more types of image synthesis tasks and extend it to face recognition tasks.

6

Conclusion

6.1 Summary of the Contributions

In this dissertation, we have presented our research study on face analysis. Our main objective was to construct several effective algorithms which improve the face analysis works under the effect of different head pose and varying lighting conditions. Indeed, the three diverse kinds of frameworks have respectively promising applications (robust face deformable model, face pose estimation, face illumination normalization). This main objective was accomplished through the non-linear dimension reduction and sparse representation frameworks. The experimental results have demonstrated the effectiveness of the proposed algorithms.

The contributions of this manuscript can be summarized as follows.

- **Introducing Kernel-Similarity Principal Component Analysis to Active Appearance Models**

  In order to construct a robust deformable model, which is able to adapt multiple face appearance variations caused by the variable lighting conditions and head rotations, we introduce two non-linear embedding models to Active Appearance Models (AAMs). One is the Probabilistic PCA, the other is the Kernel-Similarity PCA. Both of the proposed embedding models are more adaptive to the statistic distribution of the observed data than the standard PCA model.

  Experimental Results have shown that the model build by the proposed non-linear embedding methods are less sensitive to the illumination variations. The fitting performances indicate a powerful improvement on the illumination case especially on KS-PCA based AAMs. Probabilistic PCA model also brought better

performance than classical AAMs but not as obvious as KS-PCA model. With the novel methods, the fitting procedure can accurately synthesize faces semi-bright-semi-dark affected by the Illumination. Meanwhile, conditions with a variety of poses also benefit from the proposed algorithms; the ability of synthesizing faces with shape variations from a wide range of face poses has been improved. The problem of the synthesis of the complete profile faces is still waiting to be solved.

- **Robust face pose classification framework based on a joint optimization of manifold learning and sparse representation**

  Incremental PCA is an adaptive embedded model which is able to generate new manifold space without recalculating on all the observations. In this thesis, we have jointed the Incremental PCA and the sparse representation classification to train an over-complete dictionary. With the purpose of avoiding the intra-class redundancy and meanwhile keeping enough variables for the over-complete dictionary. The joint optimization strategy shows robustness on complex illumination conditions and noisy images in terms of the classification accuracy.

- **Robust face pose classification framework based on an unified criterion of dictionary learning and classification**

  We have also proposed a reasonable dictionary learning framework that employed the recently proposed K-SVD over-complete dictionary designing method. In this study, the over-complete dictionary is searched with the optimization goal of coinciding with the classification criterion. And due to this unification of the dictionary learning process and the classifying rule, the proposed framework is quite robust against various low-quality face images and over-performed than the state-of-arts methods.

- **A Dictionary Learning framework for Illumination Normalization**

  We proposed a novel Dictionary Learning framework for Illumination Normalization (DLIN). DLIN based on sparse representation in terms of coupled dictionaries jointly optimized from normal illuminated and non-standard illuminated face image patch pairs. We further utilize a GMM model to enhance the framework's capability of modeling data under complex distribution by adapting each model to a part of samples and fuse them together. Experimental results demonstrate the effectiveness of the sparsity as a prior for patch-based illumination normalization for face images. In the future work, we will adapt the clustering and patch based

jointly dictionary learning algorithm for more types of image synthesis tasks and extend it to face recognition tasks.

In all, an important characteristic of this dissertation work is that we have devised effective face analysis studies by analyzing the data distribution and combining useful ingredients from several different research domains. The ingredients include KS-PCA embedding model and Incremental PCA model from manifold learning, L_1 norm Minimization sparse solution and K-SVD over-complete dictionary design from sparse representation research, Gaussian Mixture Model of statistical analysis. This interdisciplinary research method may also be considered as a contribution.

6.2 Perspective

Some research perspectives appear at the end of this thesis.

- **Improvement of the KS-PCA based Active Appearance Models**

  For construct a more robust 2D AAMs, which works for both variety of illumination conditions and head poses, there exist two possible solutions. One is to combine the existed KS-PCA based AAMs and the proposed pose classification algorithms for building multi-model deformable system. Using multi-models is not a novel idea, Cootes et al. showed that by using five models it is sufficient to deal with faces where the head pose vary by 180 degrees (from left profile to right profile) [107]. But this kind of combination can be a practical solution for such realistic problems. The other improvement is to extend this KS-PCA method to the 3D AAMs, since the appearance of different face poses highly depends on 3D head structure.

- **Improvement of the face pose classification frameworks**

  In this study, an intriguing question for future work is whether this framework can be useful for object recognition. The usefulness of sparsity in detection has been noticed in the work in [260] and more recently explored in [261]. We believe that the full potential of sparsity in robust object detection and recognition together is yet to be uncovered. From a practical standpoint, it would also be useful to extend the algorithm to less constrained conditions.

- **Improvement of the Dictionary learning for Illumination Normalization**

  In the future work, we will adapt the clustering and patch based jointly dictionary learning algorithm for more types of image synthesis tasks and extend it to face recognition tasks. Of special interest, we highlight the need for a multi-scale structured dictionary learning paradigm, as well as methods to use such dictionaries in variety applications.

Many challenges are still open in embedding model and dictionary learning. Understanding the underlying causes of signals or the relevant information in observation becomes more challenging when the training samples are imperfect. In many applications, the training samples are noisy, distorted by the sensing process, or simply incomplete like in the case of occlusions in multi-view imaging. In all these situations, dictionary learning still faces critical research questions. Similarity, signal analysis may require more complex models for efficient classification. One can build dictionaries to be used in the definition of manifold models or graph based representations that could potentially handle transformation invariant classification problems. In general, dictionaries offer a very flexible and powerful way to represent relevant information in high-dimensional signals. However, the proper modeling of the complex underlying causes of observations poses many exciting questions about the proper construction of these dictionaries.

6.3 Relevant Publications

International Conferences

- Zhang Y., Idrissi K. and Garcia C., Incremental Principal Component Analysis-Based Sparse Representation for Face Pose Classification, Advanced Concepts for Intelligent Vision Systems. Springer International Publishing, 2013: 620-631.
- Zhang Y., Idrissi K. and Garcia C., Dictionary-Learning Sparse Representation framework for Pose Classification, IEEE International Workshop on Machine Learning for Signal Processing. September 22-25, Southampton, United Kindom 2013:
- Zhang Y., Benhamza Y., Idrissi K. and Garcia C., Kernel similarity based AAMs for face recognition, Advanced Concepts for Intelligent Vision Systems. Springer Berlin Heidelberg, 2012: 395-406.

Local Conference

- Zhang Y., Benhamza Y., Idrissi K. and Garcia C., Probabilistic Active Appearance Models. Coresa 2012, Lille.

7

Résumé en Français

7.1 Introduction

Les techniques d'analyse du visage nécessitent généralement une représentation pertinente des images, notamment en passant par des techniques de réduction de la dimension, intégrées dans des schémas plus globaux, et qui visent à capturer les caractéristiques discriminantes des signaux.

Dans cette thèse, nous fournissons d'abord une vue générale sur l'état de l'art de ces modèles, puis nous appliquons une nouvelle méthode intégrant une approche non-linéaire, Kernel Similarity Principle Component Analysis (KS-PCA), aux Modèles Actifs d'Apparence (AAMs), pour modéliser l'apparence d'un visage dans des conditions d'illumination variables. L'algorithme proposé améliore notablement les résultats obtenus par l'utilisation d'une transformation PCA linéaire traditionnelle, que ce soit pour la capture des caractéristiques saillantes, produites par les variations d'illumination, ou pour la reconstruction des visages.

Nous considérons aussi le problème de la classification automatiquement des poses des visages pour différentes vues et différentes illumination, avec occlusion et bruit. Basé sur les méthodes des représentations parcimonieuses, nous proposons deux cadres d'apprentissage de dictionnaire pour ce problème.

Une première méthode vise la classification de poses à l'aide d'une représentation parcimonieuse active (Active Sparse Representation ASRC). En fait, un dictionnaire est construit grâce à un modèle linéaire, l'Incremental Principle Component Analysis (Incremental PCA), qui a tendance à diminuer la redondance intra-classe qui peut affecter la performance de la classification, tout en gardant la redondance inter-classes, qui elle, est critique pour les représentations parcimonieuses.

La seconde approche proposée est un modèle des représentations parcimonieuses basé sur le Dictionary-Learning Sparse Representation (DLSR), qui cherche à intégrer la prise en compte du critère de la classification dans le processus d'apprentissage du dictionnaire. Nous faisons appel dans cette partie à l'algorithme K-SVD.

Nos résultats expérimentaux montrent la performance de ces deux méthodes d'apprentissage de dictionnaire.

Enfin, nous proposons un nouveau schéma pour l'apprentissage de dictionnaire adapté à la normalisation de l'illumination (Dictionary Learning for Illumination Normalization: DLIN). L'approche ici consiste à construire une paire de dictionnaires avec une représentation parcimonieuse. Ces dictionnaires sont construits respectivement à partir de visages illuminées normalement et irrégulièrement, puis optimisés de manière conjointe. Nous utilisons un modèle de mixture de Gaussiennes (GMM) pour augmenter la capacité à modéliser des données avec des distributions plus complexes. Les résultats expérimentaux démontrent l'efficacité de notre approche pour la normalisation d'illumination.

7.2 Intégration de modèles non-linéaire aux Modèles Actifs d'Apparence

L'analyse faciale consiste en un schéma global incluant l'alignement du visage, l'estimation de la pose et enfin l'extraction des caractéristiques et de l'expression de ce visage, généralement inconnu. Pour la détermination de la pose dans des conditions d'illumination variables, l'un des paramètres les plus problématiques est la non linéarité causée par les changements irréguliers de l'apparence pour les différentes conditions d'illumination. La détection s'avère encore plus difficile à réaliser si le visage est inconnu du système. Notre travail de thèse s'est intéressé à cette problématique et a consisté à proposer une solution pour la détection de visages inconnus lorsque les conditions d'illumination varient.

De part leur nature, les visages humains sont des objets non-rigides. Le problème de cette flexibilité est pris en compte dans les Modèles Actifs d'Apparence (AAMs) [84] qui sont remarquablement efficaces lorsqu'il s'agit d'extraire des caractéristiques faciales et plus généralement lorsqu'il faut aligner un visage (opération qui consiste à localiser plusieurs dizaines de points autour des yeux, du nez, de la bouche et des sourcils). Notre système d'analyse de visage est basé sur l'intégration de la non-linéarité dans les modèles actifs d'apparence.

7.2.1 Propriétés et limitations de la PCA

L'Analyse en Composantes Principales (PCA) [5] trouve des applications dans de très nombreux domaines parce qu'elle révèle des structures sous-jacentes simples dans des ensembles des données complexes, par l'utilisation de solutions analytiques de l'algèbre linéaire [120]. Généralement, la motivation primaire derrière cette méthode est de décorréler l'ensemble des données, ou en d'autres termes, d'enlever des dépendances de deuxième l'ordre dans les données. Cependant, comme il sera présenté dans la section 7.2.2 suivante, il existe des dépendances d'ordre plus élevé dans les données d'illumination. Par conséquent, enlever des dépendances du deuxième ordre sera insuffisant pour détecter toutes les structures des données.

7.2.2 Analyse statistique en fonction de l'illumination

Pour bien comprendre la différence entre un ensemble de visages éclairés uniformément ou non, une analyse statistique s'avère significative. Considérons un ensemble d'images de visages, comme indiqué dans la figure 3.3, où la pose des visages est fixe (frontale) et où les conditions d'illumination varient dynamiquement.

Au niveau des descripteurs, les contours des visages sont décrits par des vecteurs s_i contenant les coordonnées des points des contours, et les textures sont représentées par des vecteurs g_i, qui contiennent les intensités des pixels contenus dans les triangles de Delaunay.

Pour analyser la corrélation entre la forme et la texture de tous les visages, les distances Euclidiennes $d(a, b)$ entre des éléments contenus dans chaque vecteur sont calculées tant pour la forme que pour la texture :

$$d(s_i, s_j) = \sqrt{\sum_{i=1}^{D_s} (s_i - s_j)^2} \tag{7.1}$$

$$d(g_i, g_j) = \sqrt{\sum_{i=1}^{D_g} (g_i - g_j)^2} \tag{7.2}$$

Les histogrammes de ces distances sont ensuite construits comme cela est illustré dans la figure 7.2 et 3.5, puis les paramètres caractérisant ces distributions, supposées Gaussiennes, sont calculés.

