ABSTRACT

Title of dissertation: RECOGNITION OF FACES FROM SINGLE AND MULTI-VIEW VIDEOS

Ming Du, Doctor of Philosophy, 2014

Directed by: Professor Rama Chellappa
Department of Electrical and Computer Engineering

Face recognition has been an active research field for decades. In recent years, with videos playing an increasingly important role in our everyday life, video-based face recognition has begun to attract considerable research interest. This leads to a wide range of potential application areas, including TV/movies search and parsing, video surveillance, access control etc. Preliminary research results in this field have suggested that by exploiting the abundant spatial-temporal information contained in videos, we can greatly improve the accuracy and robustness of a visual recognition system. On the other hand, as this research area is still in its infancy, developing an end-to-end face processing pipeline that can robustly detect, track and recognize faces remains a challenging task. The goal of this dissertation is to study some of the related problems under different settings.

We address the video-based face association problem, in which one attempts to extract face tracks of multiple subjects while maintaining label consistency. Traditional tracking algorithms have difficulty in handling this task, especially when challenging nuisance factors like motion blur, low resolution or significant camera motions are present. We demonstrate that contextual features, in addition to face appearance itself, play an important role in this case. We propose principled methods to combine multiple features using Conditional Random Fields and Max-Margin Markov networks to infer labels for the detected faces. Different from many existing

approaches, our algorithms work in online mode and hence have a wider range of applications. We address issues such as parameter learning, inference and handling false positves/negatives that arise in the proposed approach. Finally, we evaluate our approach on several public databases.

We next propose a novel video-based face recognition framework. We address the problem from two different aspects: To handle pose variations, we learn a Structural-SVM based detector which can simultaneously localize the face fiducial points and estimate the face pose. By adopting a different optimization criterion from existing algorithms, we are able to improve localization accuracy. To model other face variations, we use intra-personal/extra-personal dictionaries. The intra-personal/extra-personal modeling of human faces has been shown to work successfully in the Bayesian face recognition framework. It has additional advantages in scalability and generalization, which are of critical importance to real-world applications. Combining intra-personal/extra-personal models with dictionary learning enables us to achieve state-of-arts performance on unconstrained video data, even when the training data come from a different database.

Finally, we present an approach for video-based face recognition using camera networks. The focus is on handling pose variations by applying the strength of the multi-view camera network. However, rather than taking the typical approach of modeling these variations, which eventually requires explicit knowledge about pose parameters, we rely on a pose-robust feature that eliminates the needs for pose estimation. The pose-robust feature is developed using the Spherical Harmonic (SH) representation theory. It is extracted using the surface texture map of a spherical model which approximates the subject's head. Feature vectors extracted from a video are modeled as an ensemble of instances of a probability distribution in the Reduced Kernel Hilbert Space (RKHS). The ensemble similarity measure in RKHS improves both robustness and accuracy of the recognition system. The proposed approach outperforms traditional algorithms on a multi-view video database collected using a camera network.

RECOGNITION OF FACES FROM SINGLE AND MULTI-VIEW VIDEOS

by

Ming Du

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2014

Advisory Committee:
Professor Rama Chellappa, Chair/Advisor
Professor K. J. Ray Liu
Professor Larry Davis
Professor David Jacobs
Professor Min Wu

Dedication

To my wife and my parents

Acknowledgments

First and foremost, I owe my deepest gratitude to my advisor, Professor Rama Chellappa, for his continuous guidance and mentoring over the past several years. He has supported and encouraged me to work on research topics that I am interested in. I always enjoyed the fruitful and inspiring discussion with him. His dedication to work, his caring attitude towards his students and his positive attitude have not only made him an extraordinary research advisor to me, but also a great example in life that I can learn from.

I would also like to thank Prof. K.J. Ray Liu, Prof. Min Wu, Prof. Larry Davis and Prof. David Jacobs for serving on my dissertation committee and providing valuable feedback. The thesis wouldnt have taken the shape without the feedback, encouragement and support of my fellow group members, among whom I would like to especially thank Aswin Sankaranarayanan, Qiang Qiu, Jaishanker Pillai, Pavan Turaga, Vishal Patel, Kaushik Mitra, Tao Wu, Jie Ni, Ruonan Li, Seo Naotoshi, Yi-Chen Chen, Sima Taheri, Raghuram Gopalan, Mahesh Ramachandran and Ming-Yu Liu. I am particularly grateful to my manager at A9.com Inc., Marle Christophe, for supporting me while I was finishing this thesis. I thank Daozheng Chen for sharing his insightful observations in research with me.

I am obliged to my friends Yongle Wu, Haipeng An, Qi Hu, Bing Shi, Yongqiang Wang, Beibei Wang, Wei-Hong Chuang, Gang Bai, Tingting Liu, Jun-Cheng Chen, Wenjun Lu, Shanshan Zheng, Jingting Zhou, Jingjing Zheng, Wei Meng, Huimin Guo, Shuo Huang, Huy Tho Ho, Garrett Warnell, Nazre Batool and Feng Zhao.

They have made my life at UMD a wonderful experience. I am thankful to my officemate Xavier Gibert Serra and roommate Hua Chen, Shihua Wen, Jialin Tao for their friendliness and support. I would also like to acknowledge the crucial help and support from Janice Perrone, Melanie Prange, Arlene Schenk, ECE staff and UMIACS computing staff, whose efforts guarantee all administrative, operational and logistical work processes to run smoothly.

I am truly blessed to have my wife, Dapeng Li, in my life. Without her constant support and encouragement, this dissertation would have been a distant dream. Words would not be enough to express my gratitude to my parents. They have sacrificed so much for me, and no matter what happened they have always been there for me. I am grateful to my sister and my brother-in-law. It is my fortune be in a family with them. I am deeply indebted to my advisor at Ryerson University, Prof. Ling Guan, who is like a family member to me.

Lastly, thank God!

I apologize to those I've inadvertently left out. This dissertation would not have been possible without you.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The general face recognition problem can be defined as identifying faces in a query database (probe) given a stored database of labeled faces (gallery). The fact that humans can accomplish this task so well has encouraged researchers to develop automatic solutions to the problem. However, in spite of the intense research activities in face recognition performed over several decades [1] and the significant advances that have been achieved, the performance of existing algorithms is not good enough for deployment.

In recent years, researchers have started to consider the role of videos in automatic face recognition. This is partly because video arises naturally in many important applications, such as surveillance and access control. But the more important reason is that videos contain more information than still images and can provide spatial-temporal characteristics of patterns for improved recognition performance. One can think of many reasons for potentially improved performance when videos are used. It could either be due to the availability of algorithms that infer 3D information from multiple views, or due to the evidence accrual process that comes into play when multiple frames of an object are processed, or being able to learn and recognize the facial dynamics. Following these possible paths, many approaches have been proposed, making video-based face recognition an active research field.

Although previous research efforts have suggested that videos do help to improve face recognition accuracy, many challenges still remain, among which the most important ones are discussed next. In a general sense, "video-based face recognition" refers to the whole end-to-end processing pipeline which also includes the face tracking or face detection module. Reliable extraction of a face track is a difficult task, yet it is essential to the recognition module. In many traditional face recognition evaluation databases, both gallery and probes are still images taken in a controlled environment. In contrast, face appearance often changes dramatically in a video, due to complicated interactions of many factors such as pose, illumination and expression. As such, we either need to estimate the states of these nuisance factors and then normalize them, or design invariants for these variations. Regarding the large volume of visual data that need to be processed, we must manage to handle the "curse of dimensionality" problem. Furthermore, a successful video-based face recognition algorithm should go beyond the naive approach that directly applies still image-based face recognition algorithms to individual frames. Coming up with a representation that can efficiently exploit the spatial-temporal information of videos is an important task to be addressed.

In this dissertation, we will investigate three different but closely-connected sub-problems of video-based face recognition, namely face association from multi-person videos, single-view face recognition using intra/extra-personal dictionary and multi-view face recognition in camera networks. Below, we will give a brief introduction for each of them.

1.1 Face Association from Videos

As the first stage in the processing pipeline, face extraction from video is the key step in any video-based face recognition system. Traditionally, we rely on tracking algorithms to accomplish this task. However, when we are confronted with unconstrained videos, crowded scene and camera motions often pose great challenges to tracking algorithms, not to mention that these algorithms inherently suffer from drifting errors. In view of this fact, we choose to adopt a "tracking by detection" scheme, which in turn requires a method to associate detected faces of the same person across frames despite the frequent presence of false detections.

Our proposed approach for face association is partially inspired by human's cognitive psychology under similar circumstances. When identifying a person, we often make our judgement not only based on his/her facial features, but also exploit contextual information such as clothes, hair style etc. Although face association is not exactly equivalent to an identification problem, this phenomenon does provide us a hint for leveraging contextual features to group faces. In this work, the features employed include clothing, relative positions and scale and uniqueness constraints. By casting the association problem in a probabilistic graphical model framework, we encode these features using unary and pairwise potential functions.

By noticing that face association has a structural output, we propose two principled methods to integrate the contextual features. They are based on Conditional Random Fields (CRFs) and Max-Margin Markov (M^3) networks, respectively. The two approaches share the same underlying graphical model and the same set of fea-

ture functions, but differ in the functions being optimized. As the graph structure is highly-connected, we present approximate inference techniques to optimize the objective functions. In contrast with most existing multi-target tracking algorithms, this proposed algorithm works in an online mode, which provides more choices in applications.

In face association, as face tracks are built in a bottom-up fashion from detection responses, it is important to address false positives, false negatives and subjects entering/leaving the scene. To this end, we explicitly model a null state using logistic regression to account for both novel faces and false positives. We also propose schemes to recover the false negatives based on a sampling strategy. The resulting framework is capable of handling dynamically a varying graph structure and working with noisy detection outputs.

1.2 Single-View Video-Based Face Recognition

In this part, we look into the video-to-video face recognition problem under single-view settings. Our work relies on the concept of intra-personal/extra-personal face variations, first proposed in a Bayesian face recognition framework [2]. To be more specific, we assume that the difference between any pair of human face images falls into one of two possible categories: those purely caused by nuisance factors such as illuminations, pose, expression etc., and those caused by different identities. However, we further separate pose variations from other factors by only investigating the differences taken at the same face pose, as we believe that they often obscure

the boundary between intra-personal and extra-personal classes.

We take a dictionary learning approach in view of the recent success achieved by sparse coding in many computer vision applications. Since the traditional dictionary learning methods are not directly related to classification tasks, we follow the Label-Consistent K-SVD (LC-KSVD) algorithm [3] to jointly learn a shared intra-personal/extra-personal dictionary and a discriminative projection matrix for each pose group. The linear transformed sparse codes are used for recognition. To efficiently exploit the high volume video data, we fit a Dirichlet process Gaussian mixture model to each video. The model exempts us from having to specify a fixed number of clusters and compresses the video to a set of representative frames which are used in training and testing.

To perform face alignment against translation and in-plane rotations, we propose a structural SVM-based face fiducial point detector. It serves an additional crucial purpose: provide pose estimation outputs that can be employed to construct pose-specific dictionaries. The detector adopts a tree mixture model that enables us to simultaneously obtain feature locations and discretized face pose. We set the optimization criterion in such a way that emphasis is placed on localization accuracy. Results of our experiments demonstrate that the proposed detector is able to work robustly "in the wild".

The proposed single-view video-based face recognition framework not only produces state-of-arts results on public databases, but also has attractive properties in scalability and generalization. Irrespective of the size of database, we always only consider a two-class classification problem. The algorithm can be readily applied to

the video-based face verification protocol and gracefully handle the cross-database recognition problem.

1.3 Multi-View Video-Based Face Recognition

In this part, we turn our attention from single-view settings to multi-view settings. Camera networks have become increasingly prevalent in surveillance environments, providing effective means for handling pose variations in face recognition. Cooperation among multiple cameras can increase the chance of capturing a targeted face in a favorable frontal pose. However, previous multi-view face recognition algorithms, irrespective of whether they use a camera network or not, require a pose estimation or model registration step. Results of such a step are vital to the recognition performance as they provide a common reference frame to compare face appearances. Unfortunately, neither of the two problems is easy to solve, and hence the desired registration or estimation accuracy can seldom be achieved. It is thus desirable that we avoid both of them as much as possible. This motivates us to investigate whether pose invariants exist when we attempt to recognize faces using camera networks.

In physics, spherical harmonics (SH) theory has been well known for its application in the study of electrons. Basri and Jacobs [4] introduced it for modeling the reflectance functions that arise in face recognition problems. We employ the SH theory to analyze the appearance of human face and propose pose-robust features based on spherical harmonics. To be specific, we approximate the human head with

a spherical model and construct a surface texture map from multi-view images. The energy dispersion of the SH coefficients of the texture map remains constant against rotations. Using this property, we are able to bypass the pose estimation step in multi-view face recognition application.

Our video tracking module combines robust visual features to locate a human head and provides a continuous supply of head appearance to the texture mapping module. As for video-level recognition, we treat the set of the SH energy dispersion features extracted from all the frames in a video as an ensemble and project them onto the reproducing kernel Hilbert space (RKHS). Ensemble similarity measured in the RKHS is our criterion for matching two videos. Experiments show the superiority of the SH energy dispersion feature and our proposed recognition scheme.

The main contribution in this part is a novel feature which is robust against pose variations for multi-view face recognition. Secondly, we developed an end-to-end system which consists of both face tracking and video-based recognition modules.

1.4 Organization of the Dissertation

The remainder of the dissertation is organized as follows: In Chapter 2, we present a comprehensive survey of existing works on video-based face recognition. We propose the Conditional Random Field and M^3 network based algorithms for online context-aided face association in Chapter 3. Our video-based face recognition approach using intra-personal dictionaries is presented in Chapter 4. We then pro-

pose the pose-robust Spherical Harmonic energy dispersion feature and describe our face tracking/recognition algorithm for camera networks in Chapter 5. Finally, in Chapter 6 we conclude the dissertation and discuss some future research directions.

Chapter 2

Literature Survey

As we discussed in the previous chapter, a video-based face recognition algorithm belongs to either one of the three categories, according to whether the gallery consists of still images, single-view videos or multi-view videos. We will review related literature naturally in this order. A survey on video-based face recognition methods can also be found in [5].

2.1 Still to Video

The most straightforward method of dealing with this problem is to match each frame to the gallery as in a still-to-still face recognition problem and then apply certain rules to integrate decisions across the frames. The most popular rules include max-sum, min-max, majority voting etc. In this approach, after faces are cropped from video frames through tracking or detection, they usually have to pass several saliency tests to be adopted as inputs for the recognition engine. Typically, these tests attempt to reject non-frontal or poorly-illuminated face crops. For example, Steffens et al. [6] picked the two frames with highest elastic graph model matching score for recognition. A procedure utilizing robust statistics to remove outliers from face image sequence is presented in [7]. In [8], still images in a subject's gallery form the initial eigenspace, which is then updated using the

fames of a test video of the same subject. Only the frames that pass a confidence test are used for the updating. The final decision is made through majority voting among recognition results of individual frames. Zhang and Martinez [9] adopted a probabilistic weighting scheme. They divided a human face into subregions and trained PCA, LDA and ICA subspaces for each of them. To combine the likelihoods of all the subregions and all the frames, a weight is assigned to each subregion in each probe frame for each subject, according to its similarity to the best matched gallery image of the subject. The similarity is measured in terms of pose and expression. Park et al. [10] used a semi-automatically trained AAM to track face and reconstruct the 3D shape. Pose can be estimated from the 3D shape. The face in probe frame is compared to the gallery face images of the same pose through three classification algorithms. The three resulting scores are min-max normalized and added together. The rule for frame-level score fusion is max-sum. Stallkamp [11] et al. proposed to use the distance-to-model (DTM) scheme, which weighs each frame according to its distance to closest match in the gallery, and the distance-to second-closest (DT2ND) scheme, which weighs each frame according to the distance between its best match and the second best match in gallery, to perform temporal fusion. When the DCT feature is adopted, the authors showed through experiments that both schemes outperformed the majority voting and sum rules and even better performance can be achieved if they are combined.

The naive fusion method often suffers from unstable performance because of the ad-hoc nature of the fusion rules. To overcome the drawback, more principled and structured algorithms have been proposed. In [12], Li and Chellappa unify

face tracking and verification in a Sequential Importance Sampling (SIS) [13] [14] framework. The idea is that the posterior distribution evaluated by SIS will achieve high values only when both of the following two conditions are met: 1) The motion parameters are accurate; 2) The appearance template used for tracking has the same identity as the subject in a probe video. Later, Zhou et al. further developed this idea by treating a subject identity and motion parameters as state variables of SIS [15]. The appearance likelihood is calculated using the probabilistic intra/extra-personal subspace algorithm [16]. By marginalizing the joint posterior distribution of the two sets of parameters, they were able to simultaneously perform tracking and recognition. An important feature of the work is its probabilistic mechanism in accumulating recognition confidence. A close, clear and frontal view of face can provide stronger evidence than one of poor visual quality, which could overturn previous wrong decision. In experiments, the proposed approach exhibits better performance than still-based face recognition algorithm. The video-to-video version of this algorithm will be discussed in Section 2.2. In order to handle pose variations, they later extended the work by adding a term in the likelihood which judges the "frontalness" of the face in a frame [17]. The non-frontal face is treated as non-informative as far as recognition is concerned. As the samples of joint distribution are constructed by repeating the samples of motion parameter space for every value of the identity variable, the number of particles grows with the number of subjects enrolled in the gallery. Thus the algorithm does not scale well with the gallery size.

2.2 Video to Video

The video-gallery/video-probe problem can be viewed as a special case of a wider category: face recognition based on image set. The difference is whether the images within a set are treated as temporally ordered information. In many scenarios, the two concepts are used interchangeably. Due to their close relationship, we shall not confine our review to the strict-sense video-video face recognition, but discuss general image set-based face recognition works as well. On the other hand, the video-video recognition problem can certainly be reduced to a still face recognition problem, as implemented in [18](Majority voting), [19] (Minimun reconstruction error) and [20] (Min-min distance).

There are different ways of representing face images in a set. The most popular ones among them are linear subspaces, manifolds, probability distribution and dynamical models.

Linear Subspace Linear subspace analysis has achieved great success in still image-based face recognition. A variety of subspace construction methods, such as Principal Component Analysis(PCA), Linear Discriminative Analysis(LDA), and Bayesian probabilistic subspace [2] have been proposed. When projected onto the trained subspace, the probe face images are generally easier to classify than in the original feature space. In the video-video (or image set-image set) case, the probe is no longer a single point in feature space. Therefore a number of projections need to be collectively considered. An effective solution to this situation is to assume that

images within each gallery or probe set span a subspace and the distances between subpaces can be utilized to characterize set similarity.

The concept of principle angles was first proposed by Jordan in 1875, to account for the relationship between two linear subspaces. Suppose there are two linear subspaces U and V, and the canonical correlations between them are recursively defined as [21]:

$$\cos(\theta_1) = \max_{\mathbf{u}\in U}\max_{\mathbf{v}\in V}\mathbf{u}^T\mathbf{v}, s.t.\mathbf{u}^T\mathbf{u} = \mathbf{v}^T\mathbf{v} = 1 \quad (2.1)$$

$$\cos(\theta_k) = \max_{\mathbf{u}\in U}\max_{\mathbf{v}\in V}\mathbf{u}^T\mathbf{v}, s.t.\mathbf{u}^T\mathbf{u} = \mathbf{v}^T\mathbf{v} = 1, \mathbf{u}^T\mathbf{u}_i = \mathbf{v}^T\mathbf{v}_i = 0, i = 2, \ldots, k \quad (2.2)$$

, where $0 \leq \theta_1 \leq \ldots \leq \theta_k \leq \pi/2$ are called the principle angles between U and V. Numerically, principle angles can be evaluated based on QR factorization of data matrices and singular value decomposition (SVD).

Yamaguchi et al. [22] directly applied the principle angles in their Mutual Subspace Method (MSM), and showed through experiments that it is superior to the still face recognition algorithm based on PCA. Later, in [23], Fukui and Yamaguchi proposed the constrained mutual subspace method (CMSM). They first assumed that the difference vectors $\mathbf{u}_i - \mathbf{v}_i$, where $\mathbf{u}_i$ and $\mathbf{v}_i$ are as defined in 2.1, form the basis of a difference subspace. They claimed the constructed subspace contains only the essential information for discriminating between two classes of faces and generalized this concept to the case of multiple subspaces. Gallery and probe sets are projected onto the generalized difference space (what they called the constrained subspace) and the projections are compared for recognition. The CMSM outperformed the MSM in their experiment. The method was further extended to the Multiple Constrained

Mutual Subspace Method (MCMSM) [24]. In MCMSM, a number of constrained subspaces are constructed, each of which is trained from image sets under a certain pose or illumination condition. The gallery and probe sets are projected onto every constrained subspace. The canonical correlations between the pairs of projections are evaluated, and they are combined through bagging or boosting.

Wolf and Shashua [25] developed a procedure for directly calculating the principle angles in RKHS from a Gram matrix when the explicit form of kernel is unknown [25]. They also proposed a positive definite kernel based on principle angles, which can be used in SVM-like classifiers. In their experiments on image set-based face recognition, each image in a set enters the data matrix as a column. The mean of the smallest 20 principle angles between the column subspaces of two data matrices is adopted as a measurement of set similarity.

Kim et al. [26] employed principle angles to reformulate the criterion function to be optimized in a nonparametric linear discriminant analysis. The correlations in the traditional definition of within-class and between-class scatter are now replaced by the canonical correlations between sets. This method can be regarded as the vector-set version of LDA.

An algorithm which does not fall in this category but has close relationship was proposed by Cevikalp and Triggs in [27]. They approximate the image set by an affine subspace or convex hull. An affine subspace is a superset of linear subspace in that an affine subspace does not have to contain the origin. A convex hull is the tightest convex model containing the samples. They use the geometric distances (distances of closest approach) to compare two sets of face images. De-

spite the conceptual simplicity, the algorithm demonstrates good performance in experiments. Hu et al. [28] also represent each image set with an affine hull, but the distance measure they use is based on the Sparse Approximated Nearest Points (SANP), which are defined as the nearest points of the two sets that can be sparsely approximated by the sample images in the respective sets.

Manifold While the linear subspace-based methods have the advantage of tractability and can conveniently borrow results from relatively matured research on still-based face recognition, their assumptions are oversimplified. Consequently, they are insufficient to characterize the geometrical distribution of face appearance in vector space. Moreover, some argued that even the top principle angles do not necessarily encode identity information [29]. On the other hand, it is generally believed that appearances of a face under smooth illumination and pose variations sit on a nonlinear manifold, whose intrinsic dimensionality is much lower than that of the embedded vector space. The most attractive property of manifold is that it can be locally approximated by the Euclidean space. As such, most of the manifold-based set-to-set face recognition algorithms focus on how to construct a locally linear model.

In practice, what we have is several training and testing videos, which represent sparse samples from the manifold structure. The general strategy of modeling this manifold from a video or image set is as follows: First use a clustering algorithm (K-means, spectral clustering etc.) to group images within the video or image set according to appearance similarity. Then for each group, learn a local linear structure from the samples in that group. In the test stage, there are two choices. One

is applying a fusion rule after evaluating the similarities between individual frames and the trained manifolds as point-manifold distances. The other is constructing a manifold from the probe set and calculate its manifold-manifold distances with respect to the trained manifolds. Since each manifold is approximated by locally linear model, the linear subspace methods reviewed in the previous section can still play a role in this step.

Li, et. al.'s work [30] is based on the concept of manifold, though they used the term "identity surface" instead. They used a 3D model to track the face, providing aligned face texture and pose information at the same time. Kernel Discriminant Analysis is applied to project face appearance onto a low-dimensional space. The identity surface for a subject is constructed as follows: The two basis coordinates stand for tilt and yaw of the face pose. At each point on this coordinate grid, the function value is the KDA vector obtained from the subject's gallery. Since the gallery can only cover a limited region of the grid, the function surface is assumed to be composed of piece-wise planes and interpolation is necessary for novel points. Probe video frames can be looked as points in this space and distances from them to each identity surface are summed for recognition.

Kokiopoulou and Frossard [31] exploited the smoothness assumption of face manifold based on graph. Instead of forming a k-NN graph whose vertices correspond to all the training and testing data, they worked on the part of data which includes all the training faces and frames of only one testing video at a time. Under the constraints that the labels of the testing video frames have to be the same, they optimized a cost function that is related to the smoothness of manifold.

Lee et al. [32] considered the face appearances under pose changes as lying on a pose-manifold. In the training step, they clustered face crops obtained from an video through the K-means algorithm. They modeled each subject's pose manifold as a collection of linear planes trained through PCA. Along with the manifold structure, a transition probability matrix which captures the dynamics of pose variation is also trained. The distance from a face in a test video frame to a manifold is defined as that of its projection on this manifold. It is a probabilistic distance determined not only by the the appearance difference, but also by the pose transition matrix. As an extension to this work, they later [33] integrated the tracking module by iteratively optimizing tracking and recognition parameters while keeping the other fixed. Fan et al. followed this method by replacing PCA with Locally Linear Embedding (LLE).

Arandjelovic and Cipolla [34] assumed that when pose is fixed and illumination is varied, all the intra-personal differences of log-transformed face images form a general shape-illumination manifold independent of identity. The manifold is represented by a probabilistic PCA mixture, which is learned from separate training videos with identities different from those in the gallery and probe videos. Every frame of a probe video is matched to its corresponding frame of each gallery video in terms of pose similarity. The matched pairs' log-differences are used to calculate point-manifold distance. The gallery video yielding shortest average distances identifies the subject in the probe video. This representation of manifold was also adopted in [29], where the proximity of two sets of face images is measured as a combination of the distance between their subspace representations and that between their manifold representations. Both distances are in terms of weighted sum of

canonical correlations between subspaces, as the manifold is also approximated by a collection of subspaces. The weights are learned by the AdaBoost algorithm. Their more recent work [35], however, is conceptually close to [32] in that the face manifold is also modeled as a pose-wise one. They learned the relationship between pose and facial feature points, which allows an automatic clustering of frames in terms of pose. The intra-person variations within such a cluster is thus assumed to be caused by illumination and form a linear subspace. Distances between corresponding pose clusters of the two videos are fused using a RBF nerual network.

Fan and Yeung [36] [37] used Hierarchical Agglomerative Clustering (HAC) to cluster images in a set, on the basis of geodesic distance approximated by the Isomap algorithm. The centers of all clusters are then selected as exemplar images, which are further subject to linear projection by the traditional subspace methods. Image set-based matching is turned into exemplar matching. Finally the decision is made based on majority voting. Liu et al. [38] selected an exemplar which maximize the in-cluster co-occurrence and representative capability from each of the expression clusters of a video. The expression clusters are formed by K-means based on a distance function considering both appearance and temporal closeness. For each expression, a non-parametric version LDA is separately performed. This work can be viewed as based on an expression-wise manifold.

In [39], Wang et al. proposed an algorithm to compute manifold-manifold distance in image set-based face recognition. They use a procedure called Maximal Linear Patch (MLP) to cluster samples in a image set or video. The idea is to let a linear subspace gradually spread from a seed sample to include more and more

nearest neighbors, until the linearity condition is broken. The linearity is measured as the ratio of geodesic distance to Euclidean distance, and the distances are calculated between a candidate neighbor and each existing sample in the cluster. For each cluster, a linear subspace is trained. The manifold-manifold distance is defined as the distance between the closest subspace pair from the two manifolds, and the subspace distance is defined as a weighted sum of canonical correlations and exemplar distance. Inspired by Zhao, et. al.'s Discriminant Clustering Embedding (DCE) work [40], Wang and Chen further developed their above-mentioned algorithm by combining discriminant analysis [41]. In this work, clustering is done through a hierarchical version of MLP. They learned a linear projection aiming to simultaneously minimize the within-class compactness and maximize the between-class separability, which are characterized by center samples' distance between clusters coming from the same or different classes. A test image set will go through the same clustering procedure and is projected with the learned function. The manifold-manifold distance is calculated as before.

Dictionaries Sparse coding and dictionary learning algorithms are extensively used for face recognition following Wright et al.'s work [42]. A popular approach is to use the pixel-representation of the gallery images directly as atoms of the dictionary and then represent each test image as sparse combination of those atoms. Extension of this approach to the video domain is straightforward and natural. Chen et al. [43] partition the video sequence so that frames with same pose and illumination are in one partition. They then built sequence-specific dictionaries

for each gallery video. For each partition, a sub-dictionary is learned where the representation error is minimized under a sparseness constraint. These partition-specific sub-dictionaries are combined to form a sequence-specific dictionary. In the recognition phase, frames from a given query video sequence are projected onto the span of atoms in every sequence-specific dictionary. From the projection on to the atoms, the residuals are computed and combined to perform recognition or verification. They later extended this approach to non-linear kernel space [44]. In [45], Ortiz et al. showed that under the assumption that all frames in a face track will produce the same sparse coefficients when projected onto a learned dictionary, the mean image is an equivalent representation of the whole video. Therefore, they are able to reduce the video-based recognition problem to a still-based problem for the mean images.

Probability Distributions If we treat faces in all frames of a video as samples generated from an underlying distribution, the resemblance of two video sequences can be measured according to probability distribution distance. Usually, it is assumed that the samples are independent and identically distributed. This is obviously not a valid assumption, at least in the video case. Another disadvantage is that the underlying distributions are usually assumed to be Gaussian in order to obtain an analytical form of the distance. This appears to be an insufficient model in a lot of situations.

Shakhnarovich et al. [46] assumed the underlying distribution to be multivariate Gaussian. They implemented the complementary orthogonal subspace learning

method to train the covariance matrix. The Kullback-Leibler (KL)-divergence between gallery and probe is compared for recognition. Arandjelovic and Cipolla [47], instead, assumed the KPCA projection of face images in a video follow a Gaussian distribution. The distance measure they used is the Resistor-Average Distance, which is defined in terms of KL-divergence as follows:

$$D_{RAD}(p,q) = (KL(p\|q)^{-1} + KL(q\|p)^{-1})^{-1} \tag{2.3}$$

The two groups of researchers later cooperated to further extend their works by modeling face instances in a video as samples from a Gaussian Mixture Model (GMM). Since the analytical form of KL-divergence for GMM is not available, they evaluated the value using Monte-Carlo simulation.

In [48], Zhou and Chellappa studied different probability distance measure (Chernoff distance, Bhattacharyya distance, KL divergence etc.) in Reproducing Kernel Hilbert Space (RKHS). They introduced a way to approximate the covariance matrix in the space given the Gram matrix and derived formulas to calculate the aforementioned probability distance. In the video-video face recognition experiment, the divergence distance and the Bhattacharyya distance in RKHS leads to significant improvement over ad-hoc measures.

Dynamical Models Some existing works model the moving face in a video as a dynamical system to capture the temporal information. A dynamical system is often characterized in a state-space way. The temporally varying features are treated as observations emitted by an underlying state variable, whose state transitions form a trajectory in the configuration space. The idea, in the face video case, is to think of

the varying condition such as pose, expression as state variable and face appearances as observations. Given training videos, a dynamical model can be learned for each subject. Note that methods of this category cannot be applied to image sets as temporal continuity is essential to a dynamical system.

Liu and Chen applied Hidden Markov Model (HMM) to the video-based face recognition in [49]. Their observation probability model is a Gaussian mixture model with respect to the eigenface projection and the states are separated through vector quantization. The face video is recognized as the subject whose model receives maximum observation likelihood. They also suggested an adaptive scheme to allow unseen appearance to be learned. Their research also suggested that the hidden state of each model seems to correspond to pose of face. One possible drawback of the algorithm, as Hadid and Pietikainen argued in [50], is that a HMM will give poor results when the video is short because the dynamical model is learned slowly. In Liu and Chen's work, the face image sequence was obtained manually. Kim et al. [51] added a particle filter-based face tracking module to the framework. Besides the incremental PCA subspace likelihood model, they imposed pose and alignment visual constraints, which function as additional terms in the likelihood function. Furthermore, the observation model in HMM was modified to be based on pose-discriminating LDA features and identity-discriminating landmark template features. The method proved to be effective on Honda/UCSD and YouTube celebrity database. Tistarelli et al. [52] built a two-level hierarchical HMM to characterize expression variation in a video. The lower-level is a spatial one, which is trained from clusters of images with the same expression in a video. Each state corresponds

to a subregion of face and the observation vector is composed of order statistics of grey level values. In the higher-level, each state is a HMM in the lower-level. Hence it captures the temporal evolution of face expressions.

Aggarwal et al.[53] used a first-order ARMA model to describe face videos. After the closed-form solution of the model parameters is obtained, they formed an extended observability matrix [54] for each model and calculated the distances between models as principle angles between system matrices.

3D Model One of the earliest works on 3D face modeling from image is the 3D morphable model (3dMM) [55] proposed by Blanz and Vetter. It models both shape and texture of a face as linear combinations of their trained PCA basis, respectively. Model fitting is solved by maximizing the posterior of combination coefficients. In an extension of this work [56], these coefficients were used to achieve face recognition for still images.

Given a monocular video sequence, Chowdhury and Chellappa [57] proposed a method of reconstructing a 3D mesh face model from a monocular video by applying the optical flow-based structure from motion. Zhang et al. [58] generated an animated 3D mesh model from a face video based on bundle adjustment. Breuer et al. [59] also presented an automated procedure for building 3DMMs for images and videos based on facial feature detection results. Although recognition was not performed in these works, it could be readily incorporated into the framework by comparing the shapes of models in 3D domain [60] or the mapped textures in 2D domain or combination of the two [61]. Furthermore, the generated model can be

utilized to synthesize augmented probe image set by varying illumination or pose conditions, which can then be compared with the gallery image sets. This is exactly what was done in [62] and [63].

3D model-based approaches are naturally robust against pose variations, but there are two main concerns about them. First, the computation overhead required for building and fitting a 3D model is large. This is because a human face is not a smooth surface and its geometry requires a delicate model. The dense matching of feature points between consecutive frames is also time-consuming. Second, when models are compared for the purpose of recognition, accurate registration and alignment are essential. Many methods have to rely on manual selection of feature points.

Miscellaneous There are also approaches that do not fall into the above categories. For example, in [64], Edwards et al. employed a filter-based approach to recognize face from video. They tracked face using Active Appearance Models (AAM) and assumed that the model parameter can be represented as a linear combination of identity and non-identity parameters. To decouple the two sets of parameters, they used a three-step procedure: first apply a trained partition model, then use a correction matrix trained by regression, and finally pass through the Kalman filter. The procedure results in an identity-parameter vector for each video. Tang and Li [65] utilized the audio signal to align different videos, under the assumption that people have similar expressions when they speak similar words. They used the unified subspace analysis classifier [66] to compare the corresponding frames of two

aligned videos and fused frame-level result with sum-rule or majority voting rule. Chen et al. [67] divided face into local blocks. They extracted low-freqency DCT coefficients in each block and used them to generate "visual words" as in the bag-of-words approach. Each video is represented by a codeword histogram by averaging the codeword histograms of individual frames. Ye and Sim [68] utilized facial motion patterns for video-based face recognition. The so called Local Deformation Profile (LDP) algorithm calculates motion field and Right Cauchy-Green deformation tensor for each frame, which measures local motion and deformation, respectively. The overall similarity of two face sequences is combination of the two, with local motion similarity also plays the role of confidence score and is used as weight in the temporal fusion stage. It is demonstrated through experiments that subjects can be identified even when the facial motion pattern is learned from a different expression.

2.3 Automatic Face Labeling for Video Archives

In recent years, there has been an increasing interest towards automatic face labeling for large-scale video databases. This application is more related to video archiving than the typical face recognition problem as the main purpose here is to annotate huge volumes of films or news videos. However, we view it as a variation of the video-based face recognition problem, since face appearance similarity is the main, if not the only, criterion of clustering the face tracks extracted from a video database. Compared to the usual video-based face recognition applications, the video archiving task places particular importance on face acquisition and registra-

tion due to the additional uncontrolled factors present in movies and news videos introduced by camera motion and scene change. It is also crucial that the face track matching (recognition) algorithm should be efficient due to the large volume of data being processed.

Berg et al. [69] clustered face images extracted from a large database of captioned news. They initially assigned one or more names to each face with the aid of caption. The faces without assignment ambiguity are used to initialize person classes. Face images then undergo an iterative clustering process: LDA projection matrix is learned from the current clusters, while the clusters are modified by applying the K-means algorithm on the projected vectors, and so on.

Raytchev and Murase [70] performed an unsupervised partition procedure on a collection of video sequences to indirectly recognize the face. The clustering process is guided by two types of interaction forces, attraction and repulsion, imposing both positive and negative values on the matrices of pairwise relations derived from traditional proximity matrices.

Arandjelovic and Zisserman [71] designed a film character retrieval system based on automatic face recognition. They set up a cascade of processing steps to obtain good-quality face tracks. These include an SVM-based eye and mouth detector, an affine transformation, a background suppression method, a bandpass filter and an occlusion detector. Video frames matched to query are retrieved based on L2 distance in the original space or after subspace projection.

Fitzgibbon and Zisserman [72] proposed to look for the closest affine-invariant subspace to the two images to be compared. The sum of the distances between

the images and their projections is shown to be a distance and is invariant to the affine transform. Prior information about transform parameters, such as translation constraints, can be incorporated into the distance function. They cluster the face images detected from films based on the distance matrix. The algorithm can normalize variations which are due to in-plane rotation, translation and scaling, but is sensitive to expression change or out-of-plane rotation. Later, they generalize the distance measure from the image-image case to the image-set versus image-set case in [73]. Each face image set, after being extracted from a movie through face detection, is represented by a PCA subspace. The manifold-manifold distance is defined as the minimum distance between points on two subspaces when the points are subject to affine transforms. They also incorporated learned priors about transformation parameters and modeled image priors as heavy-tailed distributions to handle occlusions. Clustering is performed using an agglomerative strategy.

Sivic et al. [74] proposed a face set retrieval system for movie videos based on local features. In this system, after faces are detected in frames, they are concatenated by an affine covariant region tracker to form face-tracks. Five facial feature points for each detected face are then located using a part-based "constellation" model. SIFT descriptors computed around the feature points are used to compose a bag-of-words. A dictionary of descriptors is built after vector quantization and each face set is represented as a histogram of visual words from the dictionary. Finally face set matching is achieved by calculating the χ^2 distance between histograms.

Everingham et al. [75] combined multiple cues to label the faces in videos of TV serial "Buffy the Vampire Slayer". The KLT tracker was applied to associate

face detections in consecutive frames. They used a tree-structured pictorial model to detect facial features and extracted local SIFT and intensity descriptors around them. They found the speaker in each frame by analyzing mouth motion and assigned to him/her the name obtained from subtitle and script. Face tracks without ambiguous association were treated as exemplar sets. The recognition of a face track is based on its distance to exemplar sets in terms of local descriptors and clothing color histogram. They later [76] generalized their face, facial feature and speaker detectors to profile views, and modified the face appearance classifier to be an SVM based on the HOG feature and multiple kernel learning. A real-time realization of the application can be found in [77]. To realize efficiency in implementation, the authors adopted a kernel-based regressor tracker which is trained from synthetically transformed versions of the initially detected face. For the same reason, they used a random-ferns classifier based on local patches around facial feature points and applied only simple max-max or max-sum rules to match two face sets.

Ramanan et al. [78] used a hierarchical procedure to group faces detected from the TV serial "Friends". Within the same shot, faces are tracked using a part-based color tracker. To join face tracks of different shots within the same scene, an agglomerative clustering algorithm which is based on similarity in face, hair and clothing color histogram and face appearance subspace is applied, with great importance placed on clothing. The same clustering method is also applied to across-episode face tracks, only that now greater weights are assigned to hair color.

Tapaswi et al. [79] modeled videos from the "Big Bang Theory" using a Markov Random Field and combine face, clothing appearance, speech signal and

contextual constraints in a probabilistic framework. They then perform energy minimization to produce labels for the characters. Later they investigate the problem under a semi-supervised setting [80]. They first tagged speaking faces using subtitles and fan transcripts of the videos. Then they propagate the labels to the non-talking faces using a loss function that jointly take into account all the faces and some constraints.

2.4 Multi-View Video

The term "multi-view face recognition" has been used ambiguously. In a strict sense, it only refers to the situations in which many cameras acquire the subject (or scene) simultaneously and the algorithm collaboratively utilizes images taken by different cameras. But more frequently, the term simply means recognizing faces across pose. The ambiguity does not cause any problem in the still-based recognition case. As far as pose variation is concerned, a group of face images simultaneously taken with multiple cameras and those taken with a single camera but at different view angles are equivalent. However, in the video case, the two cases diverge. A multi-camera system has not only more information at its disposal, but also can easily obtain the multi-view images at any time. In contrast, to obtain the same images, a single camera system has to passively wait for the subjects to turn his head. This difference becomes vital in non-cooperative recognition applications such as surveillance. For clarification, we shall call the multiple video sequences captured by synchronized cameras a multi-view video, and the monocular video

sequence captured when the subject changes pose a pose-variation video. With the prevalence of camera networks, multi-view surveillance videos have become more common. Nonetheless, most existing multi-view video face recognition algorithms target pose-variation videos.

A lot of literature on face recognition from pose-variation videos have been reviewed in Section 2.1 and Section 2.2. On the other hand, still image-based multi-view face recognition algorithms, including those based on frontal-view synthesis [81] [82] [83] [84], 3D model reconstruction (See the 3D model part of Section 2.2), subspace or manifold analysis [85] [86] and local feature match [87] [88] [89] [90] [91], can always be extended to pose-variation videos. In addition, data redundancy makes view selection feasible for face videos. One example is Li et al's work in [92]. They first employed edge feature-based SVM regression to estimate poses of face candidates which are targeted by skin color detector. Then, for each candidate, a face detector specific to that pose is applied to judge if it is a face. Only the frontal faces are retained for recognition. The algorithm in [93] also relied on an SVM to select the frontal faces for recognition.

The relatively small number of existing approaches based on multi-view videos reflect the fact that it is not easy to fully exploit such a rich source of information. For example, in [94], although both the gallery and the probes are multi-view videos, they are treated just like monocular sequences. Frames of a multi-view sequence are collected together to form a gallery or probe set. The frontal or near-frontal faces are picked by the pose estimator and retained, while others are discarded. The recognition algorithm is frame-based PCA and LDA fused by the sum rule. In

[95], a three-layer hierarchical image-set matching was adopted, but the cameras did not work in a cooperative way, either. Layer 1 associates frames of the same individual taken by the same camera. Layer 2 matches the frame sets obtained in Layer 1 among different cameras. Layer 3 finalizes the recognition by comparing the output of layer 2 with the training set, which is manually clustered from multi-view videos. The motivation of using multiple cameras in this work is not to handle pose variations, but to deal with occlusion when more than one subject appeared.