Figure 7.1: **Les exemples des Images du visage frontaux de la base de données de CMU PIE** - dans 20 conditions d'illumination

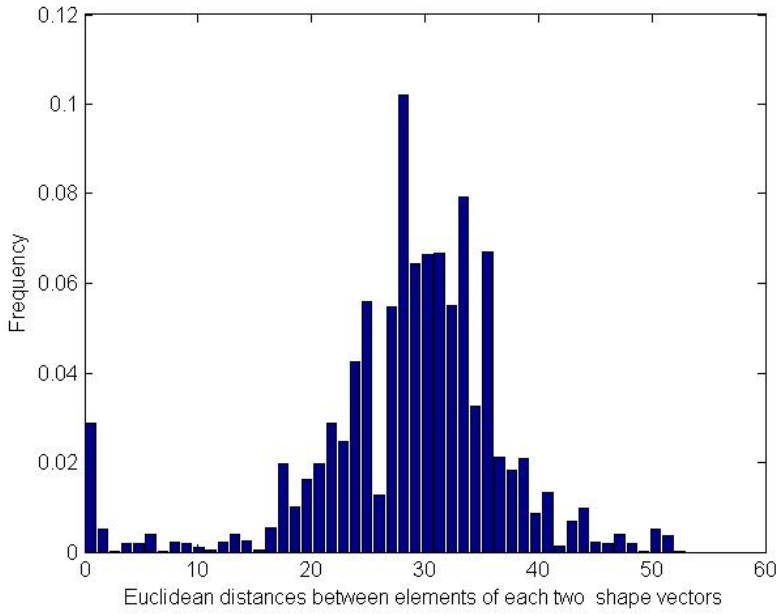

Figure 7.2: **Le Histogramme** - Histogramme des distances euclidiennes entre les vecteur de la forme

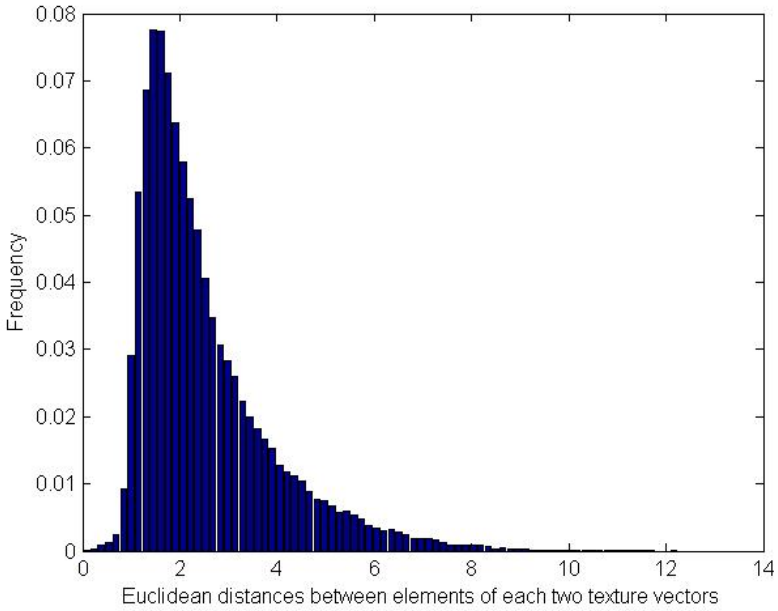

Figure 7.3: **Le Histogramme** - Histogramme des distances euclidiennes entre les vecteur de la texture

7.2.3 Analyse en Composantes Principales Probabiliste (PPCA)

L'objectif général est de chercher à relier un vecteur d'observation t de dimension d à un vecteur de variables latentes x de dimension N, avec $N << d$. Le modèle le plus commun est l'analyse factorielle où la relation reliant les deux espaces est linéaire et de la forme:

$$t = Wx + \mu + \epsilon \tag{7.3}$$

En supposant que les variables latentes sont aussi gaussiennes et conventionnellement définies par $x \sim N(0, I)$, l'utilisation d'un modèle du bruit gaussien pour ϵ nous permet d'écrire:

$$P(t|x) \sim N(Wx + \mu, \delta^2 I) \tag{7.4}$$

la distribution marginale pour les données observées t est elle-même gaussienne, et est alors aisément obtenue en intégrant les variables latentes:

$$P(t) \sim N(\mu, C) \tag{7.5}$$

où la covariance du modèle d'observation est donné par $C = WW^T + \delta^2 I$. Le log-likelyhood correspondant est alors exprimé par :

$$L = -\frac{N}{2}\left\{dln(2\pi) + ln\,|C| + tr(C^{-1}S)\right\} \tag{7.6}$$

où une fonction exponentielle de la distance est adoptée en tant que probabilité locale, ce qui nous donne:

$$S = \sum_{n=1}^{d}\sum_{m=1}^{d}(t_n - \mu_{t_n})(t_m - \mu_{t_m})P(t_n)P(t_m) \tag{7.7}$$

$$P(t_n) = \frac{1}{2\pi\delta}e^{-\frac{|t_n - \bar{t}_n|}{2\delta^2}} \tag{7.8}$$

On donne pour l'estimateur de probabilité maximale de $\bar{t}_n$ comme le moyen des données, auquel cas S est la matrice de la covariance des observations t_n.

7.2.4 Analyse en Composante Principale à Noyau

Dans la majorité des algorithmes de réduction de la dimension, l'objectif principal est de préserver dans l'espace de dimension réduite, la structure qui existe entre les objets dans l'espace d'origine. Ceci peut être réalisé en minimisant une fonction de coût, qui estime l'incohérence entre des paires de mesures (similitude/dissimilitude), effectuées dans l'espace d'origine et dans l'espace de projection. Nous proposons une nouvelle méthode de réduction de la dimension basé sur cette idée, appelée Kernel Similarity Principle Component Analysis (KS-PCA).

Clairement, on ne peut pas affirmer que la PCA linéaire est capable de détecter des structures complexes dans un ensemble de données, en revanche, l'utilisation de caractéristiques non-linéaires appropriées devrait permettre d'extraire plus d'informations. La méthode KS-PCA commence par projeter les données dans un certain espace de caractéristiques Ψ via une fonction (usuellement non-linéaire) Φ, puis exécute une PCA linéaire sur les données projetées. Comme l'espace de caractéristique Ψ pourrait être de dimension très élevée, le KS-PCA emploie le noyau de Mercer au lieu d'effectuer le calcul explicite des fonctions Φ.

$$\Phi : R^N \to H, x \to X. \tag{7.9}$$

Dans l'espace de projection H, nous supposons que la dimension des $\Phi(x)$ est arbitrairement grande, probablement infinie. Les données doivent être centrées également

dans cet espace, donc $\sum_{k=1}^{D} \Phi(x_k) = 0$. L'application du PCA trick dans l'espace de caractéristique H nous donne:

$$\bar{Cov} = \frac{1}{(N-1)} \sum_{i=1}^{N} \Phi(y_i)\Phi(y_i)^T \tag{7.10}$$

dont on extrait les valeurs propres satisfaisant:

$$\bar{\lambda} \cdot V = \bar{Cov} \cdot V \tag{7.11}$$

Les solutions V se trouvent dans l'écartement de $\Phi(y_1), \Phi(y_2), \cdots, \Phi(y_N)$. Ceci a deux conséquences utiles : d'abord, nous pouvons considérer l'équation équivalente:

$$\bar{\lambda}(\Phi(y_k)^T V) = (\Phi(y_k)^T \cdot \bar{Cov} \cdot V) \tag{7.12}$$

pour tout $k = 1, 2, \cdots, N$. Il existe des coefficients α_i $(i = 1, \cdots, N)$ tel que:

$$V = \sum_{i=1}^{N} \alpha_i \Phi(y_i) \tag{7.13}$$

En considérant les 2 équations eq. 7.12 et eq. 7.13, nous avons:

$$\bar{\lambda} \sum_{i=1}^{N} \alpha_i (\Phi^T(y_k) \cdot \Phi(y_i)) = \frac{1}{M} \sum_{i=1}^{N} \alpha_i (\Phi^T(y_k) \cdot \sum_{j=1}^{N} \Phi(y_j))(\Phi^T(y_j) \cdot \Phi(y_i)) \tag{7.14}$$

où $M = N - 1$.

Nous définissons alors la matrice K de dimension $N \times N$ par:

$$K_{i,j} = (\Phi^T(y_i) \cdot \Phi(y_j)) \tag{7.15}$$

ce qui permet d'écrire l'équation eq. 7.13 sous la forme : :

$$M\bar{\lambda}K\alpha = K^2\alpha \tag{7.16}$$

où α est le vecteur colonne composé des coefficients $\alpha_1, \cdots, \alpha_N$. Etant donné que K est symétrique, nous avons:

$$M\bar{\lambda}\alpha = K\alpha \tag{7.17}$$

Notez que K est une matrice définie semi-positif et symétrique. Nous avons alors plusieurs options pour le choix de la fonction K. En considérant l'étude statistique précédente sur les visages illuminés, le noyau gaussien semble être un choix raisonnable.

$$K_{i,j} = k(y_i, y_j) = e^{-\frac{\left\|y_i - y_j\right\|^2}{2\delta^2}} \tag{7.18}$$

L'extraction de valeurs propres dans l'espace de caractéristiques se fait par la diagonalisation de la matrice de similitude $K_{i,j}$.

7.2.5 Les Résultats expérimentaux

- **Critère d'évaluation:**

Pour estimer la performance de cette approche, nous définissons des critères d'évaluation dans cette section.

Nous comparons les erreurs point à point et pixel à pixel introduites d'une part par les méthodes proposées, et d'autre par l'approche AAMs standard. On utilise les formules suivantes:

$$E_{pt-pt} = \frac{1}{n} \sum \sqrt{(x_i - x_{gt,i})^2 + (y_i - y_{gt,i})^2} \tag{7.19}$$

où (x_i, y_i) sont les coordonnées des points d'intérêts relabélisés par l'AAMs, et (x_{gt}, y_{gt}) représente les coordonnées correspondantes à la vérité terrain.

et:

$$E_{pix-pix} = |\delta I|^2 = |I_i - I_m|^2 \tag{7.20}$$

où I_i est le valeur des vecteurs de niveau gris dans l'image, et I_m est la valeur des vecteurs de niveau gris pour le modèle courant.

$$E_{pt-pt}gain\% = \frac{E_{pt-pt}(AAMs) - E_{pt-pt}(kernel)}{E_{pt-pt}(AAMs)}\% \tag{7.21}$$

et

$$E_{pix-pix}gain\% = \frac{E_{pix-pix}(AAMs) - E_{pix-pix}(kernel)}{E_{pix-pix}(AAMs)}\% \tag{7.22}$$

Pour démontrer la supériorité des méthodes proposées, les équations eq.7.21 et 7.22 sont utilisées pour calculer le gain en termes de précision des résultats pour la forme et pour la texture. Les résultats sont présentés en valeur relatives par rapport à ceux des AAMs classiques.

Pour démontrer la supériorité des méthodes proposées, les équations eq.7.21 et 7.22 sont utilisées pour calculer le gain en termes de précision des résultats pour la forme et pour la texture. Les résultats sont présentés en valeur relatives par rapport à ceux des AAMs classiques dans le tableau 7.1.

Dans la figure 7.4, les images de la première et deuxième colonne sont synthétisées par les AAMs basés respectivement sur la KS-PCA, et sur la PCA probabiliste. Ils sont comparés avec les résultats des AAMs classique donnés par la troisième colonne, le visage original de référence étant donné par la quatrième colonne. Les images des trois premières lignes font partie de la base d'apprentissage, tandis que dans les trois dernières lignes, nous avons des images faisant partie uniquement de la base de test. Ces résultats montrent la capacité des méthodes proposées de synthétiser l'apparence d'un visage inconnu sous des conditions d'illumination variables. Une nette amélioration de la précision a ainsi été obtenue grâce à l'extraction des caractéristiques non-linéaires. Le gain quatitatif en termes précision sur les erreurs point-à-point et pixel-à-pixel est présenté dans la table 7.1.

		$E_{pt-pt} \quad gain\%$	$E_{pix-pix} \quad gain\%$
L'ensemble de training	KS-PCA	87.25%	61.36 %
	PPCA	34.58%	25.47%
L'ensemble de test	KS-PCA	76.16%	28.06 %
	PPCA	27.28%	15.06%

Table 7.1: Gain en termes de la précision - sur base de données de CMU PIE pour le problème de la illumination irrégulière.

7.2.6 Conclusion

Dans ce chapitre, nous avons proposé deux solutions pour extraire les caractéristiques faciales à base d'un modèle non linéaire, en travaillant dans un espace de projection qui a la capacité de capter les variations non-linéaires causées par l'illumination du visage. Dans ce but, nous avons proposé d'utiliser les AAMs en exploitant d'une part la PCA à noyaux (KS-PCA) et d'autre part une PCA probabiliste (PPCA). Ces deux approches ont consisté à introduire de la non linéarité dans les AAMs classique pour mieux gérer les modèles déformable non-linéaires.

Nos algorithmes ont été testés sur des images faciales de la base de données CMU PIE, pour extraire les paramètres d'apparence liés à des variations non-linéaires de l'illumination. Les résultats des méthodes proposées ont montré que les modèles,

Figure 7.4: **Comparaison des résultats** - Les résultats pour l'AAMs basé sur le KS-PCA sont dans la première colonne de gauche; les résultats pour l'AAMs basé sur le PPCA sont dans la deuxième colonne; les résultats pour l'AAMs classique sont dans la troisième colonne; dans la dernière colonne sont les visages d'original de référence.

ainsi construits, sont plus robustes aux variations d'illumination. Nous obtenons une amélioration significative des résultats par les 2 méthodes et plus particulièrement avec les AAMs basés sur le KS-PCA. Avec ces nouvelles approches, les AAMs sont capables de synthétiser des visages "semi brillant semi sombre" affecté par l'illumination, de manière plus précise.

7.3 Modèle à base des représentations parcimonieuses

Le modèle de représentation parcimonieuse est un nouveau composant des modèles intégrés. Il suppose la capacité de décrire des signaux comme des combinaisons linéaires de quelques atomes, pris dans un dictionnaire pré-défini, d'où l'importance du choix du dictionnaire. Dans ce chapitre, nous présentons nos travaux de recherche sur deux modèles d'apprentissage de dictionnaire en vue d'être utilisés pour des représentations parcimonieuses appliquées au problème de l'estimation des poses du visage.

Dans la section 7.3.1, nous proposons d'apprendre le dictionnaire en faisant appel à une Analyse en Composante Principale Incrémentale (Incremental-PCA). En combinant ce modèle avec une stratégie d'apprentissage à réactualisation, la redondance intra-classe pour chaque pose est éliminée, gardant ainsi uniquement la redondance inter-classe, utile pour la classification.

Dans la section 7.3.2 nous proposons une méthode qui réunie, dans un seul formalisme, le critère de la classification et l'optimisation de l'apprentissage du dictionnaire. Dans l'algorithme proposé, la représentation parcimonieuse est donnée par l'Orthogonal Matching Pursuit (OMP) [126] et l'apprentissage du dictionnaire est effectué par K-SVD [127].

Les résultats expérimentaux présentés dans la section 7.3.3 démontrent l'efficacité des approches proposés, notamment pour les deux applications de classification des poses qui ont été traitées, à savoir, soit dans des conditions d'illuminations dynamiques, soit sur des images de basse résolution. Ces résultats montrent également le grand pouvoir de représentativité des données en utilisant des dictionnaires construits par apprentisage.

7.3.1 Active Sparse Representation pose Classification (ASRC)

L'idée essentielle de ce schéma est de construire un dictionnaire réactualisable des poses du visage à travers l'Incremental PCA. Cependant, de nombreux paramètres peuvent fausser les résultats de la classification : nombre d'images d'apprentissage très important, grande variation de la redondance intra-classe, beaucoup de bruit, etc. Pour

contourner ces difficultés, une méthode basée sur une réactualisation conditionnelle est proposée, qui met à jour la base des vecteurs propres du visage seulement avec les images mal classées. L'algorithme 1 résume le déroulement complet de la classification.

Algorithm 1: Framework of the Adaptive Sparse Representation pose Classification (ASRC)

1:Input: a matrix of training images $I = \left\{I^1, I^2, \cdots, I^M\right\}$ from M classes, where $I^m \in R^{d \times n_m}$ represents the image set of the class m where d is the dimension of each image (in vector form), while n_m is the number of images in the class m.
2:Apply standard PCA on each image set to obtain the initial eigenspaces $A_m = \left[\begin{array}{cc} \bar{I}_m & U_m \end{array}\right] (m = 1, 2, \cdots, M)$.
3:Solve the l_1-minimization problem for the sparse representation $\left[\begin{array}{c} a_1 \\ b_1 \end{array}\right]$ of a new image from the second training subset.
4:Label the new training image with the class m for which the reconstruction residual of the new training image is minimal.
5:Check if the new image is labelled with the correct class?
Yes → return to **step 3**;
No → continue to **step 6**.
6:Update the online eigenspace of the class to which the new training image belongs to via Incremental-PCA.
7:Return to **step 3**.

Les résultats obtenus montrent que la méthode proposée rend les résultats la classification plus stables, et améliore sa performance particulièrement lorsque les images sont de très basse résolution ou que les conditions d'illumination changent dynamiquement.

7.3.2 Schéma d'apprentissage de dictionnaire pour une représentation parcimonieuse

Dans cette section, nous cherchons à répondre, dans un formalisme unique, aux deux objectifs visés, à savoir, une bonne classification et un dictionnaire adapté à la représentation parcimonieuse. Le schéma proposé est nommé DLSR dans la suite du texte, pour Dictionnary Learning Sparse Representation.

Pour le problème de la classification des poses, nous commençons par construire un dictionnaire spécifique pour chaque catégorie de pose. Puis, considérant une image de

test à classer, celle-ci est représentée par une combinaison linéaire de quelques atomes issus des dictionnaires appris, relatifs aux différentes poses. Le label correspondant est finalement déterminé en comparant les erreurs de reconstruction obtenues en n'utilisant que les coefficients liés à un seul dictionnaire, et donc à une seule pose. On voit bien que le choix d'un dictionnaire qui permet une représentation parcimonieuse des signaux est crucial pour le succès de l'estimation de la pose.

L'approche proposée modélise l'apparence des images de visages issues de la même pose, par apprentissage d'un dictionnaire D, à partir d'un ensemble de patches des images. Le dictionnaire est adapté à une représentation parcimonieuse, avec pour objectif, la minimisation de l'erreur de reconstruction de l'image cible pour coïncider avec le critère de classification de pose. Enfin, la combinaison des dictionnaires construits de toutes les classes de pose est utilisée comme un dictionnaire global pour la représentation parcimonieuse et la classification.

Les résultats expérimentaux de la section 7.3.3, présentent la performance de cette approche de classification par apprentissage de dictionnaire, comparativement à la méthode ASRC ainsi qu'à deux méthodes différentes issues de l'état de l'art.

Dans la figure 7.5, nous présentons notre approche pour le problème d'estimation de pose du visage en utilisant l'algorithme DLSR. Celle-ci montre l'apprentissage de dictionnaire et la procédure de classification de pose, où les blocs en grisé représentent les étapes de la classification des poses, tandis que les autres blocs représentent les étapes d'apprentissage des dictionnaires.

7.3.3 Les Résultats expérimentaux

Dans le but d'évaluer la performance de classification des deux algorithmes proposés, nous présentons les résultats expérimentaux obtenus pour deux bases de données d'image différentes. Nous évaluons d'abord la robustesse des nouvelles méthodes vis à vis des variations d'illumination sur la base de données CMU PIE [125]. Ensuite nous évaluons les méthodes sur la base de données LLP (le LIRIS Low-resolution Pose database) pour démontrer l'efficacité des algorithmes sur des images de basse résolution bruitées.

7.3.3.1 Variations de l'illumination:

Nous appliquons nos algorithmes sur la base de données CMU PIE. Le sous-ensemble choisi contient 12,240 images de 68 personnes, avec 9 différentes poses et 20 différentes conditions d'illumination chacune. La Figure 7.6 donne un exemple de cette base.

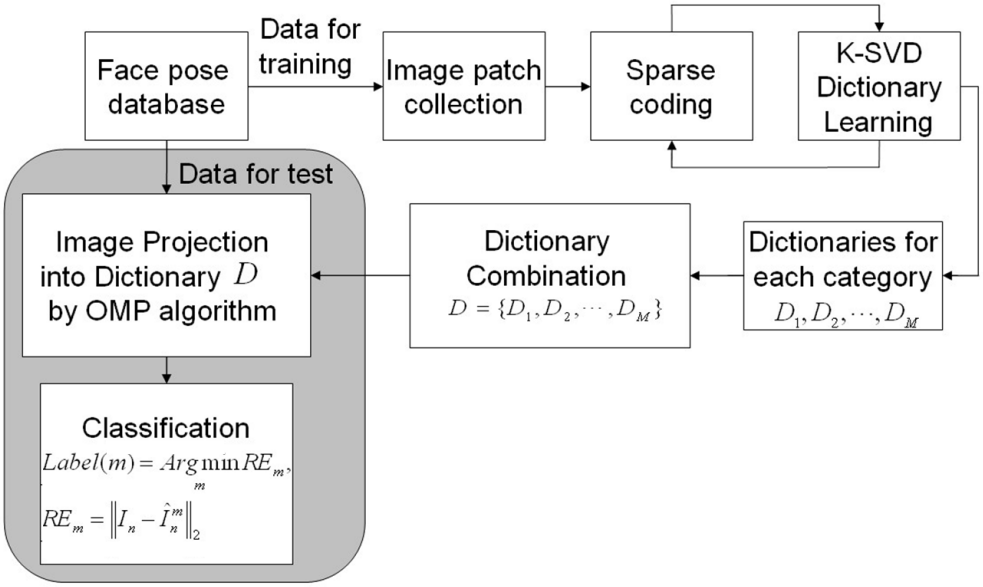

Figure 7.5: **La stratégie proposée** - Le schéma d'apprentissage de dictionnaire et la Représentation Parcimonieuse pour la classification de pose du visage

- La performance de l'algorithme ASRC

Pour l'Algorithme ASRC, nous utilisons pour l'apprentissage les images de 30 personnes (180 échantillons par personne) de différentes poses et avec des illuminations variables. Les visages sont d'abord localisés par le détecteur de visage de Viola-Jones [229], puis la région du visage est sous échantillonnée à la taille 24×24. L'espace des vecteurs propres initial pour l'Incremental-PCA est construit avec 360 images (incluant les 9 poses dans les 20 conditions d'illumination, de 2 personnes). Ensuite, les images des 28 autres personnes de l'ensemble d'apprentissage sont rajoutées dans la base des vecteurs propres en appliquant l'Incremental-PCA. La table 7.2 montre, à travers la matrice de confusion, les résultats de la classification pour les 9 différents poses. On peut remarquer que les confusions de classification sont faibles et généralement entre des catégories proches.

- Performance de l'algorithme DLSR

Pour l'algorithme DLSR, le sous-ensemble choisi de CMU est aussi divisé en deux parties, la première partie inclut les images de 30 personnes comme ensemble d'apprentissage, tandis que la deuxième partie inclut les images de 38 autres personnes comme ensemble de test. Comme précédemment, les visages sont d'abord localisés par

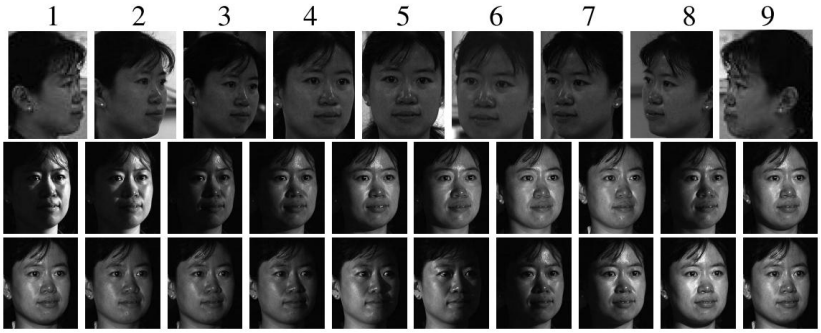

Figure 7.6: **Les exemples de la base de données CMU PIE** - Le rangée supérieure: 9 poses différents du visage dans la même condition d'illumination. Les deux rangées en bas: 20 conditions d'illumination différentes.

Class	1	2	3	4	5	6	7	8	9
1	93.7%	5.7%	0.6%						
2		97.2%	2.8%						
3		0.8%	97.5%	1.7%					
4				94.9%	5.1%				
5				0.9%	98.2%	0.9%			
6						95.0%	4.2%	0.8%	
7						0.6%	95.8%	3.6%	
8						0.08%	4.2%	93.2%	2.5%
9							1.9%	5.9%	92.2%

Table 7.2: La matrice de confusion de la classification de pose facial pour l'algorithme ASRC sur la base de données CMU PIE (dans le pourcentage) pour les 9 poses différents.

le détecteur du visage, puis sous échantillonnée à la taille 15×18. Pour chaque classe de pose, nous extrayons tous les patchs de taille 6×6 pour construire le dictionnaire initial D_0^m. Enfin, nous appliquons l'algorithme K-SVD sur D_0^m pour former un dictionnaire comme cela a été expliqué dans la section 7.3.2. Nous appliquons 50 itérations pour apprendre un dictionnaire d'analyse.

La table 7.3 montre les résultats de la classification obtenus par DLSR pour les 9 poses. Le schéma proposé apparaît effectivement très efficace.

7.3.3.2 Variations de la résolution et du bruit

Nous avons construit une grande base de données qui inclut 248,025 images de visage triées dans cinq catégories : 64770 visages frontaux; 30469 profil gauche; 39993 profil

Class	1	2	3	4	5	6	7	8	9
1	100%								
2		97.4%	2.6%						
3		1.7%	97.2%	1.1%					
4			1.6%	97.8%	0.6%				
5				0.2%	98.8%	1.0%			
6				0.08%	0.8%	97.9%	1.2%		
7					0.13%	1.5%	97.5%	0.9%	
8							0.3%	98.0%	1.7%
9							0.3%	0.7%	99.0%

Table 7.3: La matrice de confusion de la classification de pose facial pour l'algorithme DLSR sur la base de données de CMU PIE (dans le pourcentage) pour les 9 poses différents

droit; 56396 quart gauche; 56397 quart droit. (Quelques exemples sont donnés dans la figure 7.7):

Nous avons collecté des images de visages depuis le net et depuis d'autres bases de données faciales. Le détecteur de visage commercial face.com (http: // face.com) a été utilisé pour aider à l'extraction des images de visage. Avec des sélections/corrections manuelles, les images facials ont été coupées et classées dans les cinq catégories de pose. Finalement, plusieurs transformations ont été appliquées aux images existantes pour augmenter leur nombre et leur variabilité: rotations planes, translations, flips horizontaux et brouillage. Elles ont été toutes redimensionnées à la taille 36×36. La base de données est appelée LIRIS Low-resolution Pose (LLP) database.

Figure 7.7: **Les exemples de la base de données LLP** - Quelques exemples de la base de données LIRIS Low-resolution Pose.

- Performance de l'algorithme ASRC

Pour l'Algorithme ASRC, et pour la phase d'apprentissage, l'espace des vecteurs propres initial pour l'Incremental-PCA est construit à partir de 50 images choisies aléatoirement dans chaque classe de la base de données LLP, sous-échantillonnées à la taille 24×24. Puis, des échantillons supplémentaires sont rajoutés, mettant à jour l'espace propre, jusqu'à ce que le nombre d'échantillons atteigne 1000 (par classe), afin d'obtenir un bon taux de classification. Des résultats de la classification pour cinq poses différentes sont présentés dans la table 7.4. où nous pouvons constater que les confusions de classification sont faibles.