Rammath et al. [96] extended the AAM fitting and construction algorithm to the multi-view video case. They demonstrated that when 3D constraints are imposed, the resulting 2D+3D AAM is more robust than the single view case. However, recognition was not implemented in this work. Yoder et al. [97] tracked multiple faces in a wireless camera network. The observations of multiple cameras are integrated using a minimum variance estimator and tracked using a Kalman filter. A clustering protocol is responsible for dynamically creating groups of cameras that track a given face and for coordinating the distributed processing.

In [98], Liu and Chen proposed a geometrical model to normalize pose variations. By back projecting the face image to the surface of an elliptical head model, they obtained a texture map which was then decomposed into local patches. The texture maps generated from different images were compared in a probabilistic fashion. The method was later extended to multi-view videos in [99]. They followed the framework in [15] to simultaneously estimate pose and projection parameters and recognize the texture map. The texture mapping procedure was further extended by adding a geometric deviation model to describe the mapping error. However,

there were still no collaborations among cameras since tracking, texture mapping and recognition were all carried out for each view independently.

2.5 Multi-Modal Fusion

Although face may be the one receiving the most attention from researchers, other cues have also been used for human identification. For example, one can combine face and other features to boost the performance of video-based recognition algorithm. Choudhury et al. [100] fused the eigenface-based face recognition algorithm with the HMM-based speech recognition algorithm in a Bayesian framework. They tested different choices of confidence score and found that the Maximum-Probability to Average-Probability Distance (MPAP) works the best. Zhou and Bhanu [101] fused LDA-based face and gait recognition results with product and sum rules. They reported higher recognition rates than obtained from individual cues. Note that in this work, to utilize the gait feature, they used the side-view of the face for the face recognition module.

2.6 Face Localization

The role played by face localization in video-based face recognition is far more crucial than in the still-based face recognition problem, where the gallery and probe images are normally assumed to be already well aligned. The faces can be localized either by tracking or through frame-by-frame detection. The former has become less of a problem with the availability of Viola and Jones's cascaded face detection

algorithm [102] which is based on Adaboost and Harr feature. The particle filter framework [13] [14] also has become the standard choice due to its capability for handling multi-modal non-Gaussian posterior. However, these facts do not mean face localization is a solved problem. The cascaded face detector yields false positives and false negatives, and these errors can be quite frequent in some cases. Face trackers are challenged by out-of-plane face rotation and severe occlusions. Moreover, results generated by face detector or tracker are often subject to large registration error, which is a potential cause for failure in many recognition algorithms. Inspired by the success of Deformable Part Model (DPM) [103] in object detection and recognition, Zhu and Ramanan proposed a DPM-based face detector [104]. The detector is able to simultaneously detect faces and predict their poses. It has been widely adopted in numerous face-related research works and yielded superior performance in comparison to the Viola-Jones detector. The face detection and tracking algorithms used in previous video-based face recognition works are summarized in Table 2.1. Note that we have excluded the cascaded Adaboost face detector and particle filter from the table due to their dominant applications.

There has also been an increasing demand for a reliable facial feature point detection algorithm in consideration of the important role it can play in multiple tasks related to video-based face recognition. These tasks include face detection, tracking, model registration and so on. We also summarize several facial feature detection algorithms used in video-based face recognition literature in Table 2.1.

Face Detectors	stereo-based local histogram based on wavelet [105] [106] convolutional neural networks [107] skin-color based [108] HOG+SVM DPM	[6] [72] [73] [69] [74] [7] [50] [76] [80] [109]
Facial Feature Detectors	neural networks [110] elastic graph matching [111] AAM [112] edge curvature analysis circular separability filter Haar feature SVM parts-based "constellation" model [113] tree-structured pictorial model Gabor filter	[20] [6] [30] [64] [10] [101] [24] [22] [23] [35] [47] [67] [71] [69] [9] [74] [75] [76] [51]
Face Trackers	regularized kernel-based tracker eigen tracker [114] CAMSHIFT [115] affine covariant region tracker [116] KLT [117] AAM [112] Kalman filter [118] part-based color tracker [119] incremental visual tracker [120]	[77] [32] [67] [74] [53] [75] [76] [68] [64] [10] [30] [18] [78] [51]

Table 2.1: Summary of face detection, tracking and facial feature detection methods.

Chapter 3

Face Association in Videos Using Conditional Random Fields and Max-Margin Markov Networks

3.1 Introduction

Face association refers to the problem of automatically assigning identity labels to a group of faces detected in each frame of a video. An example of face association is demonstrated in Fig. 3.1. In this example, at time t_1, there are four subjects enrolled in the current identity list: Red, Blue, Purple and Yellow (we use color of the corresponding label to refer to the person). But subject Purple does not appear in this scene. At time t_2, a new subject, i.e. subject Green appears and needs to be added to the list. At the same time, there is a false detection (pink bounding box) which should not be assigned any label. At time t_3, subject Yellow, who was absent at t_2 returns to the scene. His face should be re-identified. The detector also misses subject Blue (the dotted bounding box), which should be recovered. The example presents the main issues a face association algorithm should address: false positives, false negatives, ingress/egress of subjects, and re-identification. A successful solution to these problems has immediate applications in video-based face recognition [76, 75], automatic video annotation, automatic collection of large-scale face dataset [78], just to name a few. In this work, we present an online face association algorithm due

to its versatility: aside from scenarios in which general face association algorithms can be applied, there are cases (for instance, video surveillance) that could benefit from the online processing mode.

Traditionally, automatic face association is accomplished by multi-object tracking: Each target face initiates an independent tracker. The tracker searches in successive frames over a neighborhood as determined by temporal coherence constraints for the best match. It is well known that tracking algorithms suffer from drift errors and occlusions. In recent years, the so-called "tracking-by-detection" approaches [121, 122, 123, 124, 125, 126, 127, 128] have gained popularity following the advances in object detection techniques. As the name implies, methods of this type compose detection results from individual frames into tracks. In comparison with traditional tracking methods, these bottom-up methods are more suitable for real-world video processing, as they are free of drift errors, more robust against shaky camera / occlusions and easier to recover from failure. Our algorithm also falls into this category. To emphasize the difference between the two types of schemes, we use the more particular term "face association" or "face labeling" to refer to the bottom-up framework presented in this chapter, but leave the word "multi-face tracking" for the traditional approaches.

It has become evident from empirical studies [129, 130] that contextual features play an important role in the process that human vision system uses to recognize an object. Motivated by this observation, context-based vision has received increasing interest recently. Although there exist different definitions about what context is in a vision application, it generally refers to information extracted from the neigh-

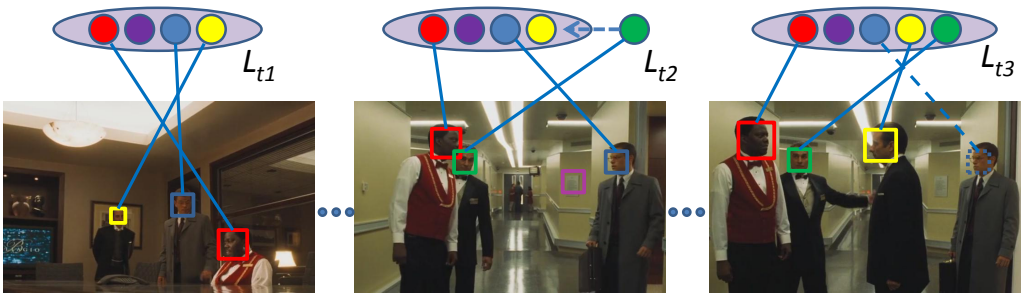


Figure 3.1: **Face association** A face association algorithm solves the correspondence problem between face detections and the identity labels.

bourhood of the region of interest and/or other sources such as maps, time stamp, etc. For face recognition problems, the context can be the hair, or clothing of a subject, or even other subjects in the image. We make intensive use of contextual features in our face association work, because the data we are concerned with are unconstrained videos. By unconstrained videos, we mean those captured under the condition that camera motion, illumination conditions, face pose/expressions, and occlusions are not intentionally controlled. Conceivably, exploiting contextual features is not just beneficial, but actually necessary under such a situation. To this end, it is important to develop a systematic approach which is able to effectively integrate evidence from different sources.

Motivated by these considerations, we propose an online face association algorithm for unconstrained videos. The problem is formulated in a graphical model framework, with each node representing a detected face (either true or false positives) in a video frame. We exploit multiple contextual features as well as local appearances at each node for reliable association. The features are encoded in the

unary and pairwise potential functions of the graphical model. For each feature, we maintain a time-adaptive model which is updated at every frame. This makes our algorithm to operate in the online mode. We present two different approaches to infer the labels of each node. One is based on Conditional Random Fields (CRF) and the other is based on the Max-Margin Markov networks (M^3 networks). The two share the same underlying graph but differ in optimization criterion. To handle the time-varying structure of the graphical model, we introduce the concept of "null state" to handle false detections and novel faces.

Our contributions can be summarized as follows: First, we propose an end-to-end fully automatic framework for face association in videos. Unlike many existing works in multi-object association [121, 122, 123], our method operates in the online mode, which is crucial to many real-world applications. Moreover, our model is dynamic in the sense that not only the features are characterized in a time-adaptive fashion, but also that the number of nodes and states can vary with time. Second, we exploit the abundant contextual features available in the video. The features enter the potential function through both unary and pairwise terms and significantly improve the performance of face association in terms of accuracy and robustness. Third, we present two different learning and inference schemes for the proposed approach and demonstrate their performance on multiple databases.

Aside from the online/offline processing mode, one important difference between our work and many other multi-target tracking algorithms is that we emphasize the use of contextual features. Moreover, to introduce the global feature, we tie a global observation node to every label node. As a consequence, the edge

potentials for labels depend on observations. The resulting model is neither a complete bipartite graph nor a directed acyclic graph, whose solutions can be found by the the Hungarian algorithm [127, 131] and the min-cost flow solution [121, 122], respectively. However, the inference problem can be addressed using CRF or M^3 networks.

The rest of this chapter is organized as follows: We first discuss related work in Section 3.2 and formulate the problem in Section 3.3. We then present the online face association algorithms in Section 3.4. Experimental results are presented in Section 3.5. Finally, we conclude the chapter in Section 3.6.

3.2 Related Works

Visual tracking is a well-studied research topic. Interested readers may refer to [132] for a comprehensive review on object tracking. Generally speaking, almost all single-object tracking algorithms can be naively extended to the multi-target case by initiating multiple independent trackers [133]. However, the naive solution inherits the weakness from single-object tracking algorithms.

In recent years, the tracking-by-detection strategy has been widely adopted for multi-target association. As the name suggests, methods of this type attempt to assemble detection responses into object tracks. This could be achieved within a traditional tracking framework. For instance, Cai et al. [125] incorporated the detector's output into a particle filter in the form of proposal distribution. Breitenstein et al. [124] assigned detection results to a set of particle filters, each of which tracks

a target, and then used the detection results to construct the observation model of the associated tracker. Alternatively, association could also be solved directly from discrete detections, or from short trajectories initialized by the detections (i.e. "tracklets"). It has been shown that association of detector responses to existing tracks can be converted to a job assignment problem and solved by the Hungarian algorithm [127, 131]. From a different perspective, the problem can also be formulated using a network-flow model [121, 122], which is a directed acyclic graph. The globally optimal association can be achieved by finding the min-cost flow. In [126], the Hidden Markov Model and the Viterbi algorithm were applied in the association step. Although the above approaches are advantageous in terms of optimization, they are only compatible with off-line processing. Besides, their structure makes it difficult to combine pair-wise cost functions, which are important if we are to explore contextual features. In association problem, appearance modeling plays an important role. In [134], Kuo et al. learned HOG-based appearance models from the training samples collected on-line for pedestrian tracking. They later extended the work by learning a subset of feature which are most discriminative [135]. The association module of both systems still works off-line. In [136], Yang and Nevatia extracted part-based appearance models for pedestrians and combined them with particle-filter-like trackers.

Two CRF-based approaches were proposed in [123] and [137] for multi-pedestrian association. In [123], a set of energy functions capturing cues from motion, appearance, smoothness etc. were learned off-line using the RankBoost algorithm. [137] further extended the work by learning some of the cues on-line. However, its associ-

ation stage is still an off-line process based on Hungarian and Iterated Conditional Modes(ICM) algorithms. While the first algorithm we present in the chapter is also based on CRF, it differs from [123] in several important aspects; First, in our work, updating of feature functions and face association are performed in an on-line manner. Second, our CRF node represents a detection response, whereas in [123], a node represents a pair of tracklets. This fundamental difference in problem formulation results in significantly different graph structures, models and features. Third, our work follows the canonical framework of CRF in which learning applies to global parameters used to integrate feature functions. In contrast, training in [123] and [137] is local to the individual energy functions themselves. Finally, our work emphasizes the use of contextual features, which is seldom explored in pedestrian tracking works, including [123] and [137].

Most existing multi-target association algorithms are concerned with pedestrian tracking. Face association has issues specific to its own. A common applications of face association is automatic face labeling for TV,movies and news videos [72, 69, 74, 78, 75, 76]. There are usually two steps in such a system, namely connecting detection responses into face tracks within a scene and clustering the face tracks across different scenes. Various face association techniques have been applied in the former case, including KLT tracker [75, 76, 109], particle filter [79], logistic regression [78], affine-invariant clustering [72], constrained agglomerative clustering [74] and Modified Census Transform [80]. However, most of these approaches do not explicitly handle the multi-face scenarios. Some schemes used to match face tracks across scenes, like SVM with multiple kernel learning [76], Markov Random Field

appearance model [79] etc., are also applicable to face association, since affinity measures are required in both cases.

The effectiveness of contextual features has been studied for face labeling and recognition in videos or consumer photos. Gallagher and Chen [139] learned group priors, i.e. the co-occurrence of people, and used it to resolve ambiguous label of faces in a graphical model framework. Clothing appearance has been combined with the face using an MRF model to improve recognition accuracy in [140, 78, 79]. Lin et al. [141] implemented joint people, event and location labeling using cross-domain context. For the people domain, face and clothing appearances were used, while for the event domain and location domain, time-stamps and background histogram were used respectively. Both [75] and [79] exploited speaker analysis to assist face labeling. Yang et al. [142] discovered the auxiliary objects with high co-occurrent frequency and motion correlation w.r.t the target object. Tracking was based on a random field and was achieved in a collaborative way. They showed that the context-aware tracking algorithm exhibits robust performance even in very challenging real world videos.

3.3 Problem Formulation

Suppose there are N detected faces in the current frame F_t of the input video. Let $\mathbf{y}_t = \{y_1, y_2, ...y_N\}$ denote the set of unknown labels we would like to associate with these faces. Let L be the number of all the subjects that have appeared in the scene up to frame F_{t-1}, then the state(label) space is $\mathbb{L}_t = \{0, 1, 2, ..., L\}$. Here we

introduce a "null" state with the label 0 to account for false detections and novel faces. The set of all possible $\mathbf{y}_t$, or the solution space $\mathcal{Y}$, is therefore $\mathbb{L}_t \times \mathbb{L}_t \times, ..., \times \mathbb{L}_t$. Note that both the number of detected faces and the state space vary with time, and the mapping from $\mathbf{y}_t$ to the state space is many-to-one.

We create a graph $G = (V, E)$ and let vertices $V = \{y_1, y_2, ...y_N, \mathbf{x}_t\}$, where $\mathbf{x}_t$ is a global observation node. To maximize the effectiveness of contextual features, we let the label nodes to be fully connected to each other. We define a set of unary or pairwise feature functions in the form of $f(\mathbf{x}_t, \mathbf{y})$ to measure the compliance of a label configuration $\mathbf{y}$ with the image observation $\mathbf{x}_t$. In our CRF and M^3 network models, we use the same graph structure and the same set of feature functions. At the t-th frame, we are trying to solve for the optimal label configuration $\mathbf{y}_t^*$ which maximizes a discriminant function $g(\mathbf{x}_t, \mathbf{y}, \mathbf{w}) = g(\mathbf{w}^T \mathbf{f}(\mathbf{x}_t, \mathbf{y}))$, where $\mathbf{w}$ are parameters. The different interpretations of the discriminant function in CRFs and M^3 networks will be discussed in detail in Sections 3.4.2 and 3.4.3. The assigned label is employed to renew the models used in feature functions and update the state space to $\mathbb{L}_{t+1}$. From this point on, we will omit the time index as long as it does not cause any confusion.

3.4 Context-Aided Face Association

3.4.1 Feature Functions

Our face association methods rely on incorporating face appearance with contextual features, which include clothing appearance, relative scale, relative position

and uniqueness of identity. As a result, the evaluation function has the following form:

$$\begin{aligned}\mathbf{w}^T\mathbf{f}(\mathbf{x},\mathbf{y}) = &\sum_{i\in V} w_\alpha f_\alpha(y_i, \mathbf{a}_i(\mathbf{x})) + \sum_{i\in V} w_\beta f_\beta(y_i, h_i(\mathbf{x})) \\ &+ \sum_{(i,j)\in E} w_\gamma f_\gamma(y_i, y_j, \gamma_{ij}(\mathbf{x})) + \sum_{(i,j)\in E} \mathbf{w}_\eta^T \mathbf{f}_\eta(y_i, y_j, \eta_{ij}(\mathbf{x})) \\ &+ \sum_{(i,j)\in E} w_\lambda f_\lambda(y_i, y_j)\end{aligned} \tag{3.1}$$

, where $f_\alpha, f_\beta, f_\gamma, \mathbf{f}_\eta$ and f_λ are the feature functions for the four aforementioned features, respectively. We demonstrate our use of contextual features in Fig. 3.2. In the following, we define each feature function individually.

Face Appearance: Face appearance provides the most direct evidence about a subject's ID, though for our case its power has been diminished by nuisance factors. We maintain an Online Appearance Model (OAM) [143] for each existing face track, motivated by the algorithm's success in modeling appearances with strong temporal coherence. In an OAM, object appearance is represented by a mixture of three components, namely the stable, wander and lost components. The stable component models steady and long-term appearance; The wander component is responsible for modeling short-term changes in appearance; The lost component accounts for outliers. Considering that alignment errors are often caused by the imperfections of the sliding-window-based detector, in the OAM we use Gabor features, which can tolerate small translation and scale variations, in lieu of raw intensity values.

The model parameters are updated by an Online-EM procedure. Denote the set of Gabor coefficients of a detected face at time t as $\mathbf{a}_t = \{a_{n,t}\}, n = 1, 2, ..., N$,

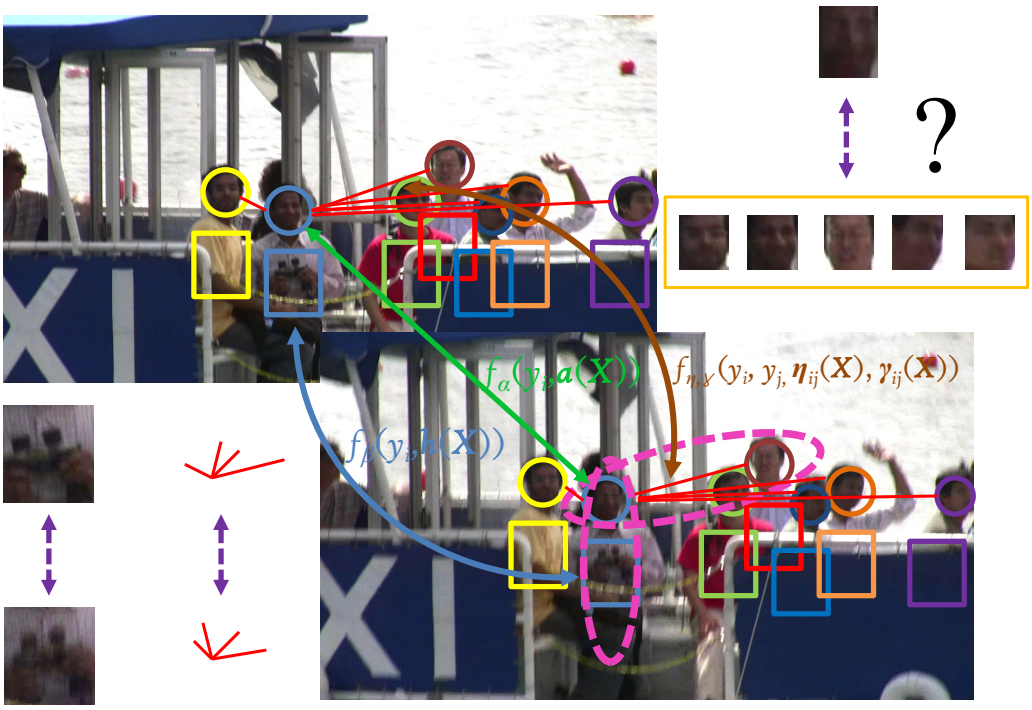


Figure 3.2: **Context-aided face matching** The face appearance alone usually is not sufficient as a strong feature to perform association. Contextual information, such as clothing appearance and relative poses, can be incorporated to make a more confident decision.

and the set of existing, recently updated OAMs as $\mathcal{A}_{t-1} = \{\mathbf{A}_{l,t-1}\}, l = 1, 2, ..., L$. In the E step, we calculate the ownership probabilities of the face with respect to the l-th OAM:

$$\mathbf{o}_{l,q}(\mathbf{a}_t) = \frac{m_{q,t} p_q(\mathbf{a}_t|\mathbf{A}_{l,t-1})}{\sum_q m_{q,t} p_q(\mathbf{a}_t|\mathbf{A}_{l,t-1})} \tag{3.2}$$

, where $q \in \{$ stable, wander, lost $\}$ is the label of the components in OAM. p_{wander} and p_{stable} are the two normal distributions whose parameters are updated every frame for each OAM. p_{lost} is a uniform distribution over the domain of observation feature values. The feature function, which evaluates how likely a node y is in state l, is defined as:

$$f_\alpha(y = l, \mathbf{a}(\mathbf{x}_t)) = \sum_n \log \sum_q o_{l,q}(a_{n,t}) p_q(a_{n,t}|\mathcal{A}_{l,t-1}) \tag{3.3}$$

. The M step happens after the label has been determined through inference. We use the appearance of the node that has been labeled as subject l to update the parameters of the l-th OAM. The set of updating equations can be found in [143]. We illustrate an example of OAM in Figure 3.3.