Class	L	QL	F	QR	R
L	92.66%	6.68 %	0.66%		
QL	3.86%	91.55%	4.59%		
F	0.81%	1.2%	95.65%	1.9%	0.36%
QR			4.03 %	89.74%	6.24%
R			0.16%	8.83%	91.00%

Table 7.4: La matrice de confusion de la classification de pose facial pour l'algorithme ASRC sur la base de données LLP

- Performance de l'algorithme DLSR

La table 7.5 montre la matrice de confusion pour la classification à base de DLSR. On voit des résultats meilleurs que précédemment et les confusions de la classification sont généralement entre des catégories proches (i.e. L et QL, R et QR). La méthode DLSR proposée semble être particulièrement robuste pour traiter des images de basse résolution et bruitées.

Class	L	QL	F	QR	R
L	96.30%	2.68 %	1.02%		
QL	2.86%	95.72%	1.42%		
F	0.36%	1.20%	97.25%	1.19%	
QR			1.06 %	96.91%	2.03%
R				1.44%	98.56%

Table 7.5: La matrice de confusion de la classification de pose facial pour l'algorithme DLSR sur la base de données LLP

Class	1	2	3	4	5	6	7	8	9	AVG
DLSR	100%	97.4%	97.2%	98.0%	97.8%	98.8%	97.9%	97.5%	99.0%	98.12%
ASRC	97.2%	96.0%	97.1%	94.6%	98.3%	96.2%	97.0%	95.7%	96.8%	96.5%
SRC	91.0%	93.3%	96.1%	91.8%	93.1%	88.2%	94.3%	92.2%	86.4%	91.82%
PSFS	96.8%	78.4%	65.0%	79.5%	96.8%	93.8%	53.4%	90.0%	92.5%	84.91%
Polynomial Kernel SVM	84.4%	83.5%	87.8%	84.6%	91.5%	85.7%	86.3%	88.0%	86.2%	86.44%
kNN(k=10)	82.8%	80.9%	83.7%	79.7%	84.8%	87.1%	81.3%	85.1%	86.2%	83.51%

Table 7.6: Comparaison des taux de la classification entre Kernel-SVM, k-Nearest Neighbors, PSFS algorithme, SRC algorithme, l'algorithme ASRC et l'algorithme DLSR sur la base de données de CMU PIE (dans pourcentage) pour les 9 poses différents.

7.3.3.3 Comparaison avec d'autres méthodes de classification

Dans cette section, nous comparons les méthodes proposées avec quatre approches, dont deux approches classiques, et deux issues de l'état de l'art récent. Les deux premières sont le Polynomial Kernel SVM [230] et le k-Nearest Neighbors [231] (k=10). Pour PK-SVM et kNN, l'espace de projection est construit directement avec les images d'apprentissage sous-échantillonnées à la résolution 24×24 (identique à l'apprentissage de ASRC). PK- SVM apprend les hyperplans en utilisant un noyau polynomial $K(x, y) = (x^T y + c)^d$ $(d = 2)$, tandis que pour le kNN, un objet est classifié par un vote majoritaire parmi ses 10 voisins dans l'espace Euclidien.

Les deux autres approches auxquelles on se compare sont [232] et [222]. Dans la méthode PSFS proposé par [232], les auteurs classifient les différentes poses via Adaboost combiné à une procédure statistique, tandis que dans SRC [222] les auteurs utilisent la $l_1 - norm$ pour classifier les visages. Tous ces tests sont effectués sur la base de données CMU PIE pour pouvoir être comparés.

La table 7.6 donne les taux de classification obtenus avec les six méthodes et où l'on voit nettement le gain réalisé avec ASRC et DLSR. Pour ASRC, ce résultat s'explique par le fait que l'apprentissage du dictionnaire par l'Incremental-PCA réduit la redondance intra-classe et permet de construire un dictionnaire de composantes orthogonales avec les variations nécessaires pour chaque catégorie de pose, sans toute fois chercher à minimiser l'erreur de reconstruction. Avec DLSR, cette dernière contrainte est prise en compte dans l'algorithme et on peut voir l'amélioration ainsi engendrée.

7.3.4 Conclusion

Nous avons présenté dans ce chapitre deux systèmes robustes pour l'estimation de poses faciales basés sur des représentations parcimonieuses et apprentissage de dictionnaires

La première méthode propose l'apprentissage d'un dictionnaire réactualisable par une Incremental-PCA afin de réduire la redondance intra-classe de chaque classe de pose, et la seconde génère un dictionnaire sur-complet permettant d'intégrer dans l'algorithme la prise en compte de l'erreur de reconstruction.

Après avoir discuté de l'influence de certains paramètres, nous avons évalué nos méthodes en utilisant les paramètres optimaux. Les deux algorithmes proposés sont comparés avec deux méthodes de classification classiques et deux méthodes de l'état de l'art.

Les classifieurs ASRC et DLSR sont robustes aux changements d'apparence dus notamment à l'illumination, le bruit ou la résolution. Les résultats des tests menés sur les bases CMU PIE, LLP ont bien montré le gain réalisé par rapport à d'autres approches.

7.4 Normalisation de l'illumination par combinaison de GMM et de l'apprentissage de dictionnaires conjointement optimisés

Dans de nombreuses situations, les variations d'illumination posent de sérieux problèmes tout au long du processus d'analyse faciale. Un moyen de contourner ces difficultés est de chercher à normaliser l'illumination afin de limiter son impact sur les traitements.

Dans cette section, nous proposons un nouveau schéma d'apprentissage du dictionnaire pour la normalisation de l'illumination (DLIN pour Dictionnary Learning for Illumination Normalization). Celui-ci est basé sur l'optimisation conjointe de 2 dictionnaires traitant respectivement des visages à illumination normale et irrégulière. Nous utilisons par la suite un modèle de mixture de Gaussiennes (GMM) pour améliorer les capacités de modélisation de l'ensemble de données étant donnée leur distribution complexe.

Concrètement, nous apprenons une paire de dictionnaires simultanément. La paire de dictionnaires vise à caractériser les deux domaines structurels de deux cas d'illumination: D_{Norm} pour les patches d'images normalement illuminées et D_{Illum} pour les patches d'images illuminées irrégulièrement. La représentation parcimonieuse d'un patch illuminé irrégulièrement sur le dictionnaire D_{Illum} sera directement utilisée pour récupérer le patch correspondant dans l'image normale de D_{Norm}. Dans cette étude, nous essayons d'apprendre les deux dictionnaires sur-complets selon une stratégie à deux étapes. Pour imposer aux paires des patches d'image d'avoir les mêmes

représentations parcimonieuse aussi bien sur D_{Illum} que sur D_{Norm}, nous apprenons les deux dictionnaires simultanément en les combinant, avec une normalisation appropriée, comme cela est illustré dans le figure 7.9.

7.4.1 Un schéma d'apprentissage de dictionnaire pour la Normalisation de l'illumination (DLIN))

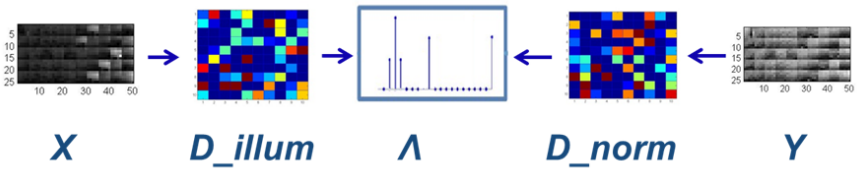

Figure 7.8: **La stratégie proposée** - Un schéma d'apprentissage de dictionnaire paire

Comme indiqué dans le figure 7.8, à chaque patch en entrée x, nous associons une représentation parcimonieuse sur le dictionnaire D_{Illum}. Ces mêmes coefficients de la base D_{Illum} seront utilisés avec la base D_{norm} pour donner les patchs normalisés.

Notons X et Y les ensembles d'apprentissage constitués par les patches des visages avec une illumination respectivement normale et particulière. Nous proposons de minimiser la fonction d'énergie ci-dessous pour trouver les dictionnaires souhaités:

$$\min_{D,\Lambda} \|X - D_{Illum}\Lambda\|_2^2 + \|Y - D_{Norm}\Lambda\|_2^2 + \lambda \|\Lambda\|_1 \, s.t. \quad \|d_{x,i}\|_{l_2} \leq 1, \|d_{y,i}\|_{l_2} \leq 1, \quad \forall i \tag{7.23}$$

où λ est le paramètre de régularisation pour équilibrer les termes dans la fonction d'énergie, $d_{x,i}, d_{y,i}$ sont respectivement les atomes de D_{Illum} et de D_{Norm}. La fonction objectif dans l'équation Eq. 7.23 n'est pas conjointement convexe par rapport à D_{Illum}, D_{Norm}, mais elle l'est par rapport à l'un des deux si l'autre est fixe. Donc, nous pouvons concevoir un algorithme itératif pour optimiser ces variables alternativement.

Etant donné les structures complexes dans les images d'illumination irrégulière, apprendre seulement une paire de dictionnaires ne suffira pas pour couvrir toutes les variations lors de la normalisation de l'illumination. Par exemple, dans la synthèse de visage normalisé, la différence d'apparence entre le visage illuminé et le visage normalisé peut varier significativement dans les différentes régions du visage. Donc un multi-modèle devrait être appris pour augmenter la robustesse. Intuitivement, une pré-classification

pourrait séparer les données d'apprentissage en plusieurs groupes permettant ainsi une description plus stable et linéaire dans chaque groupe.

Comme illustré dans la figure 7.9, L'apprentissage du dictionnaire basé sur un multi-modèle est initialisé par la pré-classification des paires de patches pris dans les images de visages normalement et irrégulièrement illuminés, et correspondant à la même position dans l'image. Nous avons choisi un Modèle de Mixture de Gaussiennes (GMM) [253] que nous appliquons pour la procédure de pré-classification

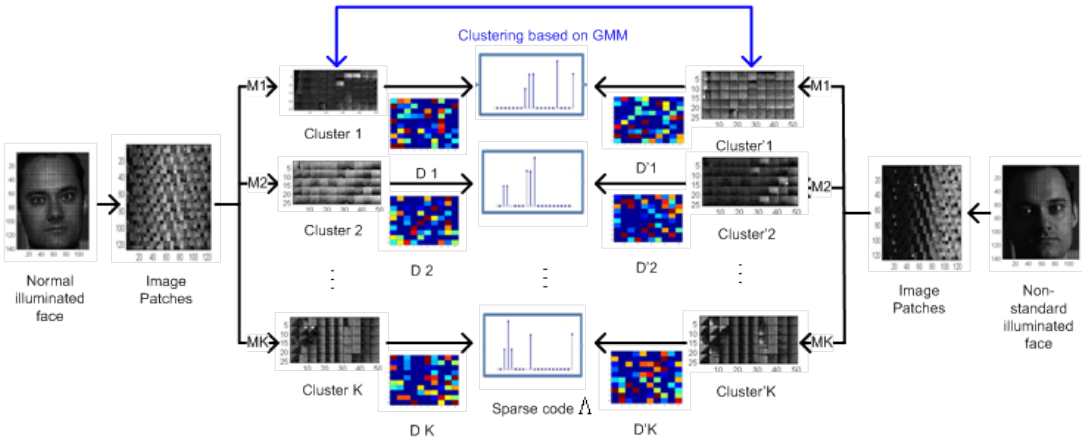

Figure 7.9: **La stratégie proposée** - Un schéma d'apprentissage de dictionnaire four la normalisation d'illumination (DLIN)

A partir de la pré-classification, nous apprenons la distribution des patches associée à chaque classe. La structure des images du visage montre bien une distribution spatiale distincte pour chaque classe comme on peut le voir sur la figure 7.10. On peut remarquer que les patches des différents groupes se concentrent en des emplacements spatiaux différents. Et nous pouvons conclure de cette étude que les résultats de la pré-classification sont fortement corrélés avec la structure des visages.

Grâce à l'apprentissage des dictionnaires D_{Illum} et D_{Norm}, nous pouvons trouver pour une image donnée x, une représentation parcimonieuse sur le dictionnaire D_{Illum}. La base correspondant aux patches à illumination normale D_{Norm} sera combinée aux coefficients pour produire le patch d'image normalisée y. Ce problème de synthèse est résolu par l'optimisation de la fonction suivante:

$$\min_{\alpha^k} \left\| x_i - D^k_{Illum}\alpha^k_i \right\|^2_2 + \left\| y_i - D^k_{Norm}\alpha^k_i \right\|^2_2 + \lambda \left\| \alpha^k_i \right\|_1 \tag{7.24}$$

où x_i est un patch de x et y_i est le patch correspondante dans l'estimation in-

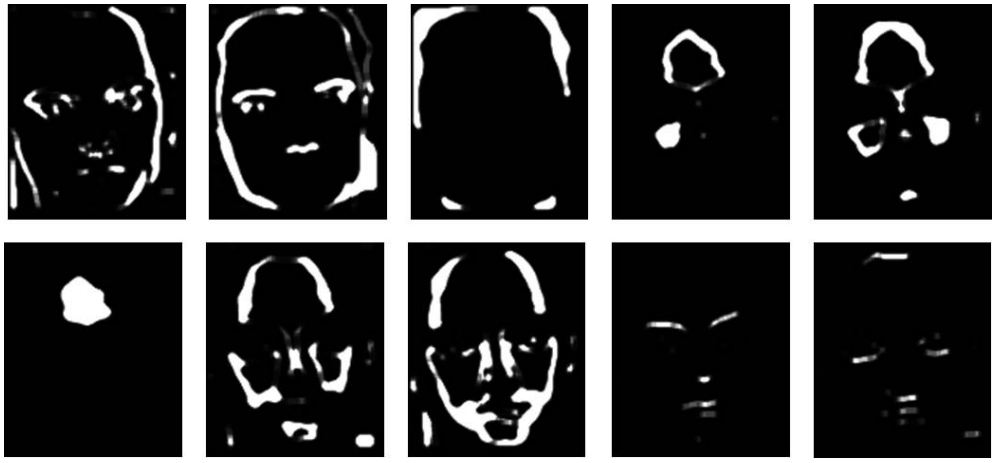

Figure 7.10: **Les exemples de la distribution des groupes** - Chaque sous-figure montrent la distribution d'un groupe

termédiaire de y qui a été synthétisé. D^k_{Illum}, D^k_{Norm} et α^k_i étant respectivement les 2 dictionnaires appris et le code parcimonieux décrivant la classe k.

L'image normalisée y_i peut être calculée par l'équation eq. 7.25, comme cela est illustré dans figure 7.11:

$$y_i = \sum_{k=1}^{K} P(M_k|Loc_i, x_i) y_i^k \tag{7.25}$$

Où M_k représente la paire de dictionnaire dans chaque groupe K, Loc_i représentent la localisation du patch x_i dans une image faciale. Puisqu'il existe une relation entre la pré-classification et la structure des visages humaine comme démontré dans la figure 7.10. L'utilisation de la probabilité commune de la localisation du patch et la sélection modèle améliore l'exactitude pour classifier le patch illuminée.

7.4.2 Les Résultats expérimentaux

La base de données CMU PIE [125] est choisie pour l'évaluation. Le CMU PIE consiste en 68 individus. Des images de visages frontaux dans 20 conditions d'illumination différentes sont choisies pour l'apprentissage et le test. Dans nos expérimentations, toutes les images sont simplement alignées et chaque image est coupée et redimensionnée à la taille 115×140.

Dans la première ligne de la figure 7.12, on peut remarquer que les images d'illumination sont bien normalisées mais il existe toute fois un phénomène de bloc évident. Ceci est causé par la combinaison des patches locaux et aussi par l'étape de

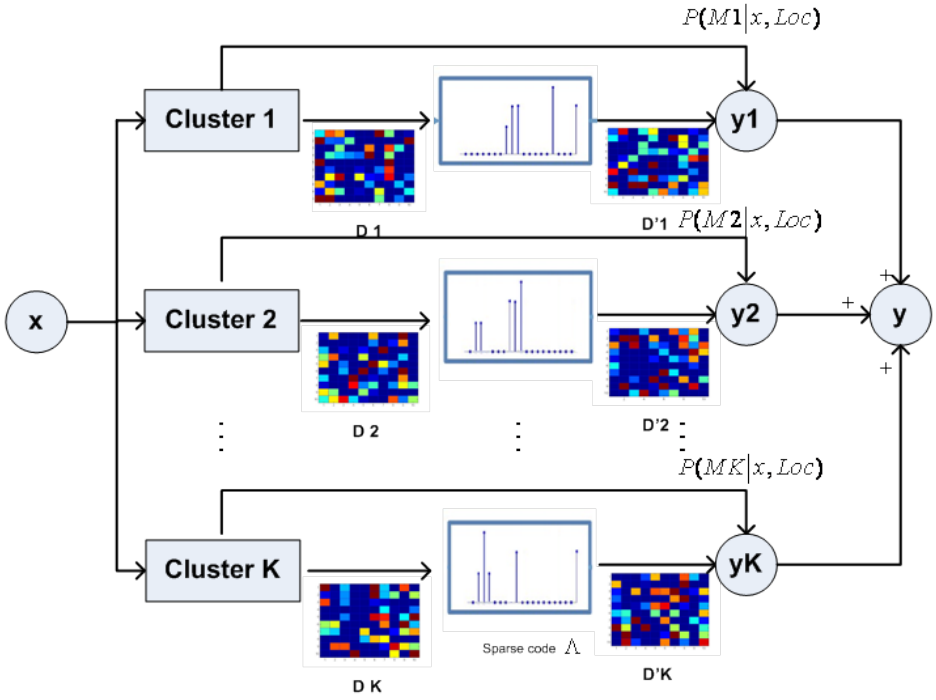

Figure 7.11: **Processus de synthèse** - Illustration du processus d'inférence GMM

classification lors de la procédure de la reconstruction. Pour éliminer l'influence des blocs, nous appliquons une méthode de débruitage [256]. Les résultats de la normalisation débruitée sont donnés dans deuxième rangée (b) du figure 7.12.

7.4.3 Conclusion

Dans cette section, Nous avons proposé un nouveau cadre d'apprentissage de dictionnaire pour la Normalisation d'Illumination (DLIN). DLIN est basé sur la représentation parcimonieuse en termes de dictionnaires couplés conjointement optimisés du visage illuminé et du visage normal. Nous utilisons plus loin un modèle de GMM pour augmenter la capacité de modélisation de l'ensemble des données de poses, vu leur distribution complexe. Les résultats expérimentaux démontrent l'efficacité de la parcimonie. Dans le cadre d'un travail futur, nous étendrons cette méthode à la synthèse d'image et à la reconnaissance faciale.

Figure 7.12: **Les résultats de la normalisation d'Illumination** - (a): Les images originals du visage normalisées. (b): Les images du visage normalisées débruitantes. (c): Les images du visage normales. (d): Les images du visage illuminées irrégulièrement originales.

7.5 Conclusion

Dans ce manuscrit nous avons présenté nos travaux de recherche sur l'analyse du visage. Notre objectif principal était de construire plusieurs algorithmes efficaces qui améliorent les fonctions d'analyse faciale sous différentes poses et conditions d'illumination. Cet objectif principal a été atteint par une réduction non linéaire de la dimension et l'utilisation de représentations parcimonieuses. Les résultats expérimentaux ont démontré l'efficacité des algorithmes proposés.

Globalement, une caractéristique importante de ce travail est que nous avons conçu des approches efficaces pour l'analyse des visages, tenant compte de la distribution de données et combinant des méthodes issues de différents domaines de recherche, incluant le modèle intégrant KS-PCA et le modèle de Incremental PCA, la Minimisation de $L_1 norme$ et le K-SVD méthode pour apprentissage de dictionnaire sur-complet, etc. Nous pouvons ainsi, considérer l'interdisciplinarité comme une contribution également importante de notre travail de recherche

7. RÉSUMÉ EN FRANÇAIS

References

[1] J. W. Cooley and J. W. Tukey. **An algorithm for the machine calculation of complex Fourier series**. *Mathematics of computation*, **19**(90):297–301, 1965. 7

[2] D. J. Field. **What is the goal of sensory coding?** *Neural computation*, **6**(4):559–601, 1994. 8

[3] A. K. Jain. *Fundamentals of digital image processing*, **3**. Prentice-Hall Englewood Cliffs, 1989. 8

[4] S. Mallat. *A wavelet tour of signal processing*. Access Online via Elsevier, 1999. 8, 9

[5] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2005. 8, 45, 52, 143

[6] A. J. Bell and T. J. Sejnowski. **An information-maximization approach to blind separation and blind deconvolution**. *Neural computation*, **7**(6):1129–1159, 1995. 9

[7] R. A. DeVore. **Nonlinear approximation**. *Acta numerica*, **7**:51–150, 1998. 9

[8] B. Schölkopf and A. Smola. **Nonlinear component analysis as a kernel eigenvalue problem**. *Neural computation*, **10**(5):1299–1319, 1998. 10, 12

[9] V. Vapnik. *The nature of statistical learning theory*. springer, 2000. 10

[10] S. Mika, G. Ratsch, J. Weston, B. Schölkopf, and K. R. Müllers. **Fisher discriminant analysis with kernels**. In *Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop*, pages 41–48. IEEE, 1999. 10

[11] R. Rosipal and L. J. Trejo. **Kernel partial least squares regression in reproducing kernel hilbert space**. *The Journal of Machine Learning Research*, **2**:97–123, 2002. 10

[12] C. Richard, J. C. M. Bermudez, and P. Honeine. **Online prediction of time series data with kernels**. *IEEE Transactions on Signal Processing*, **57**(3):1058–1067, 2009. 10

[13] B. Schölkopf, R. C. Williamson, A. J. Smola, and J. C. Shawe-Taylor, J.and Platt. **Support Vector Method for Novelty Detection**. In *NIPS*, **12**, pages 582–588, 1999. 10

[14] S. Mika, B. Schölkopf, A. Smola, K. R. Müllers, M. Scholz, and G. Rätsch. **Kernel PCA and De-Noising in Feature Spaces**. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 11*, pages 536–542. MIT Press, 1999. 10

[15] G. Camps-Valls, J. L. Rojo-Álvarez, and M. Martínez-Ramón. *Kernel methods in bioengineering, signal and image processing*. IGI Global, 2007. 10

[16] J. B. Allen and L. R. Rabiner. **A unified approach to short-time Fourier analysis and synthesis**. *Proceedings of the IEEE*, **65**(11):1558–1564, 1977. 11

[17] W. B. Pennebaker. *JPEG: Still image data compression standard*. Springer, 1992. 11

[18] D. Gabor. **Theory of communication. Part 1: The analysis of information**. *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, **93**(26):429–441, 1946. 11

[19] J. G. Daugman. **Two-dimensional spectral analysis of cortical receptive field profiles**. *Vision research*, **20**(10):847–856, 1980. 11

[20] J. G. Daugman. **Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters**. *Optical Society of America, Journal A: Optics and Image Science*, **2**(7):1160–1169, 1985. 11

[21] J. G. Daugman. **Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression**. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **36**(7):1169–1179, 1988. 11

[22] M. Porat and Y. Y. Zeevi. **The generalized Gabor scheme of image representation in biological and machine vision**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**(4):452–468, 1988. 11

[23] P. Burt and E. Adelson. **The Laplacian pyramid as a compact image code**. *IEEE Transactions on Communications*, **31**(4):532–540, 1983. 12

[24] I. Daubechies. *Ten lectures on wavelets*, **61**. SIAM, 1992. 12

[25] Y. Meyer and D. H. Salinger. *Wavelets and operators*, **1**. Cambridge university press, 1995. 12

[26] A. Grossmann and J. Morlet. **Decomposition of Hardy functions into square integrable wavelets of constant shape**. *SIAM journal on mathematical analysis*, **15**(4):723–736, 1984. 12

[27] D. L. Donoho. **Orthonormal ridgelets and linear singularities**. *SIAM Journal on Math. and Analysis*, **31**:1062–1099, 1998. 12

[28] S. Mallat and S. Zhong. **Characterization of signals from multiscale edges**. *IEEE Transactions on pattern analysis and machine intelligence*, **14**(7):710–732, 1992. 12

[29] S. G. Mallat. **A theory for multiresolution signal decomposition: the wavelet representation**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(7):674–693, 1989. 12

[30] S. Mallat and W. L. Hwang. **Singularity detection and processing with wavelets**. *IEEE Transactions on Information Theory*, **38**(2):617–643, 1992. 13

[31] R. R. Coifman, Y. Meyer, and V. Wickerhauser. **Wavelet analysis and signal processing**. In *In Wavelets and their Applications*. Citeseer, 1992. 13

[32] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. **Shiftable multiscale transforms**. *IEEE Transactions on Information Theory*, **38**(2):587–607, 1992. 14

[33] G. Beylkin. **On the representation of operators in bases of compactly supported wavelets**. *SIAM Journal on Numerical Analysis*, **29**(6):1716–1740, 1992. 14

[34] G. P. Nason and B. W. Silverman. **The stationary wavelet transform and some statistical applications**. In *Wavelets and statistics*, pages 281–299. Springer, 1995. 15

[35] R. R. Coifman and D. L. Donoho. *Translation-invariant de-noising*. Springer, 1995. 15

[36] S. Mallat and Z. Zhang. **Matching pursuits with time-frequency dictionaries**. *IEEE Trans. Signal Processing*, **41**:33973415, 1999. 15, 20, 82, 92

[37] S. S. Chen, D. L. Donoho, and M. A. Saunders. **Atomic decomposition by basis pursuit**. *SIAM journal on scientific computing*, **20**(1):33–61, 1998. 15, 20, 81

[38] A. M. Bruckstein, D. L. Donoho, and M. Elad. **From sparse solutions of systems of equations to sparse modeling of signals and images**. *SIAM review*, **51**(1):34–81, 2009. 15

[39] E. J. Candes and D. L. Donoho. **Curvelets: A surprisingly effective nonadaptive representation for objects with edges**. Technical report, DTIC Document, 2000. 17

[40] E. Candes, L. Demanet, D. Donoho, and L. Ying. **Fast discrete curvelet transforms**. *Multiscale Modeling & Simulation*, **5**(3):861–899, 2006. 17

[41] J. Starck, E. J Candès, and D. L. Donoho. **The curvelet transform for image denoising**. *IEEE Transactions on Image Processing,*, **11**(6):670–684, 2002. 17

[42] E. J. Candes and D. L. Donoho. **Continuous curvelet transform: I. Resolution of the wavefront set**. *Applied and Computational Harmonic Analysis*, **19**(2):162–197, 2005. 17