Clothing Appearance: As a contextual feature, clothing appearance assists the goal of face association effectively. This is especially the case for real-world videos, because: 1) It occupies a larger area than face and hence is easier to extract from a distance. 2) The between-class variation for clothing appearance is usually more distinguishable than face appearance. Given F_i, the center of the face of the i-th subject, we locate the torso by using a probabilistic mask $p(I \in S_i|F_i)$ (See Fig. 3.4), where I is a pixel in the current frame, and S_i is the torso region of the i-th subject. The mask is learned from the statistics of body part's spatial relationship

Figure 3.3: **The Online Appearance Model** From frame $t-20$ to $t-2$ there was partial occlusion, which is still present in the mean of the S(stable) component of the recently updated OAM $\mathcal{A}_{t-1}$ (b). The occlusion disappeared at frame $t-2$. So in the current frame t we get a clean face $\mathbf{a}_t$ (a). (c) is the mean of the W(wander) component of $\mathcal{A}_{t-1}$, which captures this recent appearance change. (d) is produced by subtracting the posterior mixture probability of S component from that of the W component. We can see that the previously occluded region is much better accounted for by the W component than by the S component.

Figure 3.4: **Probabilistic mask of torso** The H3D database (left) and the learned probabilistic mask of torso. The green square marks the position of the reference face.

on the H3D (Human in 3D) dataset [144]. If the clothing histogram feature for a detection is denoted as h, the color feature function for the tth frame is defined as:

$$f_\beta(y = l, h) = \log(1 - d(h, h_{l,t-1})) \tag{3.4}$$

, where d is the chi-square distance between two histograms. The histogram model h_l is also updated at every frame with a forgetting parameter.

It is not a rare situation that two predicted clothing regions S_i, S_j overlap significantly. In such a case, we associate the overlapped region $O = S_i \cap S_j$ to S_i if $d(h(O), h(S_i)) < d(h(O), h(S_j))$ and associate it to S_j otherwise.

Relative Pose: Camera shakes are common while acquiring real-world videos. Unfortunately video stabilization algorithms often fail when complicated or textureless background(water surface, wall etc.) are present. However, the relative scale and

distance features do not suffer as much, and they maintain a temporal coherence. Note that these features cannot be defined in a MRF framework as MRF's edge potentials cannot condition on non-local observations.

We approximate the camera with a weak perspective model. This is a reasonable model since for a camera whose field of view can hold a group of people, the depth variations of the scene points that we are interested in are usually small in comparison to Z_0, the distance between the frontal plane and the image plane. Another assumption implied by the model is that the movement of a face along the camera axis is also insignificant compared to Z_0. Let the scale-normalized distance(SND) between the images of two rigid objects A and B be $\mathbf{\Delta}_{AB} = [(\mu_B - \mu_A)/\omega_A, (\nu_B - \nu_A)/\omega_A]$, where (μ_A, ν_A) are the image coordinates of A's center, ω_A is the size of A's image, and so on. It is easy to show that, when the focal length and the principal point of the weak perspective camera change, the difference of $\mathbf{\Delta}_{AB}$ between two consecutive frames satisfies: $\mathbf{\Delta}_{AB,t} - \mathbf{\Delta}_{AB,t-1} = \delta_{AB}/\tau$, where δ_{AB} is the displacement of A with regard to B in the world coordinate system (we disregard the camera-axis direction for the aforementioned reason) during the same time interval and τ is a constant factor. That is, the SND's change is only dependent on the object's motion and is independent of the camera's zoom or translation.

We define the relative distance feature function as:

$$\begin{aligned} f_{\eta,\mu}(y_i = l_1, y_j = l_2, \eta_{ij}(\mathbf{x}_t)) &= \log \mathcal{L}(\Delta\mu_{i,j,t} - \Delta\mu_{l_1,l_2,t-1} | m_\mu, b_\mu) \\ f_{\eta,\nu}(y_i = l_1, y_j = l_2, \eta_{ij}(\mathbf{x}_t)) &= \log \mathcal{L}(\Delta\nu_{i,j,t} - \Delta\nu_{l_1,l_2,t-1} | m_\nu, b_\nu) \end{aligned} \tag{3.5}$$

, where $\mathcal{L}$ is the Laplace distribution: $\mathcal{L}(x|m,b) = \frac{1}{2b}\exp(-\frac{|x-m|}{b})$. The choice of

Laplace distribution over the more frequently used Gaussian distribution is justified by two considerations: First, the Laplace distribution has longer tails, therefore it is more robust against outliers. Second, in our experiments, the Laplace distribution can approximate the empirical distribution of the features more accurately (See Figure 3.5). Parameters of the Laplace distributions are estimated from the training data using the maximum likelihood method.

In a similar fashion, we define the feature function for relative size as:

$$f_\gamma(y_i = l_1, y_j = l_2, \gamma_{ij}(\mathbf{x}_t)) = \log \mathcal{L}(\frac{\omega_{j,t}}{\omega_{i,t}} - \frac{\omega_{l_2,t-1}}{\omega_{l_1,t-1}} | m_\gamma, b_\gamma) \tag{3.6}$$

Uniqueness: The uniqueness constraint follows naturally from a self-evident fact that no person can appear more than once in the same frame. However, the constraint does not apply to the null state introduced in Section 3.4.4, as multiple new faces and false detections can be present at the same frame. This feature function has the following form:

$$f_\lambda(y_i, y_j) = \begin{cases} -\inf & \text{if } y_i = y_j \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

As we can see, this hard constraint dominates other feature functions and enforces a zero probability when it is violated, but has no influence when it is satisfied. Thus, w_λ can be fixed as 1.

3.4.2 Conditional Random Field

To combine multiple contextual features, we first adopt the CRF-based approach. CRFs are undirected graphical models which characterize the conditional

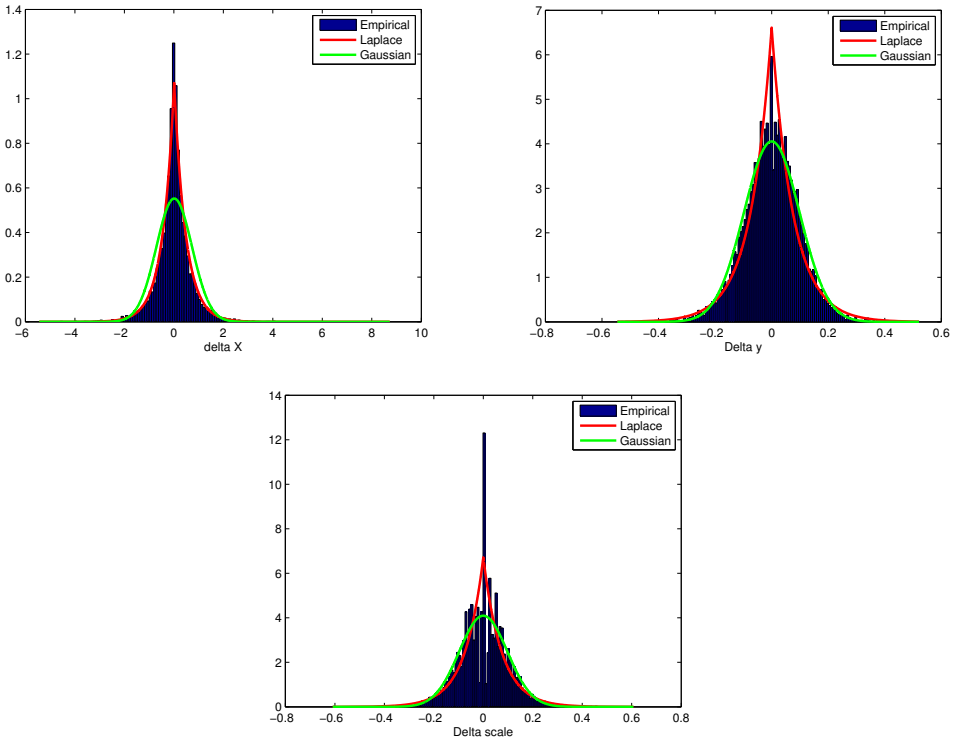


Figure 3.5: **Distributions of relative positions** The empirical distributions are visualized using histograms. The fitted Laplace distribution is plotted in red, and the Gaussian distribution is plotted in green. Parameters are set as the maximum likelihood estimates. The distribution of the x-direction distance variation is much larger than that of the y-direction, which makes sense since the human moves horizontally much more often than vertically.

probability $p(\mathbf{y}|\mathbf{x})$:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \frac{1}{Z(\mathbf{x}, \mathbf{w})} \prod_{c \in C} \Psi_c(\mathbf{y}_c|\mathbf{x}, \mathbf{w}) \tag{3.8}$$

, where C is the set of all cliques in the graph and Ψ_C is the potential function defined for clique c, $\mathbf{w}$ is CRF parameter and $Z(\mathbf{x}, \mathbf{w})$ is the normalization factor. This is in contrast to its generative counterpart Markov Random Fields (MRF), which model the joint probability $p(\mathbf{y}, \mathbf{x})$. In our work, we assume the potential function to possess the log-linear form:

$$\log p(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{f}(\mathbf{y}, \mathbf{x}) - \log Z(\mathbf{x}, \mathbf{w}) \tag{3.9}$$

. The log-linear form not only imposes positivity, but has a close relationship to the Maximum Entropy models. The idea behind Maximum Entropy Models is to find the conditional distribution which achieves the largest possible conditional entropy and at the same time is consistent with the training samples. The principle leads to the following optimization objective function:

$$J(p, \lambda) = H(\mathbf{y}|\mathbf{x}) + \sum_{k=1}^{K} \lambda_k (E(f_k(\mathbf{x}, \mathbf{y})) - E'(f_k(\mathbf{x}, \mathbf{y}))) + \lambda_{K+1} (\sum_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) - 1) \tag{3.10}$$

, where λ_k are the Lagrange multipliers, $H(\mathbf{y}|\mathbf{x})$ is the conditional entropy, $f_k(\mathbf{x}, \mathbf{y})$ are the feature functions and $E(f_k)$ and $E'(f_k)$ are the model and empirical expectations of the feature functions, respectively. Optimization of the objective function yields a probability distribution of the log-linear form. Equation (3.9) also defines $g(\mathbf{x}, \mathbf{y}, \mathbf{w})$ for the CRF case, as this is the objective function we attempt to maximize at run-time.

The main advantage of a CRF over an MRF is that it does not waste resources on modeling the data distribution $p(\mathbf{x})$, which is what we have observed. Because it directly optimizes the conditional probability, the resulting model usually demonstrates better performance than MRF in classification tasks. As an additional consequence, a CRF is capable of incorporating non-local features, and the edge potentials can be either dependent (as in the relative pose feature case) or independent (as in the uniqueness constraint case) of the observation nodes. By comparison, non-local observation node(s) will render an MRF intractable. The "global observation friendly" property makes CRF especially useful for modeling contextual features.

The parameters of the CRF are estimated using a regularized maximum likelihood procedure: Given M labeled data pair $\{\mathbf{x}^{(m)}, \mathbf{y}^{(m)}\}_{m=1,\ldots,M}$, we maximize:

$$E = L + \lambda\|\mathbf{w}\|^2 = \sum_{m=1}^{M} \log p(\mathbf{y}^{(m)}|\mathbf{x}^{(m)}, \mathbf{w}) + \lambda\|\mathbf{w}\|^2 \tag{3.11}$$

. We employ the steepest descent algorithm for optimization for this purpose. As a result, we need to compute

$$\begin{aligned}
\frac{\partial L}{\partial w_p} &= \sum_{m=1}^{M}[\sum_{i\in V} f_p(y_i^{(m)}, \mathbf{x}^{(m)}) - \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^{(m)}, \mathbf{w}) \sum_{i\in V} f_p(y_i, \mathbf{x}^{(m)})] \\
\frac{\partial L}{\partial w_q} &= \sum_{m=1}^{M}[\sum_{(i,j)\in E} f_q(y_i^{(m)}, y_j^{(m)}, \mathbf{x}^{(m)}) - \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^{(m)}, \mathbf{w}) \sum_{(i,j)\in E} f_q(y_i, y_j, \mathbf{x}^{(m)})]
\end{aligned} \tag{3.12}$$

, where f_p and f_q are unary and pairwise functions. The apparently more complicated form over MRF's parameter estimation is owing to the dependency of CRF's partition function on image observation x. In other words, because the contrastive term in the gradient step varies with each training sample rather than remaining

the same as in an MRF, parameter estimation in CRF is $O(M)$ times slower than MRF. Note that (3.12) requires an enumeration of all the possible configurations in the solution space $\mathcal{Y}$, which is generally infeasible. We use the Gibbs sampler [145] to generate samples from the label space. To calculate $p(\mathbf{y}|\mathbf{x}^{(m)}, \mathbf{w})$, we perform approximate inference on the CRF. Although many inference algorithms are applicable, we opt for Gibbs sampling inference because it allows us to reuse the samples generated above. We summarize the CRF parameter learning algorithm in Algorithm 1

At the test time, the MAP solution: $\mathbf{y}^* = \underset{\mathbf{y}\in\mathcal{Y}}{\operatorname{argmax}}\ g(\mathbf{x}, \mathbf{y}, \mathbf{w})$ is also solved by conducting inference, but with learned parameters. Here, we choose the Loopy Belief Propagation (LBP) algorithm [146] for balancing between efficiency and accuracy. Occasionally the LBP may fail to converge, in which case we switch to the variational Mean Field algorithm [147] for a max marginal solution.

3.4.3 Max-Margin Markov Networks

In the CRF approach, we learn the parameters by maximizing the log conditional likelihood. Alternatively, we may choose to apply a margin-based optimization criterion, in which case the M^3 networks will be more appropriate. As has been demonstrated by SVM, the max-margin rule usually results in models with good generalization performance. In spite of its name, an M^3 network is not necessarily an MRF, as it in general does not model a probability distribution. In M^3 networks, we are concerned with the discriminant function $g(\mathbf{x}, \mathbf{y}, \mathbf{w}) = \mathbf{w}^T\mathbf{f}(\mathbf{y}, \mathbf{x})$,

Algorithm 1: The CRF training algorithm.

Input: N labeled training samples$\{\mathbf{x}^{(m)}, \mathbf{y}^{(m)}\}$

Output: Optimal parameters $\mathbf{w}^*$

Initialization: $\mathbf{w}_0 = 0$

while $E_t - E_{t-1} > threshold$ **do**

for $m = 1 \to M$ **do**

- Calculate the un-normalized potential

  $\sum_i \sum_p \mathbf{w}_{p,t-1} f_p(y_i^{(m)}, \mathbf{x}^{(m)}) + \sum_{(i,j)\in E} \sum_q \mathbf{w}_{q,t-1} f_q(y_i^{(m)}, y_j^{(m)}, \mathbf{x}^{(m)})$;

- Apply the Gibbs Sampling algorithm to draw M model samples

  $\{\overrightarrow{\mathbf{y}}^{(k)}, k = 1, ..., K\}$;

- Do inference, calculate $Z(\mathbf{x}^{(m)}, \mathbf{w}_t)$ and $p(\mathbf{y}^{(m)}|\mathbf{x}^{(m)}, \mathbf{w}_t)$;

for $p = \{\alpha, \beta\}$,$q = \{\eta, \gamma, \lambda\}$ **do**

- Calculate $A_n = \sum_{k=1}^{K} p(\overrightarrow{\mathbf{y}}^{(k)}|\mathbf{x}^{(m)}, \mathbf{w}_t) \sum_i f_p(\overrightarrow{y}_i^{(k)}, \mathbf{x}^{(m)})$;

- Calculate $B_n = \sum_{k=1}^{K} p(\overrightarrow{\mathbf{y}}^{(k)}|\mathbf{x}^{(m)}, \mathbf{w}_t) \sum_{(i,j)\in E} f_q(\overrightarrow{y}_i^{(k)}, \overrightarrow{y}_j^{(k)}, \mathbf{x}^{(m)})$;

for $p = \{\alpha, \beta\}$,$q = \{\eta, \gamma, \lambda\}$ **do**

- Evaluate $\frac{\partial L}{\partial \mathbf{w}_p} = \sum_{m=1}^{M} [\sum_{i\in V} f_p(y_i^{(m)}, \mathbf{x}^{(m)}) - A_n]$ and

  $\frac{\partial L}{\partial \mathbf{w}_q} = \sum_{m=1}^{M} [\sum_{(i,j)\in E} f_q(y_i^{(m)}, y_j^{(m)}, \mathbf{x}^{(m)}) - B_n]$;

- Update $\mathbf{w}_{p,t} \to \mathbf{w}_{p,t-1} - \rho \frac{\partial E}{\partial \mathbf{w}_p}$, $\mathbf{w}_{q,t} \to \mathbf{w}_{q,t-1} - \rho \frac{\partial E}{\partial \mathbf{w}_q}$;

Calculate $E_t = \sum_{m=1}^{M} \log p(\mathbf{y}^{(m)}|\mathbf{x}^{(m)}, \mathbf{w}_t) + \lambda \|\mathbf{w}_t\|^2$.

where the parameter vector $\mathbf{w}$ and feature functions $\mathbf{f}$ are defined as before. This is a more flexible form than the CRF model, in which normalization has to be taken into consideration due to the underlying probabilistic interpretation. Indeed, the relationship between CRF and M^3 networks is similar to that between logistic regression and SVM. For a brief introduction to structural SVM, please refer to Appendeix A.

Among all the $\mathbf{w}$'s that satisfy the condition: $\mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}^*) \geq \max_{\mathbf{y}\in\mathcal{Y}\backslash\mathbf{y}^*} \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y})\ \forall m$, M^3 networks seek the one that maximizes the margin, defined as:

$$\eta = \min_m [\mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}^{(m)}) - \max_{\mathbf{y}\in\mathcal{Y}\backslash\mathbf{y}^{(m)}} \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y})] \tag{3.13}$$

, i.e. the smallest difference between the value of discriminant function evaluated at the ground-truth label and that at the runner-up, across all training samples. Therefore, the M^3 network's training objective function is:

$$\max_{\mathbf{w}:\|\mathbf{w}\|=1} \eta\ s.t. \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}^{(m)}) - \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}) \geq \eta,\ \forall m,\ \forall \mathbf{y} \in \mathcal{Y}\backslash\mathbf{y}^{(m)} \tag{3.14}$$

. It is more often than not that the ground-truth and competing solutions cannot be perfectly separated. A common practice is to introduce slack variables, as in SVMs, to relax the constraints. Rather than treating every constraint violation with equal importance, M^3 networks penalize them according to a loss function $\Delta(\mathbf{y}, \mathbf{y}_i)$. After some manipulations, the ultimate optimization problem to be solved is:

$$\min_{\mathbf{w},\, \xi_m \geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_m \xi_m$$

$$s.t.\ \max_{\mathbf{y}\in\mathcal{Y}} [\Delta(\mathbf{y}, \mathbf{y}^{(m)}) + \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y})] - \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}^{(m)}) \leq \xi_m\ \forall m,\ \forall \mathbf{y} \in \mathcal{Y}\backslash\mathbf{y}^{(m)} \tag{3.15}$$

. In (**??**), ξ_m is the slack variable and C is a tunable parameter which determines the trade-off between margin and error tolerance. In this work, we adopt the normalized Hamming loss function:

$$\Delta(\mathbf{y}, \mathbf{y}^{(m)}) = \sum_{i \in V} \mathbf{1}(y_i = y_i^{(m)})/|V| \tag{3.16}$$

, i.e. the percentage of labels assigned in error.

Many algorithms exist for learning the parameters in a M^3 network. However, they all require to perform inference on each training image and solve for $\mathbf{y}'^{(m)} = \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}} \Delta(\mathbf{y}, \mathbf{y}^{(m)}) + \mathbf{w}^T \mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y})$. For tree-structured graphs, exact inference may be accomplished using dynamic programming, but the same task is generally intractable for densely-connected loopy graphs, including those used in our work. To address this problem, we take an approximate inference strategy. The basic idea is to construct a reduced label configuration space $\widetilde{\mathcal{Y}}^{(m)}$ for each labeled training sample $\{\mathbf{x}^{(m)}, \mathbf{y}^{(m)}\}$ and perform inference on the pruned space. To this end, we iterate over all pairs of nodes $\{i, j\}_{i,j \in V}$. Each time, we perturb only the i-th and j-th positions in the ground truth $\mathbf{y}^{(m)}$ using all possible alternative combinations of labels. The obtained competing solutions $\{\mathbf{y}^{(m)}_{\neg(i,j)}\} = [y_1^{(m)}, ..., y_{i-1}^{(m)}, \widetilde{y}_i^{(m)}, y_{i+1}^{(m)}, ..., y_{j-1}^{(m)}, \widetilde{y}_j^{(m)}, y_{j+1}^{(m)}, ..., y_{|V|}^{(m)}]^T$, where $\widetilde{y}_i^{(m)} \neq y_i^{(m)}, \widetilde{y}_j^{(m)} \neq y_j^{(m)}$, are then added to $\widetilde{\mathcal{Y}}^{(m)}$. Here we leverage the fact that the margins are mostly dominated by the most confusing competing solutions. For a frame with N face detection responses, the approximate inference technique reduces the complexity in the original inference problem from roughly $O(N!)$ to $O(N^2)$. Once $\mathbf{y}'^{(m)}$ is inferred with respect to $\widetilde{\mathcal{Y}}^{(m)}$, we are then able to evaluate the subgradient of $\mathbf{w}$ and optimize

the objective function. This subgradient-based M^3 network learning algorithm is summarized in Algorithm 2.

Algorithm 2: M^3 network training algorithm

input : M labeled training samples$\{\mathbf{x}^{(m)}, \mathbf{y}^{(m)}\}$

number of iterations K

learning rate α

output: Optimal parameters $\mathbf{w}^*$

Initialization: $\mathbf{w}_0 = 0$

for $k = 1 \rightarrow K$ **do**

  for $m = 1 \rightarrow M$ **do**

    $\widetilde{\mathcal{Y}}^{(m)} = \emptyset$

    for $i = 1 \rightarrow |V|$ **do**

      for $j = i + 1 \rightarrow |V|$ **do**

        $\widetilde{\mathcal{Y}}^{(m)} = \widetilde{\mathcal{Y}}^{(m)} \cup \mathbf{y}^{(m)}_{\neg(i,j)}$

    $\mathbf{y}'^{(m)} = \underset{\mathbf{y} \in \widetilde{\mathcal{Y}}^{(m)}}{\text{argmax}} \Delta(\mathbf{y}, \mathbf{y}^{(m)}) + \mathbf{w}_{k-1}^T \mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y})$

  $g = \mathbf{w}_{k-1} + C \sum_{m=1}^{M} [\mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}^{(m)}) - \mathbf{f}(\mathbf{x}^{(m)}, \mathbf{y}'^{(m)})]$

  $\mathbf{w}_k = \mathbf{w}_{k-1} - \frac{\alpha}{K} g$

$\mathbf{w}^* = \mathbf{w}_K$

3.4.4 The Null State

So far, the definitions of the feature functions have not considered the issue of the null state. In other words, we have not defined $f(y_i, \mathbf{x})$ or $f(y_i, y_j, \mathbf{x})$ when $y_i = 0$ or $y_j = 0$. It is very challenging to explicitly model the null state, which is an

open-universe set. The problem becomes even more complicated when we attempt to guarantee the function value for the null state to be in an appropriate numerical scale comparable to that of other states.

If we denote the domain of a unary feature function f_p as $\mathcal{X}$, then f_p define a map $Z_p : \mathcal{X} \to R^L$:

$$Z_p(\mathbf{x}) = \mathbf{f}_p = [f(y_i = 1, \mathbf{x}), f(y_i = 2, \mathbf{x}), ..., f(y_i = L, \mathbf{x})]^T \tag{3.17}$$

, where L is the number of the non-null states. We now construct a second map: $Z'_p : R^L \to r^{L+1}$, where r is the closed interval $[0, 1]$ on the real axis, using logistic regression models:

$$Z'_p(\mathbf{f}_p) = [f'_0, f'_1, ..., f'_L]^T \ , \ f'_l = \frac{\exp[\mathbf{w}_l^T \phi(\mathbf{f})]}{\sum_{l'=0}^{L} \exp[\mathbf{w}_{l'}^T \phi(\mathbf{f})]} \tag{3.18}$$

. Here, ϕ is a set of nonlinear basis functions. The pairwise case is a little more complicated. We define a "null state set" $\mathcal{N} = \{(l_1, l_2) | l_1 = 0 \vee l_2 = 0\}$ for the edges of the graph, which contains $L+1$ elements. We can similarly learn a map for a pairwise feature function $Z'_q : R^{L^2} \to r^{(L+1)^2}$, but with an additional constraint: $f'_{(l_1,l_2)} = \rho, \ \forall (l_1, l_2) \in \mathcal{N}$. This is intuitive as there is no reason to favor one null state in $\mathcal{N}$ over another in the eye of an pairwise feature function. Although the logistic regression is a classification algorithm, its output is continuous and so can be interpreted as class-conditional probabilities. So the models define a map for feature functions, with the desired property that their outputs with respect to different states have comparable magnitudes. One limitation of our null state modeling is that we need to learn a model for each different case of state numbers, though training such a set of logistic regression models is computationally manageable.

3.4.5 Removal of False Detections and Recovery of Missed Detections

False positives and false negatives are unavoidable at the face detection stage, though the rate may vary with the detectors applied. To handle false positives, we take the following approach: We first mark those detection responses which are initially assigned with a null label as candidates for false positive or novel faces. They are examined for a number of consecutive frames. Those with low re-appearing rate are considered as false positives and discarded, and the others are kept as novel faces. For faces which are previously detected but are missing from the current MAP solution, we hypothesize according to their most recent SND features with respect to other subjects. Suppose the set of subjects in the solution given by our CRF or M^3 networks at frame F_t is Ω_t, then the set of missing subjects is $\overline{\Omega_t}$. For any $i \in \overline{\Omega_t}$ whose latest presence was at F_{t-t_0}, we generate samples of bounding boxes $\{\mathbf{s}_k = (\mu_k, \nu_k, \omega_k)^T\}_{k=1,\ldots,K} \sim \mathcal{L}(\mathbf{m}, \mathbf{B})$, where:

$$\mathbf{m} = \mathbf{M}_{j \in \Omega_t}((\psi_{t-t_0} + \mathbf{m}_\Delta) + \varphi),\ \mathbf{B} = \mathbf{M}_{j \in \Omega_t}(\omega_{j,t} e^{Ct_0} \mathbf{B}') \tag{3.19}$$

Here, we have utilized the linear transformation property of Laplace distribution. In (3.19), $\mathbf{M}_{j \in \Omega_t}$ denotes matrix-element-wise median over the set Ω_t. We choose to use median in order to get robust prediction of the missing subject's position from the discovered ones. Also, $\psi_{t-t_0} = \begin{pmatrix} \Delta\mu_{i,j,t-t_0} \\ \Delta\nu_{i,j,t-t_0} \\ \omega_{i,t-t_0}/\omega_{j,t-t_0} \end{pmatrix}$, $\mathbf{m}_\Delta = \begin{pmatrix} m_\mu \\ m_\nu \\ m_\gamma \end{pmatrix}$, $\varphi = \begin{pmatrix} \mu_j \\ \nu_j \\ 0 \end{pmatrix}$,

and $\mathbf{B}' = \begin{pmatrix} b_\mu & 0 & 0 \\ 0 & b_\nu & 0 \\ 0 & 0 & b_\gamma \end{pmatrix}$. The variables at the entries of the vectors and matrixes are defined as in (3.5). As the number of consecutive frames that subject i has been missing accumulates, the variance of the Laplace distribution is gradually increased by the exponential factor in which C is an empirically determined positive constant. We let the generated samples go through a two-step verification procedure. At the first step, we check if any of the samples is substantially overlapped with a detection response that is assigned with a null label. If this is the case, then we re-assign the missing label to the detection response. Otherwise we proceed to evaluate the unary feature functions of the samples: $f_{\mathbf{s}_k}(i, \mathbf{x}) = f_\alpha(y_{\mathbf{s}_k} = i, \mathbf{a}(\mathbf{x}))$, $f_\beta(y_{\mathbf{s}_k} = i, \mathbf{h}(\mathbf{x}))$. Only when $f_{\mathbf{s}_k}(i, \mathbf{x}) - \max_{i^* \neq i} f_{\mathbf{s}_k}(i^*, \mathbf{x}) > \epsilon$, where ϵ is a conservatively-set threshold, will a recovery of the missing face i be enforced at $\mathbf{s}_k$. On the other hand, if a subject has been consecutively absent for a certain number of frames, it is considered that he/she has left the scene and the track associated to the subject is then terminated. An example of the sampling-based face recovery result is shown in 3.6.

3.5 Experiments

3.5.1 Database

We evaluated the performance of the proposed algorithms on the following three public video databases:

- **Big Bang Theory database** It consists of 3 episodes (Episode 1, 2 and 6)

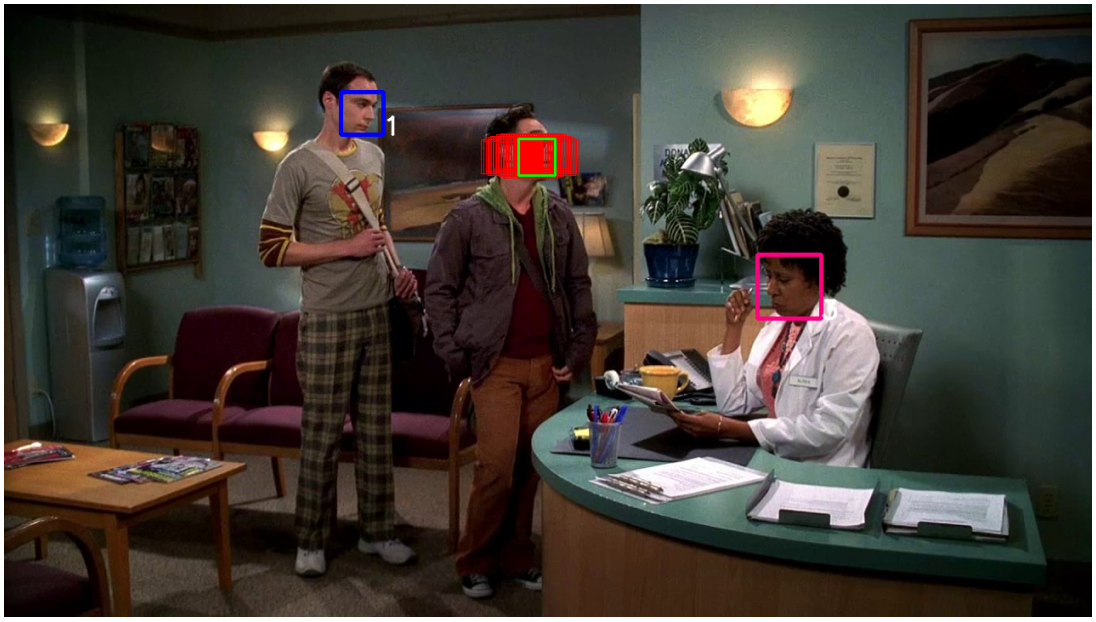


Figure 3.6: **Samples used to recover the missing faces:** Subject 2's face is missed by the face detector. Based on the previous relative position/size features and the current positions/dizes of Subjects 1 and 3, we are able to generate samples (marked by the red bounding boxes) which form the candidates for the position/size of the missed face. The green bounding box marks the final inferred face position.

from the first season of the sitcom "The Big Bang Theory". There are 95052 frames and 2207 face tracks in this database. It has been used to evaluate the TV character labeling algorithms in [79] and [80].

- **Buffy database** This database has been widely experimented with in many automatic face labeling works, including [75, 76, 80]. It contains 3 episodes (Episode 2, 5 and 6) from Season 5 of the TV series "Buffy the Vampire Slayer". The total number of frames and face tracks are 190097 and 3038, respectively. The illumination condition in this database is more challenging than the Big Bang Theory database as it contains many scenes with dim lighting.

- **QMUL Multi-Face database** This database has been used in [149] to test multi-target visual tracking algorithms. It has three video sequences, namely *frontal*, *fast* and *turning*. Although captured by a static camera, all three video sequences contain intense face motions and occlusions. In addition, subjects change their face poses frequently in the *turning* sequence and perform fast movements in the *fast* sequence. There are 2769 frames and 11 face tracks in this database.

For the Big Bang Theory database and the Buffy database, we used the labeled face tracks provided in [80] as the basis of our ground truth. We manually corrected the errors in these annotations and added some missing face tracks. For all the databases, we ran the shot boundary detection algorithms to divide each video into segments. Our face association algorithm runs from boundary to boundary within

each shot.

Aside from the three testing databases, we collected an independent real-world video dataset in outdoor environments for training purposes. It consists of 42 short video sequences with more than 3000 frames. We deliberately simulate the unconstrained conditions by introducing intense camera motions, blur, pose variations and occlusions. The illumination condition is not controlled. The number of subjects showing up in each frame ranges from 2 to 14. The training data set serves two purposes: 1) To learn the parameters for CRFs and M^3 networks. 2) Pairs of consecutive frames are also employed to learn the parameters of the Laplace distributions used to characterize the relative distance and scale features.

3.5.2 Face Detection

We applied the Deformable Part Model (DPM) based face detector [104] to each frame. We observed that the DPM detector results in considerable improvement in accuracy when compared to the Haar cascaded face detector. The detector was then followed by a skin detector using HSV color space thresholding. We marked those face candidates with an unreasonable portion of skin pixels (We empirically determine the upper and lower threshold to be 0.85 and 0.2) as tentative false positives and the remaining ones as tentative true positives. We performed CRF and M^3 network inference for the tentative true positives. However, tentative false positives were not simply discarded. When their locations coincided with a sample in the missing face recovery procedure described in Section 3.4.5, we assigned bonus score

to that sample. We attempted to fuse face detection responses with DPM-based human body detection responses, but found this to degrade the performance.

3.5.3 Evaluation Metrics

To qualitatively evaluate the performance of our algorithm, we adopted a set of metrics commonly used in multi-pedestrian tracking works [135], including:

- **GT**: the number of ground-truth face tracks.
- **Recall**: correctly labeled faces / total ground-truth detections.
- **Precision**: correctly labeled faces / total labelings made.
- **Frag**: fragments, the number of times that a ground-truth face track is interrupted.
- **IDS**: ID switch, the total number of times that a ground-truth face track changes its associated label.
- **MOTA**: multiple object tracking accuracy, defined as:

$$1 - \frac{\sum_n FP_n + FN_n + IDS_n}{GT_n} \tag{3.20}$$

, where FP_n, FN_n, IDS_n and GT_n are the number of false positives, false negatives, ID switches and ground truth faces at the n-th frame, respectively.

To determine if a predicted face corresponds to a face in a ground truth track, we follow the convention in object detection literature by measuring the intersection-

union overlap ratio. If $(A_1 \cap A_2)/(A_1 \cup A_2) > 0.35$, the two regions A_1 and A_2 are considered as matched.

3.5.4 Qualitative Evaluation

We present sample qualitative face association results generated by the proposed M^3 network on the three video databases in Figure 3.7, Figure 3.8 and Figure 3.9, respectively. As demonstrated, the M^3 network is able to produce consistent and accurate face association results even when various challenging situations are present. Fast motion (Figure 3.7, second row) is a common cause of failure when traditional trackers are used, but it is gracefully handled by our algorithm. We attribute this to the overall tracking-by-detection framework, as detectors are largely unaffected by the displacement between frames. Our algorithm is also capable of successfully handling consecutive pose variations (Figure 3.7, third row, Figure 3.8, Figure 3.9) and short-duration occlusions (e.g. first row, Figure 3.8 and first row, Figure 3.9). However, when a face was occluded for a relatively long period of time and re-appears at a novel pose, the proposed algorithm could treat it as from a new subject, as neither appearance-related contextual features nor the relative position features are updated correctly. This accounts for the observed track splits in 3.7, for subject 2 in the first row, subject 1 in the second row and subject 2 in the third row. A subject's entering and leaving the scene is similar to occlusion. A successful example can be found in the second row of Figure 3.9, in which subject 4 temporarily left the scene and re-entered, and subject 3 and 5 left the scene subsequently.

Figure 3.7: **Sample face association results on the QMUL Multi-Face database** The three rows correspond to results for the *frontal*, *fast*, and *turning* sequences, from top to bottom.

We are also able to recover some of the faces that are missed by the detector. These are marked by the black bounding boxes, e.g. fifth image, first row, Figure 3.7 and third image, second row, Figure 3.8. The white bounding boxes mark the detections which are labeled by our algorithm as false positives, e.g. second img, first row, Figure 3.8 and fourth image, third row, Figure 3.9. We can see that our scheme proposed in Section 3.4.5 works effectively in general. However, in the current implementation, there is no mechanism to recognize false negatives or false positives at the initialization stage. This is the reason why subject 4 was not associated with any face tracks for the first three images in the first row of Figure 3.8.

Figure 3.8: **Sample face association results on the Buffy database**

Figure 3.9: **Sample face association results on the Big Bang Theory database**

3.5.5 Quantitative Results

To quantitatively evaluate the performance of our algorithms, we carried out comparison experiments. In the first experiment, we compared our approach with two existing algorithms. The first one is the face processing pipeline used in [151]. It is also similar to that used in [75, 76], but replaces the Haar cascade face detector with a DPM detector. Since we are concerned with face association performance, we did not include their SIFT feature extraction and speaker detection steps in this experiment. In this method, short tracks were formed between detection responses using KLT trackers. The obtained tracklets were then grouped by applying constrained agglomerative clustering. The second compared approach is the min-cost flow algorithm proposed in [122]. It performs greedy successive shortest-path computation on an underlying flow network model. The algorithm uses birth and death states to model subject's entry/exit. It also considers occlusion handling and non-maximum suppression. Comparisons results on the three databases are summarized in Table 3.1, 3.2 and 3.3. Regarding the measures that are related to association accuracy, i.e. MOTA, Frag and IDS, both of the proposed CRF and M^3 network framework consistently outperform the other two algorithms on all three databases. This demonstrates the power of contextual features. The M^3 network algorithm further improve face association accuracy over the CRF. This is an expectable consequence by adopting max-margin rule. The min-cost flow approach achieves the lowest association accuracy, most likely because its transition model is not robust enough. On the other hand, the precision and recall rates provide insights about the

performance of our missing face recovery and false face removal mechanisms. The proposed algorithm again outperforms the compared methods except on the Buffy database, where the tracking-clustering algorithm produces a higher recall rate. Difference in the characteristics of the databases also impacts the performance. For example, the substantially more frequent IDS and fragment errors made on the QMUL Multi-face database are caused by the complicated motions and occlusions that are intentionally introduced to the dataset. On the other hand, the relatively simpler background in this database leads to high recall and precision rates. Aside from performance, the two alternative approaches work in an off-line mode, but ours operates in an on-line fashion, which has a wider range of real-world applications.

In the second experiment, we compared the relative importance of different features on the Buffy database. To this end, we removed the face/clothing/relative pose feature function one at a time in the M^3 networks framework. We kept the uniqueness constraint all the time, otherwise the performance would be substantially degraded. The comparative result is presented in Table 3.4. As the result suggests, face appearance is still the most important evidence for face association accuracy, as MOTA drops most drastically after we remove it from the feature set. This feature plays an especially important role in avoiding ID switch errors. This is intuitive since clothing with similar colors or occluded by the same object tend to cause confusions (e.g. third row, Fig. 3.8). The spatial features, i.e. the relative position and scale feature, rank the second place in terms of effectiveness. They are the most reliable cues under certain circumstances, including: blur, false negatives, or shaky cameras. Disregarding this feature leads to high occurrences of fragment error and significantly

Table 3.1: Comparison of face association algorithms on the QMUL Multi-Face database

Method	**GT**	**Recall**	**Precision**	**MOTA**	**Frag**	**IDS**
Tracking-Clustering	11	96.1%	98.3%	61.8%	50	23
Min-Cost Flow	11	93.6%	98.1%	53.7%	61	29
CRF	11	96.5%	98.5%	65.2%	42	20
M^3 Networks	11	96.7%	98.5%	68.8%	37	17

lowers the recall rate. Performance degradation after dropping the clothing feature can be mostly accounted by cases in which blurred and low-resolution faces are present. In general, every feature plays an important role, and we achieve the best performance by combining all of them in the proposed approach.

3.6 Conclusions

In this chapter, we proposed an automatic end-to-end on-line face association framework. We made use of multiple contextual features that can effectively assist the task under challenging situations. We demonstrated that CRFs are particularly suitable for integrating contextual features due to their ability to handle global observations. We further improved the association performance by switching from maximum likelihood to max-margin optimization criterion, resulting in an M^3 networks solution. We presented approximate inference methods to address tractability

Table 3.2: Comparison of face association algorithms on the Big Bang Theory database

Method	**GT**	**Recall**	**Precision**	**MOTA**	**Frag**	**IDS**
Tracking-Clustering	2207	79.1%	91.8%	68.3%	202	32
Min-Cost Flow	2207	76.9%	87.6%	64.7%	247	30
CRF	2207	81.3%	94.1%	74.9%	188	27
M^3 Networks	2207	81.7%	95.0%	75.3%	179	24

Table 3.3: Comparison of face association algorithms on the Buffy database

Method	**GT**	**Recall**	**Precision**	**MOTA**	**Frag**	**IDS**
Tracking-Clustering	3038	76.2%	88.5%	65.3%	345	52
Min-Cost Flow	3038	73.9%	86.4%	59.7%	352	59
CRF	3038	75.4%	92.3%	68.2%	311	45
M^3 Networks	3038	75.9%	93.9%	70.7%	305	39

Table 3.4: Comparison of contextual features on the Buffy database

Method	**GT**	**Recall**	**Precision**	**MOTA**	**Frag**	**IDS**
M^3 Networks (no face feature)	3038	73.8%	89.7%	61.5%	407	81
M^3 Networks (no clothing feature)	3038	74.9%	90.6%	66.1%	332	44
M^3 Networks (no relative distance/scale feature)	3038	69.4%	94.2%	63.2%	435	56
M^3 Networks	3038	75.9%	93.9%	70.7%	305	39

issues when applying CRF or M^3 networks to our fully connected graph. We introduced null label and sample-based face recovery mechanism to manage false positives, false negatives and faces entering/leaving the scene. Our algorithms achieved promising experimental results on challenging public face association databases. In the future, we will concatenate the current algorithm with cross-scene face recognition module, in order to develop a complete system for face extraction and naming from videos. We believe the proposed methods have laid a solid foundation for this goal.