[43] L. Ying, L. Demanet, and E. Candes. **3D discrete curvelet transform**. In *Optics & Photonics 2005*, pages 591413–591413. International Society for Optics and Photonics, 2005. 17

[44] M. N. Do and M. Vetterli. **Contourlets: a directional multiresolution image representation**. In *2002 International Conference on Image Processing. 2002. Proceedings*, **1**, pages I–357. IEEE, 2002. 17

[45] M. N. Do and M. Vetterli. **The contourlet transform: an efficient directional multiresolution image representation**. *IEEE Transactions on Image Processing*, **14**(12):2091–2106, 2005. 17

[46] Y. Lu and M. N. Do. **A new contourlet transform with sharp frequency localization**. In *2006 IEEE International Conference on Image Processing*, pages 1629–1632. IEEE, 2006. 17

[47] Y. M. Lu and M. N. Do. **Multidimensional directional filter banks and surfacelets**. *IEEE Transactions on Image Processing*, **16**(4):918–931, 2007. 17

[48] A. L. Da Cunha, J. Zhou, and M. N. Do. **The nonsubsampled contourlet transform: theory, design, and applications**. *IEEE Transactions on Image Processing*, **15**(10):3089–3101, 2006. 17

[49] R. Eslami and H. Radha. **Translation-invariant contourlet transform and its application to image denoising**. *IEEE Transactions on Image Processing*, **15**(11):3362–3374, 2006. 17

[50] E. Le Pennec and S. Mallat. **Sparse geometric image representations with bandelets**. *IEEE Transactions on Image Processing*, **14**(4):423–438, 2005. 18

[51] G. Peyré and S. Mallat. **Surface compression with geometric bandelets**. In *ACM Transactions on Graphics (TOG)*, **24**, pages 601–608. ACM, 2005. 18

[52] N. Kingsbury. **Complex wavelets for shift invariant analysis and filtering of signals**. *Applied and computational harmonic analysis*, **10**(3):234–253, 2001. 18

[53] I. W. Selesnick, R. G. Baraniuk, and N. C. Kingsbury. **The dual-tree complex wavelet transform**. *IEEE Signal Processing Magazine*, **22**(6):123–151, 2005. 18

[54] G. Easley, D. Labate, and W. Q. Lim. **Sparse directional image representations using the discrete shearlet transform**. *Applied and Computational Harmonic Analysis*, **25**(1):25–46, 2008. 19

[55] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P. L. Dragotti. **Directionlets: anisotropic multidirectional representation with separable filtering**. *IEEE Transactions on Image Processing*, **15**(7):1916–1933, 2006. 19

[56] S. Mallat. **Geometrical grouplets**. *Applied and Computational Harmonic Analysis*, **26**(2):161–180, 2009. 19

[57] J. A. Tropp. **Greed is good: Algorithmic results for sparse approximation**. *IEEE Transactions on Information Theory*, **50**(10):2231–2242, 2004. 20

[58] R. Tibshirani. **Regression shrinkage and selection via the lasso**. *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**:267288, 1996. 20, 81

[59] I. F. Gorodnitsky and B. D. Rao. **Sparse signal reconstruction from limited data using FOCUSS: A reweighted norm minimization algorithm**. *IEEE Transactions on Signal Processing*, **45**:600616, 1997. 20

[60] D. P. Wipf and B. D. Rao. **Sparse Bayesian learning for basis selection**. *IEEE Transactions on Signal Processing*, **52**(8):2153–2164, 2004. 20

[61] J. A. Tropp and S. J. Wright. **Computational methods for sparse solution of linear inverse problems**. *Proceedings of the IEEE*, **98**(6):948–958, 2010. 20

[62] B. A. Olshausen and D. J. Field. **Sparse coding with an overcomplete basis set: A strategy employed by V1?** *Vision research*, **37**(23):3311–3325, 1997. 21, 84

[63] K. Engan, S. O. Aase, and H. J. Hakon. **Method of optimal directions for frame design**. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999. Proceedings.*, **5**, pages 2443–2446. IEEE, 1999. 21

[64] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. Lee, and T. J. Sejnowski. **Dictionary learning algorithms for sparse representation**. *Neural computation*, **15**:349–396, 2003. 21

[65] X. Jiang. **Extracting image orientation feature by using integration operator**. *Pattern Recognition*, **40**(2):705–717, 2007. 22

[66] M. Yaghoobi, T. Blumensath, and M. E. Davies. **Dictionary learning for sparse approximations with the majorization method**. *IEEE Transactions on Signal Processing*, **57**(6):2178–2191, 2009. 22

[67] M. S. Lewicki and T. J. Sejnowski. **Learning overcomplete representations**. *Neural computation*, **12**(2):337–365, 2000. 22

[68] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. **Online learning for matrix factorization and sparse coding**. *The Journal of Machine Learning Research*, **11**:19–60, 2010. 22

[69] D. M. Bradley and J. A. Bagnell. **Differentiable sparse coding**. *Advances in Neural Information Processing Systems*, **21**:113–120, 2008. 22

[70] P. Schmid-Saugeon and A. Zakhor. **Dictionary design for matching pursuit and application to motion-compensated video coding**. *IEEE Transactions on Circuits and Systems for Video Technology*, **14**(6):880–886, 2004. 22

[71] M. Aharon, M. Elad, and A. M. Bruckstein. **The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representations**. *IEEE Transactions on Signal Processing*, **54**:43114322, 2006. 22, 84, 119

[72] M. Yaghoobi, L. Daudet, and M. E. Davies. **Parametric dictionary design for sparse coding**. *IEEE Transactions on Signal Processing*, **57**(12):4800–4810, 2009. 23

[73] Boris Mailhé, Sylvain Lesage, Rémi Gribonval, Frédéric Bimbot, Pierre Vandergheynst, et al. **Shift-invariant dictionary learning for sparse representations: extending K-SVD**. In *16th EUropean SIgnal Processing COnference (EUSIPCO'08)*, 2008. 23

[74] P. Sallee and B. A. Olshausen. **Learning sparse multiscale image representations**. *Advances in neural information processing systems*, pages 1351–1358, 2003. 23

[75] Kjersti Engan, Karl Skretting, and John Håkon Husøy. **Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation**. *Digital Signal Processing*, **17**(1):32–49, 2007. 24

[76] R. Gribonval and M. Nielsen. **Sparse representations in unions of bases**. *IEEE Transactions on Information Theory*, **49**:33203325, 2003. 24, 84

[77] M. Kass, A. Witkin, and D. Terzopoulos. 30

[78] T. Cootes, C. Taylor, D. Cooper, and J. Graham. **Active shape models - their training and application**. *Computer Vision and Image Understanding*, **61**(1):38–59, 1995. 30, 32

[79] L. Wiskott, J. Fellous, N. Kruger, and C. von der Malsburg. **Face Recognition by Elastic Bunch Graph Matching**. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **19(7)**:775–779, 1997. 31

[80] V. Blanz and T. Vetter. **A morphable model for the synthesis of 3D faces**. In *SIGGRAPH '99 Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999. 31

[81] V. Blanz and T. Vetter. **Face recognition based on fitting a 3D morphable model**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**:1063–1074, September 2003. 32, 112, 115

[82] A. Lanitis, C. Taylor, T. Cootes, and T. Ahmed. **Automatic interpretation of human faces and hand gestures using flexible models**. *IEEE International Workshop on Face and Gesture recognition*, 1995. 32

[83] A. Lanitis, C. Taylor, T. Cootes, and T. Ahmed. **Automatic interpretation of human faces and hand gestures using flexible models**. *IEEE International Workshop on Face and Gesture recognition*, 1995. 33

[84] T.F. Cootes, G.J Edwards, and C.J. Taylor. **Active Appearance Models**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(6):681–685, Jun 2001. 33, 37, 142

[85] T.F. Cootes, G.J Edwards, and C.J. Taylor. **Statistical models of appearance for computer vision**. Technical report, World Wide Web Publication. 33, 37, 38

[86] C. Goodall. **Procrustes Methods in the Statistical Analysis of Shape**. *Journal of the Royal Statistical Society. Series B (Methodological)*, **53**(2):285–339, Oct. 2006. 34

[87] T. F. Cootes, G. J. Edwards, and C. J. Taylor. **Constrained Active Appearance Models**. Technical report, 2001. 37

[88] M. B. Stegmann. **Generative Interpretation of Medical Images**. *PhD thesis, In formatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2004. URL Awarded the Nordic Award for the Best Ph.D. Thesis in Image Analysis and Pattern Recognition in the years2003-2004 at SCIA05.* 38

[89] T. F. Cootes and C. J. Taylor. **An algorithm for tuning an active appearance model to new data**. *British Machine Vision Conference*, **3**:919–928, 2006. 40

[90] T. F. Cootes, G. J. Edwards, and C. J. Taylor. **A comparative evaluation of active appearance model algorithms**. *British Machine Vision Conference*, pages 680–689, 1998. 40

[91] T. F. Cootes and P. Kittipanya-ngam. **Comparing variations on the active appearance model algorithm**. *British Machine Vision Conference*, **2**:837–846, 2002. 40, 42

[92] X. Hou, S. Z. Li, H. Zhang, and Q. Cheng. **Direct appearance models**. *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, **1**:828–833, 2001. 40

[93] H. S. Lee and D. Kim. **Illumination-robust face recognition using tensor-based active appearance model**. *8th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–7, 2008. 40

[94] M. Alex, O. Vasilescu, and D. Terzopoulos. **Multilinear analysis of image ensembles: Tensorfaces**. *Proceedings of the European Conference on Computer Vision*, pages 447–460, 2002. 41

[95] C. Hu, R. Feris, and M. Turk. **Active wavelet networks for face alignment**. *British Machine Vision Conference*, 2003. 41

[96] I. Matthews and S. Baker. **Active Appearance Models Revisited**. *INTERNATIONAL JOURNAL OF COMPUTER VISION*, **60**:135–164, 2003. 42, 44, 79

[97] H. Mercier, J. Peyras, and P. Dalle. **Toward an efficient and accurate AAM fitting on appearance varying faces**. *Proc. 7th International Conference on Automatic Face and Gesture Recognition*, pages 363–368, 2006. 42

[98] S. Baker and I. Matthews. **Lucas-kanade 20 years on: A unifying framework**. *International Journal of Computer Vision*, **56**(3):221 – 255, 2004. 42

[99] J. A. Nelder and R. Mead. **A simplex method for function minimization**. *The Computer Journal*, **7**:308–313, 1964. 42

[100] Y. Aidarous, S. L. Gallou, A. Sattar, and R. Seguier. **Face alignment using active appearance model optimized by simplex**. *VISAPP 2007: Proceedings of the Second International Conference on Computer Vision Theory and Applications*, pages 231–236, 2007. 42

[101] Y. Aidarous, S. L. Gallou, and R. Seguier. **Simplex optimisation initialized by gaussian mixture for active appearance models**. *Proc. 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications*, pages 79–84, 2007. 42

[102] J. A. Paterson and A. W. Fitzgibbon. **3D head tracking using non-linear optimization**. In *Proceedings of the British Machine Vision Conference*, pages 63.1–63.10. BMVA Press, 2003. doi:10.5244/C.17.63. 42, 44

[103] D. Cristinacce and T. F. Cootes. **Facial feature detection and tracking with automatic template selection**. *Proc. 7th International Conference on Automatic Face and Gesture Recognition FGR 2006*, pages 429–434, 2006. 42

[104] D. E. Goldberg. *Genetic algorithms in search optimization and machine learning*. Addison-Wesley, Reading, 1989. 42

[105] C. McIntosh and G. Hamarneh. **Genetic algorithm driven statistically deformed models for medical image segmentation**. *ACM Workshop on Medical Applications of Genetic and Evolutionary Computation Workshop*, 2006. 43

[106] P. Ghosh and M. Mitchell. **Segmentation of medical images using a genetic algorithm**. *8th annual conference on Genetic and evolutionary computation*, pages 1171–1178, 2006. 43

[107] T. F. Cootes, K. N. Walker, and C. J. Taylor. **View-based active appearance models**. *International Conference on Automatic Face and Gesture Recognition*, pages 227–232, 2000. 43, 137

[108] T. Shan, B. C. Lovell, and S. Chen. **Face recognition robust to head pose from one sample image**. *18th International Conference on Pattern Recognition (ICPR06)*, pages 515–518, 2006. 43

[109] J. Sung and D. Kim. **Pose-robust facial expression recognition using view-based 2D+3D AAM**. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, **38**(4):852–866, 2008. 43, 44

[110] S. Z. Li, Y. Shuicheng, H. Zhang, and Q. Cheng. **Multi-view face alignment using direct appearance models**. *Proc. Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 324–329, 2002. 44

[111] S. V. Duhn, L. Yin, M. J. Ko, and T. Hung. **Multiple-view face tracking for modeling and analysis based on non-cooperative video imagery**. *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR07)*, pages 1–8, 2007. 44

[112] R. Ishiyama, M. Hamanaka, and S. Sakamoto. **An appearance model constructed on 3D surface for robust face recognition against pose and illumination variations**. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, **35**:326–334, Aug. 2005. 44

[113] S. Malassiotis and M. G. Strintzis. **Real-time head tracking and 3D pose estimation from range data**. *Proc. International Conference on Image Processing ICIP 2003*, **3**:59–62, 2003. 44

[114] J. Xiao, S. Baker, I. Matthews, and T. Kanade. **Real-time combined 2D+3D active appearance models**. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, **2**:535–542, 2004. 44, 79