Chapter 4

Video-Based Face Recognition By Intrapersonal Dictionary Learning

4.1 Introduction

We are witnessing a growing interest in video-based face recognition (VFR) research in recent years. Part of this is driven by the increasing demand for processing digital video contents over the Internet. It is reported that over 14,000 hours of new videos are uploaded to YouTube every day. From a technical perspective, the attraction of videos comes from the fact that they contain extra spatial-temporal information that can be exploited to improve recognition performance. Moreover, videos arise naturally in many applications like surveillance. It is expected that VFR can play an important role in cases where the still image-based algorithms do not return satisfactory results.

In this chapter, we attempt to improve the performance of VFR in the following two aspects:

Face Localization and Normalization As the first steps in almost every VFR algorithm, face localization and normalization are of vital importance to the processing of unconstrained videos. This is where we try to bridge the gap between unconstrained and constrained videos in terms of source data quality. Recent advances in object detection technology have stimulated new research on "tracking by detection" approaches and facial feature detectors. The former is more robust

against drift errors in comparison with traditional trackers. The latter enables us to perform accurate face alignment when large pose variations are present. However, tracking and aligning faces "in the wild" is still a highly challenging task.

Scalability and Generalization The majority of existing VFR algorithms are devoted to discovering features which are closely correlated with identity. However, it requires a large amount of training data to effectively characterize a subject. More often than not, we have insufficient training samples to account for all possible variations for each subject. As a result, decision boundaries of the classifiers are often highly dependent on the training data and are prone to change every time we add new subjects to the database. Such a strategy is inherently inflexible and unscalable. Moreover, for practical uses, while it is desirable that a VFR algorithm be capable of working across databases, most existing approaches have difficulty in addressing this issue.

Our fully automatic VFR algorithm works as follows: Faces are first localized from videos using a tracking by detection approach. A fiducial point detector is then applied to each tracked face. The detector is based on a structural SVM (SSVM) learned by optimizing an objective function that emphasizes on improved localization accuracy. It provides both coordinates of feature points and the quantized face pose. Based on the estimated pose, the localized faces are then aligned to pose-specific common reference coordinate frames. They are further clustered using a non-parametric Bayesian model to remove temporal redundancy. Classification is performed on these cluster centers. We construct pose-specific dictionaries as our classifiers. However, in our work, the discriminative dictionaries do not directly

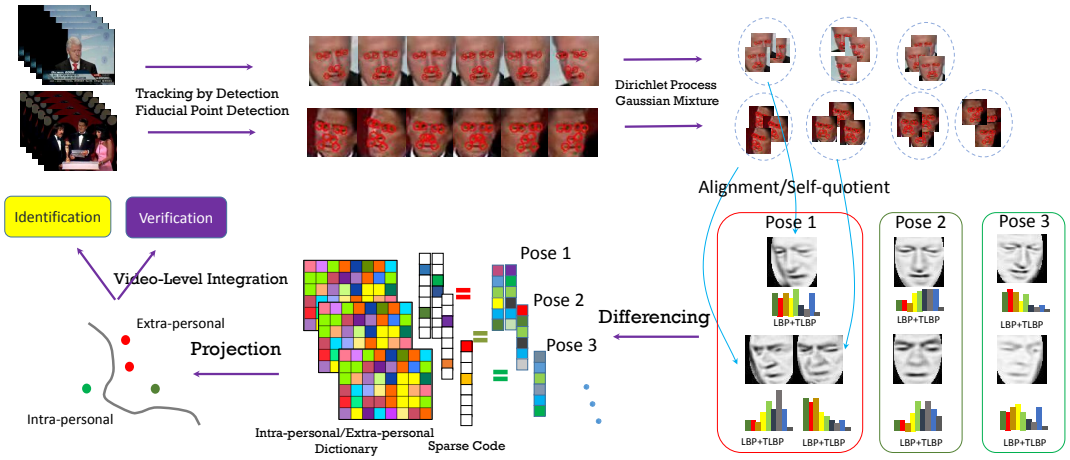


Figure 4.1: Processing pipeline of the proposed video-based face recognition algorithm.

assign an identity label to each test sample. Rather, it attempts to distinguish the intra-personal face appearance variations from the extra-personal ones. Such dictionaries are generic in nature and are capable of working across data domains. An overview of the proposed approach is given in Figure 4.1.

Our contributions are three-fold: First, we develop a novel VFR algorithm based on discriminative dictionary learning and the concept of intra-personal variations. As a result, the algorithm can achieve good performance in terms of accuracy, generalizability and scalability at the same time. Second, we propose an end-to-end solution to the real-world VFR problem. It allows us to reliably localize and recognize face videos "in the wild". Third, we demonstrate through comprehensive experiments that the proposed algorithm outperforms state-of-arts methods on multiple public VFR databases.

The rest of the chapter is organized as follows: We first discuss related work in Section 4.2. In Section 4.3, we present the SSVM-based fiducial point detector and the face localization module. The proposed intra-personal space VFR framework is then described in Section 4.4 together with background knowledge in discriminative dictionary learning. Finally we present the experimental results in Section 4.5 and conclude the chapter in Section 4.6.

4.2 Related Works

Video-based face recognition can be viewed as a special case of a broader problem: face recognition based on image set. In practice, the two terms are often used interchangeably when the image sets are sampled from videos. Various representations of image sets have been explored, including linear subspaces, dictionaries, manifolds, probability distributions, dynamical models etc. By modeling image sets as subspaces, Yamaguchi et al. developed series of methods (Mutual Subspace Method(MSM) [22], constrained Mutual Subspace Method (CMSM) [23], Multiple Constrained Mutual Subspace Method (MCMSM) [24]) to measure the similarity of face videos using canonical correlations or principle angles. The measure of principle angles is also adopted by Wolf and Shashua in a kernel-based approach [25] and by Kim et al. in nonparametric linear discriminant analysis of face image sets [152]. Wang et al. [153] model image sets with their covariance matrices and derive a kernel to map the covariance matrices from the Riemannian manifold to Euclidean space. Cevikalp and Triggs [27] approximate an image set by a convex hull or an

affine subspace and compare them using the geometric distance. Hu et al. [28] also represent each image set with an affine hull, but the distance measure they use is based on the Sparse Approximated Nearest Points (SANP), which are defined as the nearest points of two sets that can be sparsely approximated by the sample images in their respective sets. Chen et al. [43] build sequence-specific dictionaries for each gallery videos and use reconstruction residual error to match galleries and probes. In [45], Ortiz et al. show that under the assumption that all frames in a face track will produce the same sparse coefficients when projected onto a learned dictionary, the mean image is an equivalent representation of the whole video. The algorithms which model image sets as manifolds follow a common strategy: First they apply clustering algorithms like K-means or Hierarchical Agglomerative Clustering (HAC) [41] to group video frames, then construct linear subspaces for each cluster. Different types of manifold-manifold distance are then defined for the purpose of recognition. In addition to learning the manifolds for recognition, Lee et al. [33] also use them to aid tracking by capturing the dynamics of face pose variations with a transition probability matrix. Wang et al. studied manifold-manifold distance in [39] and conducted discriminative manifold analysis for image sets in [41]. There are also existing works in which video frames are interpreted as samples from an underlying distribution. For example, Arandjelovic and Cipolla [47] assume a Gaussian distribution for Kernel PCA projections of face images. Resistor-Average Distance is then used to characterize the similarity of two videos. They later extended the approach to the Gaussian Mixture Model case. Zhou and Chellappa [48] performed video-based face recognition using various probability distance measures (Chernoff

distance, Bhattacharyya distance, KL divergence etc.) in a Reproducing Kernel Hilbert Space (RKHS). Besides the above-mentioned works which mostly explore the visual information of videos, dynamical models such as Hidden Markov Model [49] and ARMA [53] have also been used to incorporate the temporal information in VFR.

In [2], Moghaddam et al. first proposed the Bayesian face recognition algorithm. The intrapersonal subspace Ω_{In} is defined as the subspace constructed from within-class sample differences $\{\Delta_{In}\}$ using Principal Component Analysis. It accounts for appearance variations of the same subject that arise from factors like pose, lighting, expression etc. Similarly, the extrapersonal subspace Ω_{Ex}, which characterizes appearance variations caused by intrinsic identity differences, is constructed using the between-class sample differences $\{\Delta_{Ex}\}$. At test time, the difference between a probe face image I_p and a galley image I_g, Δ_{pg}, is projected onto Ω_{In}. The likelihood $P(\Delta_{pg}|\Omega_{In})$ is computed as a product of two Gaussian densities: one for the projection onto the principal space and the other for the complement space. $P(\Delta_{pg}|\Omega_{Ex})$ is evaluated likewise. Finally a Bayesian classifier is applied to obtain the recognition result. Inspired by the success of this algorithm in still image-based face recognition, many works followed. Wang and Tang [66] showed the relationship between Bayesian face recognition and two other frequently used subspace approaches: PCA and LDA. Chen et al. [154] modeled the joint distribution of a pair of faces in the original feature space instead of the difference vector space. The joint distribution is assumed to be Gaussian and their covariance matrices are learned using the EM algorithm. The metric learning approach proposed in [155]

also explored the difference images. It attempted to learn a symmetric positive definite matrix Q, which can be used to calculate the Mahalanobis distance for a pair of images: $d_Q(\Delta) = \Delta^T Q \Delta$. This is closely related to the method based on Gaussian densities proposed for Bayesian face recognition.

In recent years, sparse coding [156] has gained popularity in the field of image classification. Wright et al. [42] successfully applied their Sparse Representation-based Classification (SRC) framework to the still image-based face recognition problem. In this work, the dictionary used to reconstruct face images is composed of all the training samples. In more general cases, dictionaries need to be learned from data. The K-SVD algorithm [157], an iterative method to learn over-complete dictionaries, is one of the most widely used dictionary learning approaches. However, as it focuses on the reconstruction error using sparse codes, K-SVD is not well suited for classification tasks. Many discriminative dictionary learning algorithms which include classification error terms in the objective function have been proposed. Jiang et al. [3] presented a discriminative dictionary learning framework by enforcing label consistency constraints in addition to sparsity and reconstruction error terms. The projection matrix used for classification is learned along with the dictionary. In [158], the additional constraints include the discriminative fidelity terms and the discriminative coefficient term based on Fisher's discriminant.

Face fiducial points detection has been shown to be critical for solving the unconstrained face recognition problem. The detectors often utilize both spatial relationship and appearance information to localize the feature points. Everingham et al. [75, 76] used a mixture Gaussian tree spatial model and an Adaboost trained

Haar feaure classifier to detect facial features on faces found in TV videos. Belhumeur et al. [159] formulated the feature point localization problem in a Bayesian framework and use a RANSAC-like trial-and-error procedure to find the optimal configuration from a pool of plausible samples. One limitation of the above work is that they train a single detector to apply to faces of various poses. Since the spatial arrangements and visibility of facial features can change dramatically from pose to pose, these detectors may encounter difficulties in processing face images in the unconstrained settings. Zhu and Ramanan [104] extended the Deformable Parts Model (DPM) for face detection, pose estimation and feature localization. The model is also a mixture of tree-structure sub-models, each of which corresponds to a pose prototype. It is trained using the max margin criterion and hence can be globally optimized. The facial feature detector used in our work shares some similarities with this work in that we also enforce max margin constraints to train a mixture of pose-specific models. However, as we show in Section 4.3, while their objective function is designed to guarantee the capabilities of detecting both the whole face and the facial features, ours is tuned to improved accuracy in fiducial points localization.

4.3 Face Localization and Alignment

Our face localization module falls in the "tracking-by-detection" paradigm. We apply a Viola-Jones face detector to each frame of a video. Then we evaluate

the image likelihood of each face candidate as:

$$L(\mathbf{x}_{i,t}|I_t) = \ln \mathcal{N}(\mathbf{x}_{i,t}|\mathbf{x}_{t-1}, \mathbf{\Sigma}) + \lambda \ln p(\mathbf{x}_{i,t}|W_{t-1}) \tag{4.1}$$

, where $\mathbf{x}_{i,t}$ is the bounding box's coordinates of the i-th face candidate found in frame I_t, and W is a WSL appearance model [143] updated at each frame. Apparently, the two terms penalize location inconsistency and appearance inconsistency respectively. The candidate with the largest likelihood is added to the face track and updates the appearance model. If no faces have likelihood values above a set threshold or no faces are detected at all in the current frame, a particle filter will be initiated. It performs face tracking until the detector starts to find a valid face again. The particle filter also uses the likelihood model as defined in (4.1). This simple strategy proved to be very effective in our experiments.

To detect face fiducial points from the localized face, we train a structural SVM [160] (For a brief introduction to structural SVM, please refer to Appendeix A). Its coefficients control the relative weights of feature functions which are computed based on a mixture of pictorial structure models $\{T_m, m = 1, 2, ..., M\}$. Each component of the mixture accounts for the configuration of fiducial points for a specific range of face poses. Here, we divide the poses according to the yaw angle of face and the boundaries are set as $\{-45°, -30°, -15°, 15°, 30°, 45°\}$. We opt for multiple pose-specific models rather than a single shared model for two reasons. First, face fiducial points could have totally different configurations across poses. For example, when a face is in profile pose, half of the fiducial points will be occluded. Even for those feature points which are visible in all the poses, the pose-specific model

can enforce constrains on the state space. Second, such a mixture model will allow us to estimate the face pose as a byproduct. Pose information is required when we construct the intra/extra-personal dictionaries at the next stage. Note that for the purpose of face alignment, usually a set of sparse features is sufficient. Following [76], we pick eye corners, mouth corners, nose corners and nose tip as points of interest. Intuitively, the number of feature points in each model varies due to occlusion.

The structure of our face fiducial point model is similar to that of the mixture of pictorial model defined in [104]. For a fiducial point configuration $z = \{L, m\}$, where $L = \{l^i\} = \{(x^i, y^i)\}$ are the image coordinates and m is index of the mixture component that the fiducial points are associated with, we define its score function as:

$$f(I, \mathbf{z}) = \mathbf{w}^T \mathbf{\Phi}(I, \mathbf{z}) = \mathbf{w}_m^T \phi_m(I, L) = \sum_{i \in V_m} \mathbf{q}_m^{iT} \psi_m(I, l^i) + \sum_{ij \in E_m} a_m^{ij} dx^2 + b_m^{ij} dx + c_m^{ij} dy^2 + d_m^{ij} dy \tag{4.2}$$

, where:

$$\mathbf{w}^T = [\mathbf{w}_1^T, \mathbf{w}_2^T, ..., \mathbf{w}_M^T], \;\; \mathbf{\Phi}(I, \mathbf{z})^T = [0, ..., 0, \phi_m(I, L), 0, ...0] \tag{4.3}$$

. In (4.2), V_m and E_m are the nodes and edges of the m-th pictorial model in the mixture, respectively. $\psi_m(I, l^i)$ is a local visual descriptor extracted at the neighbourhood of l^i. In our case, the CENTRIST descriptor [161] is used. For every pair of fiducial points connected by an edge, the pairwise term in (4.2) captures their spatial relationship. As defined in [104], dx and dy are the displacements of fiducial point i w.r.t. fiducial point j in x and y directions. The sparse augmented feature

function $\phi_m(I, L)$ only activates the mixture component whose index is encoded in $\mathbf{z}$. We can jointly localize the fiducial points and estimate the face pose by maximizing the potential function:

$$\mathbf{z}^* = \{L^*, m^*\} = \underset{L,m}{\operatorname{argmax}}\, \mathbf{w}_m^T \phi_m(I, L) \tag{4.4}$$

To learn the parameter $\mathbf{w}$, we solve the following margin re-scaling structure SVM problem:

$$\begin{aligned} &\min_{\mathbf{w},\, \xi_n \geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_n \xi_n \\ &s.t.\ \max_{\mathbf{z}\in\mathcal{Z}}[\Delta(\mathbf{z}, \mathbf{z}_n) + \mathbf{w}^T\mathbf{\Phi}(I_n, \mathbf{z})] - \mathbf{w}^T\mathbf{\Phi}(I_n, \mathbf{z}_n) \leq \xi_n,\ \forall n,\ \forall \mathbf{z} \in \mathcal{Z}\backslash\mathbf{z}_n \end{aligned} \tag{4.5}$$

, or equivalently:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_n \max_{\mathbf{z}\in\mathcal{Z}}[\Delta(\mathbf{z}, \mathbf{z}_n) + \mathbf{w}^T\mathbf{\Phi}(I_n, \mathbf{z})] - \mathbf{w}^T\mathbf{\Phi}(I_n, \mathbf{z}_n) \tag{4.6}$$

. In (4.5) and (4.6), $(I_n, \mathbf{z}_n)$ is an image-label pair in the training database and $\mathcal{Z}$ is the viable label configuration set. As in the single-output SVM case, each training sample is assigned with a slack variable ξ_n to relax the constraints. $\Delta(\mathbf{z}, \mathbf{z}_n)$ is the loss function of a output $\mathbf{z}$ when measured against the ground-truth label $\mathbf{z}_n$. Suppose there are K fiducial points in total and the subset of indexes of those fiducial points visible for the m-th pictorial model is $S(m)$. The loss function is defined as follows:

$$\begin{aligned} \Delta(\mathbf{z}, \mathbf{z}_n) &= \sum_{k=1}^{K} \|\delta_k\|_2 \\ \delta_k &= \begin{cases} L_k - L_{n,k} & \text{if } k \in S(m) \cap S(m_n) \\ L_{n,k} & \text{if } k \in S(m_n) \setminus S(m) \\ c & \text{if } k \in S(m) \setminus S(m_n)) \end{cases} \end{aligned} \tag{4.7}$$

. We assign a constant c in the third case because if a false positive feature point shows up in prediction, it should be penalized uniformly, irrespective of its coordinates.

In comparison, the optimization function used in [104] is:

$$\begin{gathered}\min_{\mathbf{w},\,\xi_n\geq 0}\frac{1}{2}\|\mathbf{w}\|^2+C\sum_n\xi_n\\ s.t.\ \forall n,\ \forall\ I_n\in neg,\ \mathbf{z}\in\mathbf{z}\ \mathbf{w}^T\mathbf{\Phi}(I_n,\mathbf{z})\leq -1+\xi_n,\\ \forall I_n\in pos,\mathbf{w}^T\mathbf{\Phi}(I_n,\mathbf{z}_n)\geq 1-\xi_n\ \forall k,\mathbf{w}_k\leq 0\end{gathered}\tag{4.8}$$

, where *pos* contains the positive training images with a face and *neg* contains the negative ones with only background. Apparently, the constraints in (4.8) focus on the margin between face and non-face images. In contrast, we use a different definition about positive and negative training samples: Every training image in our case has a face in it. The positive samples are the ground-truth fiducial point configurations of the faces and the negative samples are just any configurations other than the ground-truth ones. Therefore, our objective function explicitly imposes constraints on the margin between correct and wrong landmark predictions. Moreover, while [104] treats all the fiducial point configuration equally for a negative training image, in our case the margin is re-scaled by a loss function $\Delta(\mathbf{z},\mathbf{z}_n)$ which penalizes the negative training samples according to their misalignment errors. In summary, our method is not designed to detect face and facial feature points at the same time as in [104]. Instead, it aims for higher accuracy in localizing the landmarks from a previously detected face.

We employ the subgradient algorithm to learn the parameter $\mathbf{w}$. The update

equation is:

$$\begin{aligned} g =& \mathbf{w}_{t-1} + C \sum_{n=1}^{N} [\mathbf{\Phi}(I_n, \mathbf{z}_n) - \mathbf{\Phi}(I_n, \mathbf{z}'_n)] \\ \mathbf{w}_t =& \mathbf{w}_{t-1} - \frac{\alpha}{T} g \end{aligned} \tag{4.9}$$

. Here T is the number of iterations, and $\mathbf{z}'_n$ is the configuration leading to the most violated constraints. At test time, we follow a two-step procedure to solve the inference problem defined in (4.4). First, we solve for the best L for each individual model in the mixture. Although the cardinality of the entire configuration space is extremely large (in the order of 10^{18}), we only need to be concerned with a very small portion of it at run-time, thanks to the models' tree structure. Dynamic programming (more specifically in this case, the Max-sum inference algorithm) can be applied at this step to select the best configuration efficiently. Then we compare across models to choose the optimal solution. The result of model selection also gives a rough estimate of face pose. We detect fiducial points on every face localized by the detector or tracker. A linear conformal image transformation calculated from point correspondences is then applied to align faces to a canonical frame. Note that there are M such canonical frames, each of which is associated with a model from the mixture.

4.4 Intrapersonal Dictionary Learning

4.4.1 Sparse Coding

We now discuss the problem of modeling intrapersonal face appearance differences using sparse coding. Since video can be viewed as a special case of an image set,

we will first discuss general image/frame-based recognition using the intrapersonal dictionary and leave the video case to Section 4.4.2. Let $\mathbf{X} = \{\mathbf{x}_i, i = 1, 2, ..., N\} \in R^{d \times N}$ be the set of vectorized intrapersonal difference training images/frames. The intrapersonal dictionary $\mathbf{D} = [D_1, D_2, ..., D_K]$, where $D_k \in R^d$, is learned by solving the following constrained optimization problem:

$$\min_{D,\alpha} \sum_{i=1}^{N} \frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 \ s.t. \ \forall i, \ \|\alpha_i\|_0 < \epsilon \tag{4.10}$$

. In other words, the goal is to minimize the L_2 reconstruction error and guarantee the reconstruction coefficient vector to be sparse at the same time. Each D_k is called an atom of the dictionary, and α_i is called a sparse code. Due to the intractability of L_0 terms, it is a common practice to replace them with the L_1 norm. As a result, the actual objective function is:

$$\min_{\mathbf{D},\alpha} \sum_{i=1}^{N} \frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \lambda \|\alpha_i\|_1 \tag{4.11}$$

. One of the most frequently used dictionary learning methods is the K-SVD algorithm [157]. It is an iterative procedure with two alternating optimization steps: First fix the dictionary to solve for the sparse code, and then fix the sparse code to update the dictionary.