[115] C. Hu, J. Xiao, I. Matthews, S. Baker, J. F. Cohn, and T. Kanade. **Fitting a single active appearance model simultaneously to multiple images**. *British Machine Vision Conference*, 2004. 44, 79

[116] S. Koterba, S. Baker, I. Matthews, C. Hu, J. Xiao, J. Cohn, and T. Kanade. **Multi-view AAM fitting and camera calibration**. *Proc. Tenth IEEE International Conference on Computer Vision ICCV 2005*, **1**:511–517, 2005. 44

[117] K. Ramnath, S. Koterba, J. Xiao, C. Hu, I. Matthews, S. Baker, J. Cohn, and T. Kanade. **Multi-view AAM fitting and construction**. *International Journal of Computer Vision*, **76**:183–204, February 2008. 44

[118] J. Xiao, J. Chai, and T. Kanade. **A closed-form solution to non-rigid shape and motion recovery**. *European Conference on Computer Vision*, pages 573–587, 2004. 44

[119] H. Hotelling. **Analysis of a complex of statistical variables into principal components**. *Journal of Educational Psychology*, **24**:417–441, 1933. 45

[120] J. Shlens. **A tutorial on principal component analysis**. *University of California at San Diego Systems Neurobiology Laboratory*, 2005. 46, 143

[121] M. E. Tipping and C. M. Bishop. **Probabilistic principal component analysis**. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **61**(3):611–622, 1999. 49, 51

[122] D. B. Rubin and D. T. Thayer. **EM algorithms for ML factor analysis**. *Psychometrika*, **47**(1):69–76, 1982. 50

[123] B. Boser, I. Guyon, and V. Vapnik. **A training algorithm for optimal margin classifiers**. *In D. Haussler, editor, Proc. COLT*, page 144152, 1992. 53

[124] M. M. Nordstrøm, M. Larsen, J. Sierakowski, and M. B. Stegmann. **The IMM Face Database - An Annotated Dataset of 240 Face Images**. Technical report. 57

[125] T. Sim, S. Baker, and M. Bsat. **The CUM pose illumination and expression database**. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **25**(12):1615–1618, 2003. 60, 86, 97, 127, 153, 162

[126] A. Tropp Joel and C. G. Anna. **Signal recovery from random measurements via orthogonal matching pursuit**. *IEEE Transactions on Information Theory*, **53**(12):4655–4666, 2007. 73, 93, 94, 96, 151

[127] R. Rubinstein, T. Peleg, and M. Elad. **Analysis K-SVD: A dictionary-learning for the analysis sparse model**. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5405–5408, 2012. 73, 94, 95, 151

[128] J. Tu, T. Huang, and Y. Xiong. **Calibrating head pose estimation in vidoes for meeting room event analysis**. In *IEEE International Conference on Image Processing (ICIP)*, pages 3193–3196, 2006. 74

[129] E. Murphy-Chutorian, A. Doshi, and M. M. Trivedi. **Head pose estimation for driver assistance systems: A robust algorithm and experimental evaluation**. In *IEEE Intelligent Transportation Systems Conference ITSC 2007*, pages 709–714, 2007. 74, 75

[130] S. Ohayon and E. Rivlin. **Robust 3D head tracking using camera pose estimation**. In *International Conference Pattern Recognation (ICPR)*, pages 1063–1066, 2006. 74

[131] E. Murphy-Chutorian and M. Trivedi. **Head Pose Estimation in Computer Vision: A Survey**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**(4):607–626, 2009. 74

[132] D. J. Beymer. **Face recognition under varying pose**. In *Proc. IEEE Conference Computer Vision and Pattern Recognition 1994. Proceedings CVPR'94.*, pages 756–761. 75

[133] S. Niyogi and W. T. Freeman. **Example-based head tracking**. In *Proceedings of the Second International Conference on. IEEE Automatic Face and Gesture Recognition*, pages 374–378, 1996. 75

[134] J. Ng and S. Gong. **Composite support vector machines for detection of faces across views and pose estimation**. *Image and Vision Computing*, **20**:359–368, 2002. 75

[135] J. Ng and S. Gong. **Multi-view face detection and pose estimation using a composite support vector machine across the view sphere**. In *Proceedings. International Workshop on. IEEE Workshop Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 14–21, 1999. 75

[136] Y. Li, S. Gong, and J. Sherrah. **Support vector machine based multi-view face detection and recognition**. *Image and Vision Computing*, **22**:413–427, 2004. 75

[137] Y. Li, S. Gong, and H. Liddell. **Support vector regression and classification based multi-view face detection and recognition Automatic Face and Gesture Recognition**. In *Proceedings Fourth IEEE International Conference on. IEEE*, pages 300–305, 2000. 75

[138] Y. Ma, Y. Konishi, and K. Kinoshita. **Sparse bayesian regression for head pose estimation**. In *IEEE 18th International Conference on Pattern Recognition ICPR 2006*, pages 507–510, 2006. 75

[139] H. Moon and M. L. Miller. **Estimating facial pose from a sparse representation**. In *IEEE 2004 International Conference on Image Processing ICIP'04*, pages 75–78, 2004. 75

[140] C. M. Bishop. *Neural networks for pattern recognition.* 1995. 76

[141] E. Seemann, K. Nickel, and R. Stiefelhagen. **Head pose estimation using stereo vision for human-robot interaction**. In *Proceedings. Sixth IEEE International Conference on. Automatic Face and Gesture Recognition*, pages 626–631, 2004. 76

[142] R. Stiefelhagen, J. Yang, and A. Waibel. **Modeling focus of attention for meeting indexing based on multiple cues**. *IEEE Transactions on Neural Networks*, **13**:928–938, 2002. 76

[143] R. Rae and H. J. Ritter. **Recognition of human head orientation based on artificial neural networks**. *IEEE Transactions on Neural Networks*, **9**:257–265, 1998. 76

[144] J. Bruske, E. Abraham-Mumm, J. Pauli, and G. Sommer. **Head-pose estimation from facial images with subspace neural networks**. In *In Proc. of Int. Neural Network and Brain Conference*, pages 528–531, 1998. 76

[145] V. Krüger and G. Sommer. **Gabor wavelet networks for efficient head pose estimation**. *Image and vision computing*, **20**:665–672, 2002. 76

[146] Y. LeCun, L. Bottou, and Y. Bengio. **Gradient-based learning applied to document recognition**. In *Proceedings of the IEEE*, **86**, pages 2278–2324. 76

[147] M. Osadchy, Y. L. Cun, and Miller M. L. **Synergistic face detection and pose estimation with energy-based models**. *The Journal of Machine Learning Research*, **8**:1197–1215, 2007. 76

[148] S. J. McKenna and S. Gong. **Real-time face pose estimation**. *Real-Time Imaging*, **4**:333–347, 1998. 77

[149] J. Sherrah, S. Gong, and E. J. Ong. **Understanding pose discrimination in similarity space**. In *10th British Machine Vison Conference*, pages 523–532, 1999. 77

[150] J. Sherrah, S. Gong, and E. J. Ong. **Face distributions in similarity space under varying head pose**. *Image and Vision Computing*, **19**:807–819, 2001. 77

[151] J. Wu and M. M. Trivedi. **A two-stage head pose estimation framework and evaluation**. *Pattern Recognition*, **41**:1138–1158, 2008. 77

[152] S. Srinivasan and K. L. Boyer. **Head pose estimation using view based eigenspaces**. In *IEEE Proceedings. 16th International Conference on Pattern Recognition*, pages 302–305. IEEE, 2002. 77

[153] S. Z. Li, Q. Fu, and L. Gu. **Kernel machine based learning for multi-view face detection and pose estimation**. In *Proceedings. Eighth IEEE International Conference on Computer Vision ICCV 2001*, pages 674–679, 2001. 77

[154] B. Ma, W. Zhang, and S. Shan. **Robust head pose estimation using LGBP**. In *18th International Conference on Pattern Recognition, 2006. ICPR 2006.*, pages 512–515, 2006. 77

[155] L. Chen, L. Zhang, and Y. Hu. **Head pose estimation using fisher manifold learning**. In *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures*. IEEE Computer Society, 2003. 77

[156] B. Raytchev, I. Yoda, and K. Sakaue. **Head pose estimation by nonlinear manifold learning**. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, pages 462–466. IEEE, 2004. 77, 78

[157] J. B. Tenenbaum, V. De Silva, and J. C. Langford. **A global geometric framework for nonlinear dimensionality reduction**. *Science*, **290**:2319–2323, 2000. 77

[158] S. T. Roweis and L. K. Saul. **Nonlinear dimensionality reduction by locally linear embedding**. *Science*, **290**:2323–2326, 2000. 77

[159] M. Belkin and P. Niyogi. **Laplacian eigenmaps for dimensionality reduction and data representation**. *Neural computation*, **15**:1373–1396, 2003. 77

[160] V. N. Balasubramanian, J. Ye, and S. Panchanathan. **Biased manifold embedding: A framework for person-independent head pose estimation**. In *IEEE Conference onComputer Vision and Pattern Recognition CVPR'07.*, pages 1–7, 2007. 78

[161] Y. Fu and T. S. Huang. **Graph embedded analysis for head pose estimation**. In *7th International Conference on Automatic Face and Gesture Recognition*, pages 6–8, 2006. 78

[162] X. He, S. Yan, and Y. Hu. **Learning a locality preserving subspace for visual recognition**. In *Proceedings. Ninth IEEE International Conference on Computer Vision*, pages 385–392, 2003. 78

[163] N. Hu, W. Huang, and S. Ranganath. **Head pose estimation by non-linear embedding and mapping**. In *IEEE International Conference on Image Processing ICIP 2005*, pages 342–345, 2005. 78

[164] S. Yan, Z. Zhang, and Y. Fu. 78

[165] Z. Li, Y. Fu, and J. Yuan. **Query driven localized linear discriminant models for head pose estimation**. In *2007 IEEE International Conference on Multimedia and Expo*, pages 1810–1813, 2007. 79

[166] Z. Gui and C. Zhang. **3D head pose estimation using non-rigid structure-from-motion and point correspondence**. In *Proc. IEEE Region 10 Conference 2006*, pages 1–3. IEEE, 2006. 79

[167] S. Baker, I. Matthews, J. Xiao, R. Gross, T. Kanade, and T. Ishikawa. **Real-time non-rigid driver head tracking for driver mental state estimation**. In *11th World Congress Intelligent Transportation Systems*, 2004. 79

[168] D. L. Donoho. **For most large underdetermined systems of linear equations the minimal L1-norm solution is also the sparsest solution**. *Communications on pure and applied mathematics*, **59**(6):797–829, 2006. 81, 83, 89

[169] H. L. Taylor, S. C. Banks, and J. F. McCoy. **Deconvolution with the L1 norm**. *Geophysics*, **44**:3952, 1979. 81

[170] J. F. Claerbout and F. Muir. **Robust modeling with erratic data**. *Geophysics*, **38**:826844, 1973. 81

[171] F. Santosa and W. W. Symes. **Linear inversion of band-limited reflection seismograms**. *SIAM Journal on Scientific and Statistical Computing*, **7**:13071330, 1986. 81

[172] D. L. Donoho and P. B. Stark. **Uncertainty principles and signal recovery**. *SIAM Journal on Scientific and Statistical Computing*, **49**:906931, 1989. 81

[173] D. L. Donoho and B. F. Logan. **Signal recovery and the large sieve**. *SIAM Journal on Applied Mathematics*, **52**:577591, 1992. 81

[174] L. I. Rudin, S. Osher, and E. Fatemi. **Nonlinear total variation based noise removal algorithms**. *Physica D: Nonlinear Phenomena*, **60**:259268, 1992. 81

[175] P. Blomgren and T.F. Chan. **Color TV: total variation methods for restoration of vector-valued images**. *IEEE Transaction on Image Processing*. 81

[176] L. Vandenberghe, S. Boyd, and A. El Gamal. **Optimal wire and transistor sizing for circuits with non-tree topology**. In *Proceedings of the 1997 IEEE/ACM International Conference on Computer Aided Design*, page 252259. IEEE, 1997. 81

[177] Boyd S. El Gamal A. Vandenberghe, L. **Optimizing dominant time constant in RC circuits**. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **17**:110–125, 1998. 81

[178] A. Hassibi, J. How, and S. Boyd. **Low-authority controller design via convex optimization**. *Journal of guidance, control, and dynamics*, **22**:862872, 1999. 81

[179] M. Dahleh and I. Diaz-Bobillo. *Control of Uncertain Systems: A Linear Programming Approach*. Prentice Hall; 1st edition, June 1995. 81

[180] M. Lobo, M. Fazel, and S. Boyd. **Portfolio optimization with linear and fixed transaction costs**. *Annals of Operations Research*, **152**:341–365, 2006. 81

[181] A. Ghosh and S. Boyd. **Growing well-connected graphs**. In *Proceedings of the 45th IEEE Conference on Decision and Control*, page 66056611, December 2006. 81

[182] J. Sun, S. Boyd, L. Xiao, and P. Diaconis. **The fastest mixing Markov process on a graph and a connection to a maximum variance unfolding problem**. *SIAM Review*, **48**:681699, 2006. 81

[183] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004. 81

[184] S. J. Kim, K. Koh, S. Boyd, and D. Gorinevsky. **L1 trend filtering**. *SIAM Review*, **51**:339–360, 2008. 81

[185] D. L. Donoho and X. Huo. **Uncertainty principles and ideal atomic decomposition**. *IEEE Transactions on Information Theory*, **47**:28452862, 2001. 81