Our ultimate goal is to assign an identity label to a probe image (or frame) I_p. Although dictionary learning falls in the category of unsupervised learning algorithms, sparse codes have been used for classification in a number of different ways. A typical strategy is based on reconstruction errors. More specifically, we can calculate the difference vector $\mathbf{x}_c$ between I_p and every gallery image I_c, $c = 1, 2, ..., C$,

and compare the reconstruction errors:

$$ID(I_p) = \underset{c}{\operatorname{argmin}} \|\mathbf{x}_c - \mathbf{D}\alpha_c\|_2^2 \tag{4.12}$$

. Alternatively, one may choose to learn an additional extrapersonal dictionary $\mathbf{D}'$ following a similar procedure, and compare reconstruction error ratios:

$$ID(I_p) = \underset{c}{\operatorname{argmin}} \frac{\|\mathbf{x}_c - \mathbf{D}\alpha_c\|_2^2}{\|\mathbf{x}_c - \mathbf{D}'\alpha'_c\|_2^2} \tag{4.13}$$

4.4.2 Label-Consistent Dictionary Learning for Video-Based Face Recognition

It has been argued that separating dictionary learning from classier design may lead to sub-optimal solutions for the final classification task. In view of this, we follow the Label-Consistent K-SVD (LC-KSVD) algorithm [3] to jointly learn a generative shared dictionary and a discriminative projection matrix. Although the shared dictionary is composed of two sub-dictionaries corresponding to intra-personal and extra-personal differences respectively, the sparse code of any input difference vector is computed by using the complete set of atoms in the dictionary. This is different from the class-specific dictionaries in Section 4.4.1. On the other hand, a matrix $\mathbf{W} \in R^{2\times d}$ that encodes the discriminative information of the sparse codes is learned along with the shared dictionary. For the sparse codes $\mathbf{A} = [\alpha_1, \alpha_2, ..., \alpha_N]$ resulting from s set of intra-personal and extra-personal difference vectors, the projection $\mathbf{WA}$ is supposed to form two well-separated clusters. Aside from that, the LC-KSVD also looks for a linear transformation $\mathbf{B} \in R^{K\times d}$

which encourages the samples from the same class to be reconstructed using similar atoms, i.e. the entries in the sub-dictionary of that class. This constraint can be written in the form: $\mathbf{BX} = \mathbf{Q}$, where $\mathbf{Q} \in R^{K \times N}$ has a block diagonal form: The c-th block contains entry $Q_{ij}, i \in \mathbf{v}_c, j \in \mathbf{h}_c$, where v_c are the indices of atoms from class c (i.e. intra-personal or extra-personal) and $\mathbf{h}_c$ are the indices of training instances from class c. All the non-zero entries in $\mathbf{Q}$ are assigned with unit value. To summarize, the final optimization problem has the following form:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_2^2 + \mu\|\mathbf{Q} - \mathbf{BA}\|_2^2 + \sigma\|\mathbf{F} - \mathbf{WA}\|_2^2 + \lambda \sum_i \|\alpha_i\|_1 \tag{4.14}$$

, where the columns of $\mathbf{F} \in R^{2 \times N}$ are labels of the training instances in $\mathbf{X}$, represented using the 1-of-K coding scheme.

Solution to (4.14) can be converted to a typical K-SVD objective function:

$$\min_{\mathbf{D},\mathbf{A}} \|\widetilde{\mathbf{X}} - \widetilde{\mathbf{D}}\mathbf{A}\|_2^2 + \lambda \sum_i \|\alpha_i\|_1 \tag{4.15}$$

by defining $\widetilde{\mathbf{X}} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Q} \\ \mathbf{F} \end{pmatrix}$ and $\widetilde{\mathbf{D}} = \begin{pmatrix} \mathbf{D} \\ \mathbf{B} \\ \mathbf{W} \end{pmatrix}$. Therefore, we can conveniently apply K-SVD and extract $\mathbf{W}$ and $\mathbf{D}$ from the resulting augmented matrices.

According to a large body of empirical research, pose variations often cause within-class variance to exceed between-class variance in face recognition. Predictably, they present a great challenge to the intrapersonal/extrapersonal difference dictionary learning. Therefore, we choose to separate pose from other nuisance factors which case variations in the intrapersonal/extrapersonal domain. To this end, we first group the aligned training images according to face pose that has been

estimated along with the fiducial points in 4.3. The difference images are then calculated within each pose group and are used to learn pose-specific shared dictionaries $\{\mathbf{D}^m\}$, where m corresponds to the mixture index in Section 4.3. Naturally, to predict the class label (i.e. same-person or different-person) of a difference image with pose m at test time, only the dictionary $\mathbf{D}^m$ is relevant and will be activated in calculations. Therefore, we drop the mixture/pose superscript to avoid cluttered notation and keep the dependency on pose implicit.

In our work, calculating sparse codes for every frame pair is not only computationally expensive, but also unnecessary due to the significant temporal redundancy present in video signals. The redundancy can be removed by finding representative frames, which was often accomplished using the K-means algorithm. However, it is still an open problem to adaptively determine K at run-time, and it is obvious that a pre-determined K would be unsatisfactory considering the large variations of video contents. In view of that, we choose to fit a non-parametric Bayesian model to each video. The resulting model has infinite number of Gaussian mixtures controlled by a Dirichlet process $DP(\beta, H)$ [162], where β is the concentration parameter and H is the base probability measure. The mixture weights $\{\pi_k, k = 1, 2, ..., \infty\}$ are generated from the Griffiths-Engen-McClosky (GEM) process [163], i.e.:

$$\pi_k = \rho_k \prod_{l=1}^{k-1}(1 - \rho_l) \quad \rho_k \sim Beta(1, \beta) \tag{4.16}$$

. The mean and covariance parameters $\{\theta_k\}$ of the mixtures are sampled from H. Given a video V, we assume that each frame $\{I_f,\ f = 1, ..., F\}$ is assumed to be generated by first drawing a component label z_f from a Multinoulli distribution with

parameter $\{\pi_k, k = 1, 2, ..., \infty\}$ and then sample from a Gaussian distribution with parameter $\{\theta_k\}$. We adopt the variational inference approach to fit the model due to its efficiency. The posterior distribution $P(z_f|V)$ is used for clustering. By using the Dirichlet process mixture model, new clusters can be generated when more frames are observed, and there is no need to know number of clusters a priori.

After fitting the model, a video $\mathbf{V}$ with K clusters can be characterized by the set of cluster centers. We further exrtact feature vectors $\{\mathbf{v}^k,\ k = 1, 2, ..., K\}$ from these representative images. Both training and test videos go through this process. For the training videos, the intrapersonal features $\{\mathbf{x}_{In} = \mathbf{v}_i^m - \mathbf{v}_j^n\,, ID(\mathbf{V}_i) = ID(\mathbf{V}_j)\}$ and the extrapersonal ones $\{\mathbf{x}_{Ex} = \mathbf{v}_i^m - \mathbf{v}_j^n\,, ID(\mathbf{V}_i) \neq ID(\mathbf{V}_j)\}$ are employed to learn the dictionary $\mathbf{D}$ and the projection matrix W. At the test stage, we iterate over every probe-gallery video pair $\{\mathbf{V}_p, \mathbf{V}_g\}$ and calculate feature difference vectors $\{\mathbf{x}_{p,g}^{m,n} = \mathbf{v}_p^m - \mathbf{v}_g^n\}$ from the representative cluster centers. We then solve for the sparse representation of $\mathbf{x}_{p,g}^{m,n}$:

$$\alpha_{p,g}^{m,n} = \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{N} \frac{1}{2} \|\mathbf{x}_{p,g}^{m,n} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \tag{4.17}$$

. As mentioned earlier, there is an implicit pose index in the equations above. That is, we only calculate feature vector differences for face images with the same pose, and activate the dictionary of the corresponding pose to compute sparse codes.

For video-based recognition, we have: $ID(\mathbf{V}_p) = \underset{g}{\operatorname{argmax}}\, s(p, g)$, where

$$s(p, g) = \sum_{m=1}^{M} \sum_{n=1}^{N} \mathbf{1}(\mathbf{t}_1 \mathbf{W} \alpha_{p,g}^{m,n} > \mathbf{t}_0 \mathbf{W} \alpha_{p,g}^{m,n})/MN \tag{4.18}$$

. Here, we use $\mathbf{t}_0 = [0, 1]^T$ and $\mathbf{t}_1 = [1, 0]^T$ to denote the 1-of-K coding label for intra-personal and extra-personal class, respectively. $\mathbf{1}(\cdot)$ is the indicator function.

One of the attractive features of the proposed algorithm is that it naturally fits in with the verification protocol. In a hard decision scheme, for each video pair $\{\mathbf{V}_p, \mathbf{V}_g\}$, we apply majority voting on top of the binary "same person/different person" results of the frame pairs. This will yield a single operating point on the ROC curve. Alternatively, we may adopt a soft decision rule. The entry of the similarity matrix is the same as the $s(p, g)$ defined in (4.18).

4.5 Experiments

In this section, we first present the results of our face fiducial points localization algorithm. Then we compare our video-based face recognition methods with existing algorithms on three public databases.

4.5.1 Facial Feature Localization

We trained our facial feature detector and evaluated its performance on a subset of the Annotated Facial Landmarks in the Wild (AFLW) database [164]. The database contains about 25,000 face images downloaded from Flickr, each manually annotated with up to 21 fiducial points. There are 5872 and 2000 face images in the selected training set and the test set, respectively. They are mutually exclusive. Some example images from the database are shown in Figure 4.2. We cropped the face region using the response of a Viola-Jones face detector and normalize it to 60×60. The training data were partitioned into groups according to pose. Although filter responses were computed for M mixture components at test time, we trained

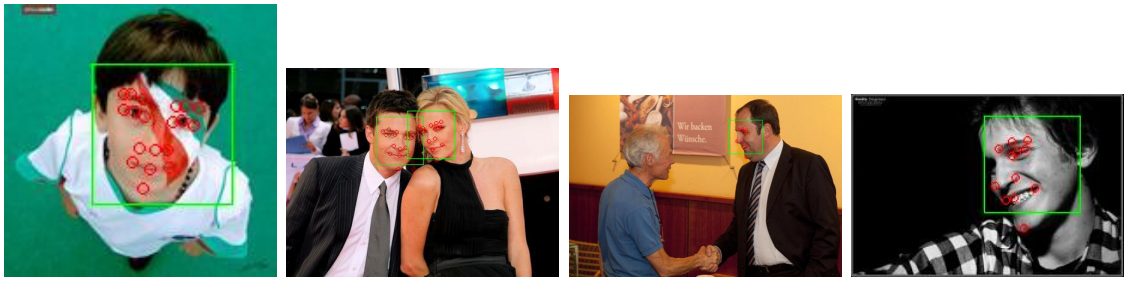


Figure 4.2: Example images from the AFLW database. Red circles mark the annotated face fiducial points.

$\frac{M-1}{2}$ of them by utilizing the symmetric property of a human face and mirroring the left-posed face images. Within each group of data, we collected statistics of L to determine the configuration space $\mathcal{Z}$. The reference algorithms used for comparison were the DPM-based one proposed in [104] and the one based on the Haar feature + Gaussian mixture tree [76]. The localization error was measured by the average distance (in pixels) between the predicted fiducial points and the ground truth ones, and normalized by inter-ocular distance. As shown in Figure 4.3, the proposed facial feature localization algorithm outperforms the two reference algorithms. However, the DPM detector is able to provide face detection output that is not supported by our method. It has also been observed that for large poses, the advantage of the proposed approach in localization accuracy is more evident.

4.5.2 Video-Based Face Recognition

Youtube Celebrity Video Database: The Youtube Celebrity database [51] has been widely adopted for evaluating the video-based face recognition algorithms. The database contains 1910 Youtube video clips of 47 subjects. Most of the videos were

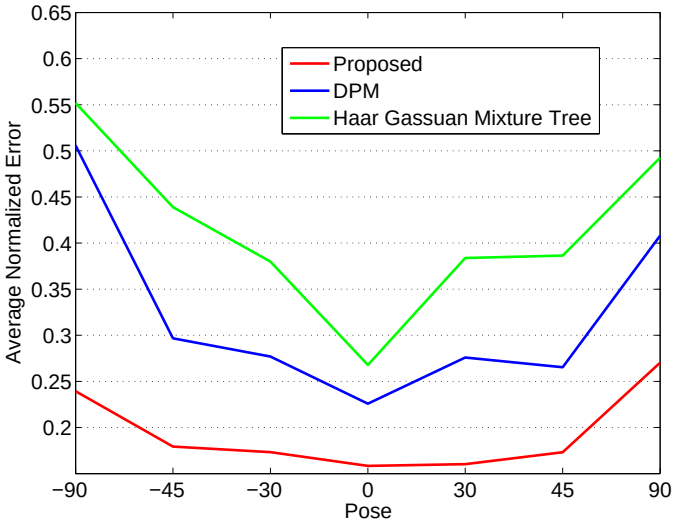


Figure 4.3: Face fiducial point detection results on the AFLW database.

extracted from news TV or movies, and hence exhibit large pose and illumination variations. The low resolution of the videos also poses a challenge to face recognition. In other words, this database aims to test the performance of VFR algorithms under uncontrolled settings. We follow the protocol in [41, 165, 45], i.e., randomly choosing 3 clips per subject as galleries and 6 per subject as probes.

Honda UCSD Database: The Honda UCSD database [32] consists of 59 videos of 20 subjects. The videos are divided into a training set which contains one video per subject and a testing set which contains 1 to 4 videos per subject. Each video sequence is recorded in an indoor environment at 15 frames/second and lasts at least 15 frames. Faces in the database undergo significant head motions and expression variations.

Buffy Database: This dataset consists of 639 face tracks from the TV series "Buffy the Vampire Slayer". We removed the face tracks whose id is labeled as unknown

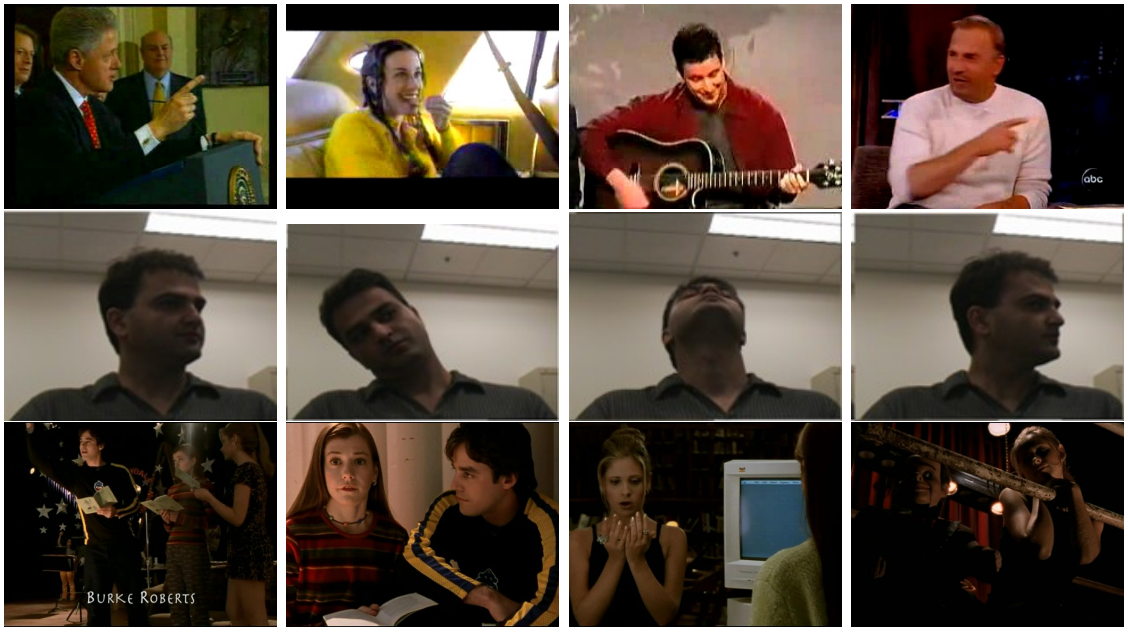


Figure 4.4: Example frames from the three public video-based face recognition databases: Youtube Clebrity (top row), Honda UCSD (middle row) and Buffy (bottom row).

characters, leaving a subset of 483 face tracks for 8 main characters. Following [166], they are separated into a training set of size 227 and a test set of size 256.

We show some example frames from the three video databases in Figure 4.4. For the Youtube Celebrity Video database and the Honda UCSD database, we applied the tracking-by-detection method as described in Section 4.3 to localize the face. For the Buffy dataset, we used the face tracks provided by the ground truth directly. We then simultaneously detected facial fiducial points and estimated the face pose using the proposed structural-SVM detector. The result was then employed to align the face region to a canonical frame pre-specified for the corresponding pose. We calculated the self-quotient image to normalize the illumination. Pose-specific

masks were imposed to suppress the background pixels. LBP and TP-LBP features were extracted and concatenated to form the feature vector. PCA was applied to reduce the dimension of feature vector to 400.

We trained our shared dictionary under two different settings. In the first one, the dictionary was learned from each database's own training set. We call this the same-database dictionary mode. Alternatively, because the intra-personal/extra-personal face variations are generic, we can learn a dictionary using training data of an entirely different set of subjects. We call this second case the cross-database dictionary mode. The number of intra-personal or extra-personal feature vector pairs that can be used for training is in $O(NK^2)$ and $O(N^2K^2)$ respectively, where N is the number of subjects and K is the average number of clusters discovered by the Dirichlet process Gaussian mixture model from the videos of the same subject. The potential number is huge for a large database like the Youtube Celebrity Video dataset, especially when we are concerned with the extra-personal pairs. This is also true if we are to learn a dictionary from an external database. On the other hand, the number of intra-personal pairs generated from a small training set, such as that of the Honda/UCSD database, might be insufficient for learning a dictionary. In the former case, we pruned candidate pairs by keeping only around 4000 samples in each of the intra-personal and extra-personal training set. We attempted to distribute the samples as evenly as possible and avoided only using samples from a small subset of videos. In the latter case, we augmented the pool of intra-personal pairs with samples from external data. To train the dictionary in the cross-database mode, we used the LFW database[167], which has 5749 people, among which 1680 subjects

Table 4.1: Comparison of Video-Based Face Recognition Results on the Youtube Celebrity Video and the Honda/UCSD database

Method	**Youtube Celebrity**	**Honda/UCSD**
MSM[22]	61.1	92.5
MMD[39]	62.9	97.1
MDA[41]	65.3	**100.0**
CHISD[27]	66.3	90.5
SANP[28]	68.4	93.6
COV + PLS[153]	70.1	**100.0**
MA[165]	74.6	99.0
MSSRC[45]	80.8	-
Proposed-I(Same-Database)	**81.9**	97.4
Proposed-II(Cross-Database)	78.6	97.4

Table 4.2: Comparison of Video-Based Face Recognition Results on the Buffy database

Method	**Buffy Database**
LDML[166]	85.9
MSSRC[45]	86.3
Proposed-I(Same-Database)	**88.3**
Proposed-II(Cross-Database)	85.2

have two or more images. We expect that the different variations covered by the database can lead to a dictionary with good generalization property.

We compare the proposed methods with several existing VFR algorithms on the three databases. It is apparent from a careful study of reported experimental results that not all the algorithms are compared on all the databases. On the Youtube Celebrity Video database and the Honda/UCSD database, the compared existing algorithms include: Mutual Subspace Method (MSM)[22], Manifold-Manifold Distance (MMD)[39], Manifold Discriminant Analysis (MDA)[41], Convex Hull based Image Set Distance (CHISD)[27], Sparse Approximated Nearest Point (SANP)[28], Covariance Partial Least Square (COV + PLS)[153], Manifold Alignment (MA)[165] and Mean Sequence Sparse Representation-based Classification (MSSRC) [45]. On the Buffy database, we compare with Logistic Discriminant-based Metric Learning (LDML) [166] and MSSRC. The results are presented in Tables 4.1 and 4.2. As

shown in tables, in all three databases, both of the same-database and the cross-database dictionary modes of the proposed algorithm achieve comparable results with respect to the state-of-the-art. On the most challenging Youtube Celebrity Video database, our method produces slightly better results than the one most recently reported in [45] and outperforms the other algorithms by a large margin. The relative lower classification rate on the Honda/UCSD database may be due to insufficient training samples. A noticeable fact is that using the cross-database dictionary learned from the external database usually leads to a degraded performance. This is consistent with our intuition that cross-domain learning is in general a more difficult problem. But the cross-domain dictionary is advantageous in terms of scalability and flexibility, as the training difference vectors are complementary to each other and can be shared. Finally, the proposed framework naturally supports the face verification protocol. Therefore, we also investigate the performance of our algorithm in the verification mode that is described in Section 4.4. The result on the Youtube Celebrity Video database is plotted in the form of ROC curves in Figure 4.5. We compare with the MMD and MDA because their outputs are distances, from which the ROC curves can be conveniently generated.