[186] M. Elad and A. M. Bruckstein. **A generalized uncertainty principle and sparse representation in pairs of bases**. *IEEE Transactions on Information Theory*, **48**:25582567, 2002. 82

[187] J. A. Tropp. **Just relax: convex programming methods for identifying sparse signals in noise**. *IEEE Transactions on Information Theory*, **52**:10301051, 2006. 82

[188] E. J. Candès and T. Tao. **Near optimal signal recovery from random projections: Universal encoding strategies**. *IEEE Transactions on Information Theory*, **52**:54065425, 2006. 82

[189] D. Donoho. **Compressed sensing**. *IEEE Transactions on Information Theory*, **52**:1289–1306, 2006. 82, 85

[190] D. L. Donoho and J. Tanner. **Counting faces of randomly-projected polytopes when the projection radically lowers dimension**. *Journal of the American Mathematical Society*, **22**:1–53, 2009. 82

[191] E. J. Candès, J. Romberg, and T. Tao. **Stable signal recovery from incomplete and inaccurate measurements**. *Communications on pure and applied mathematics*, **59**:12071223, 2006. 82

[192] D. Donoho and Y. Tsaig. **Extensions of compressed sensing**. *Signal Processing*, **86**:533548, 2006. 82

[193] D. Takhar, V. Bansal, M. Wakin, M. Duarte, D. Baron, K. F. Kelly, and R. G. Baraniuk. **A compressed sensing camera: New theory and an implementation using digital micromirrors**. In *Proceedings of Comp. Imaging IV at SPIE Electronic Imaging*, January 2006. 82

[194] E. J. Candès, J. Romberg, and T. Tao. **Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information**. *IEEE Transactions on Information Theory*, **52**:489509, 2006. 82

[195] M. Lustig, D. Donoho, and J.M. Pauly. **Sparse MRI: The application of compressed sensing for rapid MR imaging**. *Magnetic resonance in medicine*, **58**:1182–1195, 2007. 82

[196] E. J. Candes and T. Tao. **Decoding by linear programming**. *IEEE Transactions on Information Theory*, **51**:42034215, 2005. 82

[197] E. J. Candes and Randall P. A. **Highly robust error correction byconvex programming**. *IEEE Transactions on Information Theory*, **54**:2829–2840, 2008. 82

[198] Compressive wireless sensing. **Bajwa, W. and Haupt, J. and Sayeed, A. and Nowak, R.** In *Proceedings of Fifth International Conference on Information Processing in Sensor Networks*, page 134142, 2006. 82

[199] D. Baron, M.B. Wakin, M.F. Duarte, S. Sarvotham, and R.G. Baraniuk. **Distributed compressed sensing**. Technical report, Department of Electrical and Computer Engineering Rice University, 2005. 82

[200] S. Chen, S. A. Billings, and W. Luo. **Orthogonal least squares methods and their application to non-linear system identification**. *International Journal of control*, **50**:18731896, 1989. 82

[201] G. Davis, S. Mallat, and Z. Zhang. **Adaptive time-frequency decompositions**. *Optical Engineering*, **33**:2183–2191, 1994. 82

[202] E. D. Livshitz. **On the optimality of the Orthogonal Greedy Algorithm for μ-coherent dictionaries**. *Journal of Approximation Theory*, **164**(5):668–681, 2012. 83

[203] J. A. Tropp. **Just relax: Convex programming methods for identifying sparse signals in noise**. *IEEE Transactions on Information Theory*, **52**(3):1030–1051, 2006. 83

[204] V. N. Temlyakov. **Weak greedy algorithms**. *Advances in Computational Mathematics*, **12**(2-3):213–227, 2000. 83

[205] M. S. Lewicki and B. A. Olshausen. **A probabilistic framework for the adaptation and comparison of image codes**. *Journal of the Optical Society of America A: Optical, Image Science, and Vision*, **16**:15871601, 1999. 83

[206] B. A. Olshausen and D. J. Field. **Natural image statistics and efficient coding**. *Network: computation in neural systems*, **7**:333–339, 1996. 83

[207] J. A. Tropp. **Greed is good: Algorithmic results for sparse approximation**. *IEEE Transactions on Information Theory*, **50**:22312242, Oct. 2004. 83, 84

[208] D. L. Donoho and M. Elad. **Optimally sparse representation in general (non-orthogonal) dictionaries via L1 minimization**. In *Proc. Natl Acad. Sci. USA 100 2197202*, 2003. 83

[209] D. L. Donoho, M. Elad, and V. Temlyakov. **Stable recovery of sparse overcomplete representations in the presence of noise**. *IEEE Transactions on Information Theory*, **52**:6–18, 2006. 83

[210] J. A. Tropp. **Just relax: Convex programming methods for subset selection and sparse approximation**. *IEEE Transactions on Information Theory*, **52**:1030–1051, 2006. 83

[211] Julien Mairal, Guillermo Sapiro, and Michael Elad. **Learning multiscale sparse representations for image and video restoration**. Technical report, SIAM MMS, 2007. 84

[212] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. **Supervised Dictionary Learning**, 2008. 84, 85

[213] Optimal sparse representation in general (nonorthogoinal) dictionaries via l1 minimization. **Donoho, D. L. and Elad, M.** In *Proc. of the National Academy of Sciences*, **100**, page 21972202, March 2003. 84

[214] M. Elad. **Optimized projections for compressed-sensing**. *IEEE Transactions on Signal Processing*, **55**:56955702, Dec. 2007. 84

[215] K. Schnass and P. Vandergheynst. **Dictionary preconditioning for greedy algorithms**. *IEEE Transactions on Signal Processing*, **56**:19942002, 2008. 84

[216] J. M. Duarte-Carvajalino and G. Sapiro. **Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization**. *IEEE Transactions Image Processing*, **18**:1395–1408, 2009. 84, 85

[217] Julien Mairal, Michael Elad, and Guillermo Sapiro. **Sparse representation for color image restoration**. *IEEE Transactions on Image Processing*, **17**(1):53–69, 2008. 84

[218] S. P. Awate and R. T. Whitaker. **Unsupervised, information-theoretic, adaptive image filtering for image restoration**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(3):364–376, 2006. 84

[219] A. Buades, B. Coll, and J. M. Morel. **A review of image denoising algorithms, with a new one**. *Multiscale Modeling Simulation*, **4**(2):490–530, 2005. 84

[220] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. **Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space**. In *IEEE International Conference on Image Processing, 2007. ICIP 2007.*, **1**, pages I–313. IEEE, 2007. 84

[221] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. **Discriminative learned dictionaries for local image analysis**. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008.*, pages 1–8. IEEE, 2008. 84, 85

[222] 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 (PAMI)*, **31**(2):210–227. 84, 86, 88, 109, 133, 158

[223] G. Peyre. **Sparse modeling of textures**. *Journal of Mathematical Imaging and Vision*, **34**:17–31, 2009. 84

[224] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. **Self-taught learning: transfer learning from unlabeled data**. In *Proceedings of the 24th international conference on Machine learning*, page 759766, 2007. 84

[225] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. **Discriminative sparse image models for class-specific edge detection and image interpretation**. *In European Conference on Computer Vision*, pages 43–56, 2008. 84, 85

[226] F. Rodriguez and G. Sapiro. **Sparse representations for image classification: Learning discriminative and reconstructive non-parametric dictionaries**. Technical report, DTIC Document, 2008. 84, 85

[227] E. J. Candes. **Compressive sampling**. In *Proceedings oh the International Congress of Mathematicians: Madrid, August 22-30, 2006: invited lectures*, pages 1433–1452, 2006. 85

[228] P. M. Hall, D. Marshall, and R. R. Martin. **Incremental Eigenanalysis for Classification**. In *British Machine Vision Conference (BMVC)*, pages 286–295, 1998. 87, 88

[229] P. Viola and M. Jones. **Robust Real-time Object Detection**. *International Journal of Computer Vision*, 2001. 104, 105, 154

[230] C. Cortes and V. Vapnik. **Support-vector networks**. *Machine learning*, **20**(3):273–297, 1995. 108, 158

[231] T. Cover and P. Hart. **Nearest neighbor pattern classification**. *IEEE Transactions on Information Theory*, **13**(1):21–27, 1967. 108, 158

[232] M.S. Sarfraz and O. Hellwich. **Head Pose Estimation in Face Recognition Across Pose Scenarios**. In *VISAPP (1)'08*, pages 235–242, 2008. 109, 158

[233] Y. Moses, Y. Adini, and S. Ullman. **Face recognition: The problem of compensating for changes in illumination direction**. In *Computer VisionECCV'94*, pages 286–296. Springer, 1994. 112

[234] G. Aggarwal and R. Chellappa. **Face recognition in the presence of multiple illumination sources**. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005.* 112

[235] R. Basri and D. W Jacobs. **Lambertian reflectance and linear subspaces**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(2):218–233, 2003. 112

[236] N. Gudur and V. Asari. **Gabor wavelet based modular PCA approach for expression and illumination invariant face recognition**. In *Applied Imagery and Pattern Recognition Workshop, 2006. AIPR 2006. 35th IEEE*, pages 13–13. IEEE, 2006. 112, 114

[237] W. S. Chen, P. C. Yuen, J. Huang, and J. Lai. **Face classification based on shannon wavelet kernel and modified fisher criterion**. In *7th International Conference on Automatic Face and Gesture Recognition, 2006. FGR 2006.*, pages 467–474. IEEE, 2006. 112, 114

[238] S. Srisuk and A. Petpon. **A gabor quotient image for face recognition under varying illumination**. In *Advances in Visual Computing*, pages 511–520. Springer, 2008. 112, 133

[239] T. Zhang, Y. Y. Tang, B. Fang, Z. Shang, and X. Liu. **Face recognition under varying illumination using gradientfaces**. *IEEE Transactions on Image Processing*, **18**(11):2599–2606, 2009. 112, 118

[240] T. Ojala, M. Pietikainen, and D. Harwood. **A comparative study of texture measures with classification based on featured distributions**. *Pattern recognition*, **29**(1):51–59, 1996. 112, 117

[241] R. Zabih and J. Woodfill. **Non-parametric local transforms for computing visual correspondence**. In *Computer VisionECCV'94*, pages 151–158. Springer, 1994. 112, 117

[242] S. Z. Li and A. K. Jain. *Handbook of face recognition*. springer, 2011. 113

[243] R. Basri and D. W. Jacobs. **Lambertian reflectance and linear subspaces**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(2):218–233, 2003. 113

[244] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. **From few to many: Illumination cone models for face recognition under variable lighting and pose**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(6):643–660, 2001. 114

[245] S. Chen, B. C. Lovell, and T. Shan. **Robust adapted principal component analysis for face recognition**. *International Journal of Pattern Recognition and Artificial Intelligence*, **23**(03):491–520, 2009. 114

[246] K. K. Sung and T. Poggio. **Example-based learning for view-based human face detection**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**(1):39–51, 1998. 116

[247] J. Ruiz-del Solar and P. Navarrete. **Eigenspace-based face recognition: a comparative study of different approaches**. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, **35**(3):315–325, 2005. 116

[248] D. J. Jobson, Z. Rahman, and G. A. Woodell. **Properties and performance of a center/surround retinex**. *IEEE Transactions on Image Processing*, **6**(3):451–462, 1997. 116

[249] E. H. Land and J. J. McCann. **Lightness and retinex theory**. *Journal of the Optical society of America*, **61**(1):1–11, 1971. 116

[250] R. Ramamoorthi. **Analytic pca construction for theoretical analysis of lighting variability in images of a lambertian object**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(10):1322–1333, 2002. 116

[251] H. Wang, S. Z. Li, and Y. Wang. **Face recognition under varying lighting conditions using self quotient image**. In *Proceedings of Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 819–824. IEEE, 2004. 117

[252] T. Ahonen, A. Hadid, and M. Pietikäinen. **Face recognition with local binary patterns**. In *Computer Vision-ECCV 2004*, pages 469–481. Springer, 2004. 118

[253] A. R. Webb. *Statistical pattern recognition*. Wiley. com, 2003. 121, 122, 161

[254] A. Beck and M. Teboulle. **A fast iterative shrinkage-thresholding algorithm for linear inverse problems**. *SIAM Journal on Imaging Sciences*, **2**(1):183–202, 2009. 124

[255] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. **Least angle regression**. *The Annals of statistics*, **32**(2):407–499, 2004. 124

[256] L. Zhang, W. Dong, D. Zhang, and G. Shi. **Two-stage image denoising by principal component analysis with local pixel grouping**. *Pattern Recognition*, **43**(4):1531–1549, 2010. 129, 163

[257] N. Ahmed, T. Natarajan, and K. R. Rao. **Discrete cosine transform**. *IEEE Transactions on Computers*, **100**(1):90–93, 1974. 133

[258] M. Shensa. **The discrete wavelet transform: wedding the a trous and Mallat algorithms**. *IEEE Transactions on Signal Processing*, **40**(10):2464–2482, 1992. 133

[259] T. Ahonen, A. Hadid, and M. Pietikainen. **Face description with local binary patterns: Application to face recognition**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(12):2037–2041, 2006. 133

[260] D. Geiger, T. Liu, and M. J. Donahue. **Sparse representations for image decompositions**. *International Journal of Computer Vision*, **33**(2):139–156, 1999. 137

[261] R. Zass and A. Shashua. **Nonnegative sparse PCA**. In *Advances in Neural Information Processing Systems*, pages 1561–1568, 2006. 137