4.6 Conclusion

We introduced a novel framework for video-based face recognition. It is based on the generic concept of intra-personal/extra-personal variations, and hence leads to greater scalability. We exploited the strength of sparse codings in classification

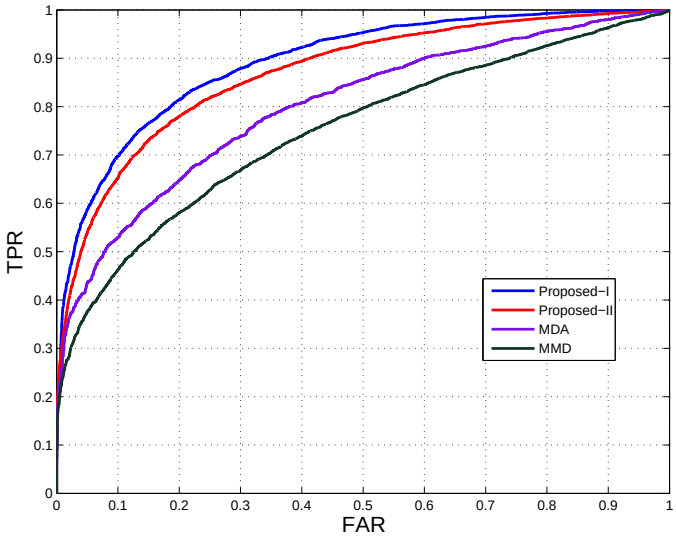


Figure 4.5: The face verification results on the Youtube Celebrity Video database.

and learned a discriminative dictionary from these variations. In addition, we presented a facial feature detection method for accurate face alignment in unconstrained videos. Our scheme is flexible enough to work in both identification and verification modes. It can also be trained and tested on different databases. We conducted experiments on three public databases and demonstrated the performance of the proposed approach through comparison with existing algorithm.

Chapter 5

Video-Based Face Recognition Using a Camera Network

5.1 Introduction

We now extend our focus to the multi-view video case. Single-view based object recognition is inherently affected by information loss that occurs during image formation. Although there exist many works addressing this problem, pose variation remains as one of the major nuisance factors for face recognition. In particular, self-occlusion of facial features, as the pose varies, raises fundamental challenges to designing robust face recognition algorithms. A promising approach to handle pose variations and their inherent challenges is the use of multi-view data.

In recent years, multi-camera networks have become increasingly common for biometric and surveillance systems. Having multiple viewpoints alleviates the drawbacks of a single viewpoint since the system has more information at its disposal. For example, in the context of face recognition, having multiple views increases the chances of the person being in a favorable frontal pose. However, to reliably and efficiently exploit the multi-view video data, we often need to estimate the pose of the person's head. This could be done explicitly by computing the actual pose of the person to a reasonable approximation, or implicitly by using a view selection algorithm. While there are many methods for multi-view pose estimation [168, 169], solving for the pose of a person's head is still a hard problem, especially when the

resolution of the images is poor and the calibration of cameras (both external and internal) is not sufficiently precise to allow robust multi-view fusion. Such a scenario is especially true in the context of surveillance.

Face recognition using a multi-camera network is the focus of this chapter. At this point, it is worth noting that the problem we study goes beyond face recognition across pose variations. In our setting, at a given time instant, we obtain multiple images of the face in different poses. Invariably these images could include a mix of frontal, non-frontal images of the face or in some cases, a mix of non-frontal images. This makes registration of the faces extremely important. Registration can be done once we decide to impose a 3D model onto the face. However, registration to a 3D model (essentially, aligning eyes to eyes, nose to nose, etc.) is hard and computationally intensive for low-resolution imagery. Toward this end, we choose to use a spherical model of the face and a feature that is insensitive to pose variations.

In this chapter, we propose a robust feature for multi-view recognition that is insensitive to pose variations[1]. For a given set of multi-view video sequences, we first use a particle filter to track the 3D location of the head using multi-view information. For each video frame, we build the texture map associated with the face using a spherical head model. Given that we have the 3D location of the head from the tracking algorithm, we back-project the image intensity values from each of the views onto the surface of the spherical model, and construct a texture map for the whole face. We then compute a Spherical Harmonic (SH) transform of the texture map, and construct a robust feature that is based on the properties of the SH projection. Building rotational tolerances into our feature allows us to completely bypass the

pose estimation step. For recognition with videos, we exploit the ensemble feature similarity which is measured by the limiting Bhattacharyya distance of features in the Reproducing Kernel Hilbert Space. The proposed approach outperforms traditional features and algorithms on a multi-view video database collected using a camera network.

5.2 Related Work

The term *multi-view face recognition,* in a strict sense, only refers to situations where multiple cameras acquire the subject (or scene) simultaneously and an algorithm collaboratively utilizes the acquired images/videos. But the term has frequently been used to recognize faces across pose variations. This ambiguity does not cause any problem for recognition with (still) images; a group of images simultaneously taken with multiple cameras and those taken with a single camera but at different view angles are equivalent as far as pose variations are concerned. However, in the case of video data, the two cases diverge. While a multi-camera system guarantees the acquisition of multi-view data at any moment, the chance of obtaining the equivalent data by using a single camera is unpredictable. Such differences become vital in non-cooperative recognition applications such as surveillance. For clarity, we shall call the multiple video sequences captured by synchronized cameras

[1]In many contexts such as camera pose estimation, pose typically refers to the 3D translation and 3D rotation of the camera/object. However, in face recognition, pose typically refers only to the 3D rotation of face with respect to a reference orientation. We follow this convention. For most of this chapter, we use the term pose and rotation interchangeably.

a multi-view video, and the monocular video sequence captured when the subject changes pose, a single-view video. With the prevalence of camera networks, multi-view surveillance videos have become more and more common. Nonetheless, most existing multi-view video face recognition algorithms exploit single-view videos.

Still image-based recognition: There is a large body of research on still image-based multi-view face recognition. Existing algorithms include those based on view synthesis [81, 82, 83, 84, 170], 3D model construction [55, 56, 59], subspace or manifold analysis[85, 86, 171], regularized regression [172], stereo matching [173, 174] and local feature matching [87, 88, 89, 90, 91]. In recent years, local patch/feature-based approaches have become popular due to their effectiveness in handling pose variations. Cao et al. [175] compare the local descriptors in a pose-adaptive way: they estimate the poses of the pair of input faces images and select an SVM classifier customized for that pose combination to perform verification. Yin et al. [176] generate a collection of generic intra-person variations for local patches. Given a pair of face images to verify, they look up in the collection to "align" the face part's appearance in one image to the same pose and illumination of the other image. This method will also require the poses and illumination conditions to be estimated for both face images. This "generic reference set" idea has also been used to develop the holistic matching algorithm in [177], where the ranking of look-up results forms the basis of matching measure. There are also works which handles pose variations implicitly without estimating the pose explicitly. For example, by modeling the location-augmented local descriptors using a Gaussian Mixture Model,

Li et al. [178] perform probabilistic elastic matching on a pair of face images even when large pose variations exhibit.

Video-based recognition: Video contains more information than still images. A straightforward way to handle single-view videos is to take advantage of the data redundancy and perform view selection. Li et al. [92] employ a combination of skin color detector and edge feature-based SVM regression to localize face candidates and estimate their poses. Then, for each candidate, a face detector specific to that pose is applied to determine if it is a face. Only the frontal faces are retained for recognition. The algorithm in [93] also relies on an SVM to select frontal faces from video for recognition. The continuity of pose variation in video has inspired the idea of modeling face pose manifolds [32, 35]. The typical method is to cluster the frames of similar pose and train a linear subspace to represent each pose cluster. Here, the piecewise linear subspace model is an approximation to the pose manifold. Wang et al. [39] grow each such linear subspace gradually from a seed sample to include more and more nearest neighbors, until the linearity condition is violated. The linearity is measured as the ratio of geodesic distance to Euclidean distance, and the distances are calculated between a candidate neighbor and each existing sample in the cluster. They define the manifold-manifold distance as the distance between the closest subspace pair from the two manifolds, and the subspace distance is defined as a weighted sum of canonical correlations and exemplar distance. Also assuming that all images of the same person lie on a manifold, Arandjelovic et al. [179] model face videos using Gaussian Mixture Models. The manifold-manifold

distance is then measured using the KL divergence between the Gaussian mixtures. Single-view videos have also been modeled using Hidden Markov Models [49], or ARMA models [53]. 3D face models can be estimated from single-view videos as done in [57, 58, 59]. The 3D model can be then used in a model-based algorithm (e.g. [63]) to perform face recognition.

Multi-view-based recognition: In contrast to single-view/video-based face recognition, there are relatively a smaller number of approaches for recognition using multi-view videos. In [94], although both the gallery and the probe are multi-view videos, they are treated just like single-view sequences. Frames of a multi-view sequence are collected together to form a gallery or probe set. The frontal or near-frontal faces are picked by the pose estimator and retained, while others are discarded. The recognition algorithm is frame-based PCA and LDA fused by the sum rule. In [95], a three-layer hierarchical image-set matching technique is presented. The first layer associates frames of the same individual taken by the same camera. The second layer matches the groups obtained in the first layer among different cameras. Finally, the third layer compares the output of the second layer with the training set, which is manually clustered using multi-view videos. Though multi-view data is used to deal with occlusions when more than one subject is present, pose variations are not effectively addressed in this work. Ramnath et al. [96] extend the AAM framework to the multi-view video case. They demonstrate that when 3D constraints are imposed, the resulting 2D+3D AAM is more robust than the single view case. However, recognition was not attempted in this work. Liu and

Chen [98] use geometrical models to normalize pose variations. By back-projecting a face image to the surface of an elliptical head model, they obtained a texture map which was then decomposed into local patches. The texture maps generated from different images were compared in a probabilistic fashion. Our work shares some similarities with theirs in the texture mapping stage. This method has been extended to multi-view videos in [99]. The texture mapping procedure was further elaborated by adding a geometric deviation model to describe the mapping error. However, tracking, texture mapping and recognition steps were all carried out for each view independently.

As mentioned earlier, almost all of the above referenced algorithms incorporate a pose estimation or model registration step, or even assume that pose is known a priori. The problem naturally arises when we try to compare face appearances described by pose-sensitive features.

Video processing in multi-camera networks: Camera networks have been extensively used for surveillance and security applications [180]. Research in this field has been focused on distributed tracking, resource allocation, activity recognition and active sensing. Yoder et al. [97] track multiple faces in a wireless camera network. The observations of multiple cameras are integrated using a minimum variance estimator and tracked with a Kalman filter. Song and Roy-Chowdhury present a multi-objective optimization framework for tracking in a camera network in [181]. They adapt the feature correspondence computations by modeling the long-term dependencies between them and then obtain statistically optimal paths

for each subject. Song et al. [182] incorporate the concept of consensus into distributed camera networks for tracking and activity recognition. The estimate made by each camera is shared with its local neighborhood, and the consensus algorithms combine the decisions from single cameras to make a network-level decision. A detailed survey on video processing in camera networks can be found in [183].

Spherical harmonics (SH) in machine vision: Basri and Jacobs [184] use SH to model Lambertian objects under varying illumination. Specifically, they proved that the reflectance function produced by convex, Lambertian objects under distant, isotropic lighting can be well approximated using the first nine SH basis functions. Ramamoorthi [185] revealed the connection between SH and PCA, showing that the principal components are equal to the SH basis functions under appropriate assumptions. Zhang and Samaras [186] proposed an algorithm to estimate the SH basis images for a face at a fixed pose from a single 2D image based on statistical learning. When the 3D shape of the face is available, the SH basis images can be estimated for test images with different poses. Yue et al. [187] adopted a similar strategy where the distribution of SH basis images is modeled as Gaussian and its parameters are learned from a 3D face database. Note that all these works are based on Lambertian reflectance model. As a result, they require a 3D face model and face pose estimation to infer the face appearance. In contrast, we use an SH-based feature to directly model face appearance rather than the reflectance function, and hence do not require a 3D face surface model or a pose estimation step.

5.3 Robust Feature

The robust feature presented here is based on the theory of spherical harmonics. Spherical harmonics are a set of orthonormal basis functions defined over the unit sphere, and can be used to linearly expand any square-integrable function on $\mathbb{S}^2$ as:

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} f_{lm} Y_{lm}(\theta, \phi), \tag{5.1}$$

where $Y_{lm}(\cdot, \cdot)$ defines the SH basis function of degree $l \geq 0$ and order $m \in (-l, -l+1, \ldots, l-1, l)$. f_{lm} is the coefficient associated with the basis function Y_{lm} for the function f. Note that we are using the spherical coordinate system. $\theta \in (0, \pi)$ and $\phi \in (0, 2\pi)$ are the zenith and azimuth angles, respectively. There are $2l + 1$ basis functions for a given order l [188].

The SH basis function for degree l and order m has the following form:

$$Y_{lm}(\theta, \phi) = K_{lm} P_l^m(\cos\theta) e^{im\phi} \tag{5.2}$$

where K_{lm} denotes a normalization constant such that:

$$\int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} Y_{lm} Y_{lm}^* d\phi d\theta = 1 \tag{5.3}$$

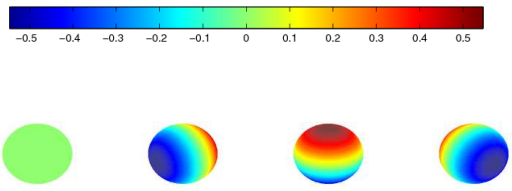
Here, $P_l^m(x)$ are the associated Legendre functions.

In this work, we are interested in modeling real-valued functions (eg. texture maps) and thus, we are more interested in the real Spherical Harmonics which are

defined as

$$Y_l^m(\theta,\phi) = \begin{cases} Y_{l0} & \text{if } m = 0 \\ \frac{1}{\sqrt{2}}(Y_{lm} + (-1)^m Y_{l,-m}) & \text{if } m > 0 \\ \frac{1}{\sqrt{2}i}(Y_{l,-m} - (-1)^m Y_{lm}) & \text{if } m < 0 \end{cases} \tag{5.4}$$

The real SHs are also orthonormal and they share most of the important properties of the general Spherical Harmonics. For the rest of the chapter, we will use the word "spherical harmonics" to refer exclusively to real SHs. We visualize the SH for degree l = 0, 1, 2 in Fig. 5.1.



(a) l=0, m=0 (b) l=1, m=-1 (c) l=1, m=0 (d) l=1, m=1

(e) l=2, m=-2 (f) l=2, m=-1 (g) l=2, m=0 (h) l=2, m=1 (i) l=2, m=2

Figure 5.1: Visualization of the first three degree of Spherical Harmonics.

As with Fourier expansion, the SH expansion coefficients f_l^m can be computed as:

$$f_l^m = \int_\theta \int_\phi f(\theta,\phi) Y_l^m(\theta,\phi) d\theta d\phi \tag{5.5}$$

The expansion coefficients have a very important property which is directly related to our "pose free" face recognition application.

Proposition: If two functions $f(\theta, \phi)$ and $g(\theta, \phi)$, defined on $\mathbb{S}^2$, are related by a rotation $R \in SO(3)$, i.e. $g(\theta, \phi) = f(R(\theta, \phi))$, and their SH expansion coefficients are f_l^m and g_l^m, respectively, the following relationship exists:

$$g_l^m = \sum_{m'=-l}^{l} D_{mm'}^l f_l^{m'} \tag{5.6}$$

and the $D_{mm'}^l$s satisfy:

$$\sum_{m'=-l}^{l} (D_{mm'}^l)^2 = 1 \tag{5.7}$$

In other words, (5.6) suggests that after rotation, the SH expansion coefficients at a certain degree l are linear combinations of those before the rotation, and coefficients at different degrees do not affect each other. This can also be represented in a matrix form:

$$\begin{pmatrix} f_0^0 \\ f_1^{-1} \\ f_1^0 \\ f_1^1 \\ f_2^{-2} \\ \vdots \\ \vdots \\ f_2^2 \\ \vdots \end{pmatrix} = \left[\begin{array}{c|ccc|ccccc|c} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & . \\ \hline 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ \hline 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ \hline . & . & . & . & . & . & . & . & . & . \end{array}\right] \begin{pmatrix} g_0^0 \\ g_1^{-1} \\ g_1^0 \\ g_1^1 \\ g_2^{-2} \\ \vdots \\ \vdots \\ g_2^2 \\ \vdots \end{pmatrix}, \tag{5.8}$$

where the x denotes non-zero entries corresponding to appropriate $D^l_{mm'}$ values.

This proposition is a direct result of the following Lemma [188] [189].

Lemma: Denote by E_l the subspace spanned by $Y_l^m(\theta, \phi)$, $m = \{-l, \ldots, l\}$, then E_l is an irreducible representation for the rotation group SO(3).

The proof of the proposition is as follows:

Proof Let us denote the lth degree frequency component as $f_l(\theta, \phi)$:

$$f_l(\theta, \phi) = \sum_{m=-l}^{l} f_l^m Y_l^m(\theta, \phi) \tag{5.9}$$

, then $f_l(\theta,\phi) \in E_l$. According to the Lemma:

$$\begin{aligned} g_l(\theta,\phi) &= R(f_l(\theta,\phi)) \\ &= R(\sum_{m=-l}^{l} f_l^m Y_l^m(\theta,\phi)) \\ &= \sum_{m=-l}^{l} f_l^m R(Y_l^m(\theta,\phi)) \\ &= \sum_{m=-l}^{l} f_l^m \sum_{m'=-l}^{l} D_{mm'}^l Y_l^{m'}(\theta,\phi) \\ &= \sum_{m'=-l}^{l} \sum_{m=-l}^{l} f_l^m D_{mm'}^l Y_l^{m'}(\theta,\phi) \end{aligned} \tag{5.10}$$

Equation (5.6) follows by comparing (5.10) with

$$g_l(\theta,\phi) = \sum_{m'=-l}^{l} g_l^{m'} Y_l^{m'}(\theta,\phi) \tag{5.11}$$

As for Equation (5.7), notice that Y_l^ms and $Y_l^{m'}$ are both orthonormal basis:

$$\begin{aligned} RHS &= 1 \\ &= \int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} Y_l^m Y_l^m d\phi d\theta \\ &= \sum_{m'=-l}^{l} (D_{mm'}^l)^2 \int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} Y_l^{m'} Y_l^{m'} d\phi d\theta \\ &= \sum_{m'=-l}^{l} (D_{mm'}^l)^2 \\ &= LHS \end{aligned} \tag{5.12}$$

We further look into a energy vector associated with a $f(\theta,\phi)$ defined on $\mathbb{S}^2$ as:

$$e_f = (\|\mathbf{f}_0\|_2, \|\mathbf{f}_1\|_2, \|\mathbf{f}_l\|_2, \ldots), \tag{5.13}$$

where $\|\cdot\|_2$ denotes the ℓ_2-norm, and $\mathbf{f}_l$ consists of all the SH decomposition coefficients of $f(\theta,\phi)$ at degree l:

$$\mathbf{f}_l = \{f_l^m, m = -l, \ldots, l\}. \tag{5.14}$$

Equation (5.7) guarantees that e_f is invariant when $f(\theta, \phi)$ is rotated. In practice, we find that subsequent normalization of e_f with respect to total energy increases reliability. This results in a feature which describes the spectrum of the SH coefficients. We refer to it as the SH spectrum feature.

The specific form of the function $f(\theta, \phi)$ varies with applications and is often numerically defined for sampled points on the surface of a sphere. In our multi-view face recognition scenario, $f(\theta, \phi)$ is the face appearance as represented by a texture map/template. To be more specific, we use a sphere to approximate the human head and the relevant image regions in multi-view data are mapped onto the surface of the sphere according to projective geometry. This procedure will be described in detail in Section 5.4. Note that the spherical model is different from the 3D face model in a general sense as one does not have to estimate the surface normals. Using a simple spherical model is often sufficient when we deal with low-resolution images and hence, is suitable for camera networks. Constructing a reasonable 3D face model usually requires much higher image resolution and computations. More importantly, this model enables us to set up a connection between multi-view face image and SH representation. Indeed, even when the face undergoes extreme pose variations, the SH spectrum feature extracted from the texture maps remains stable, leading to pose-robust face recognition. Note that the normalization step in feature extraction is equivalent to assuming that all the texture maps have the same total energy,

and in a loose sense functions as an illumination normalization step. Although this means that skin color information is not used for recognition, experimental results are good. Fig. 5.2 shows an example. One can see that features extracted from the same subject's texture map are very close even when large pose variations are present, and they are much closer than those extracted from different subjects but under the same pose.

Another advantage of the SH spectrum feature is its ease of use. There is only one parameter to be determined, namely the number of degrees in the SH expansion. Apparently, a trade-off exists for different choices of parameter values: A higher degree number means better approximation, but it also comes with a price of more expensive computational cost. In Fig. 5.3, we visualize a 3D head texture map as a function defined on $\mathbb{S}^2$, and its reconstruction resulting from 20, 30 and 40 degree SH transform respectively. The ratio of computation time for the 3 cases is roughly 1:5:21. (On a PC with Xeon 2.13GHz CPU, it takes roughly 1.2 seconds to do a 20 degree SH transform for 18050 points.) We have empirically observed that the 30-degree transform usually achieves a reasonable balance between approximation error and computational cost.

5.4 Multi-Camera Tracking and Texture Mapping

In this section, we describe a robust multi-view tracking algorithm based on Sequential Importance Resampling (SIR) (particle filtering) [14]. Tracking is an essential stage in camera-network-based video processing. It automates the local-

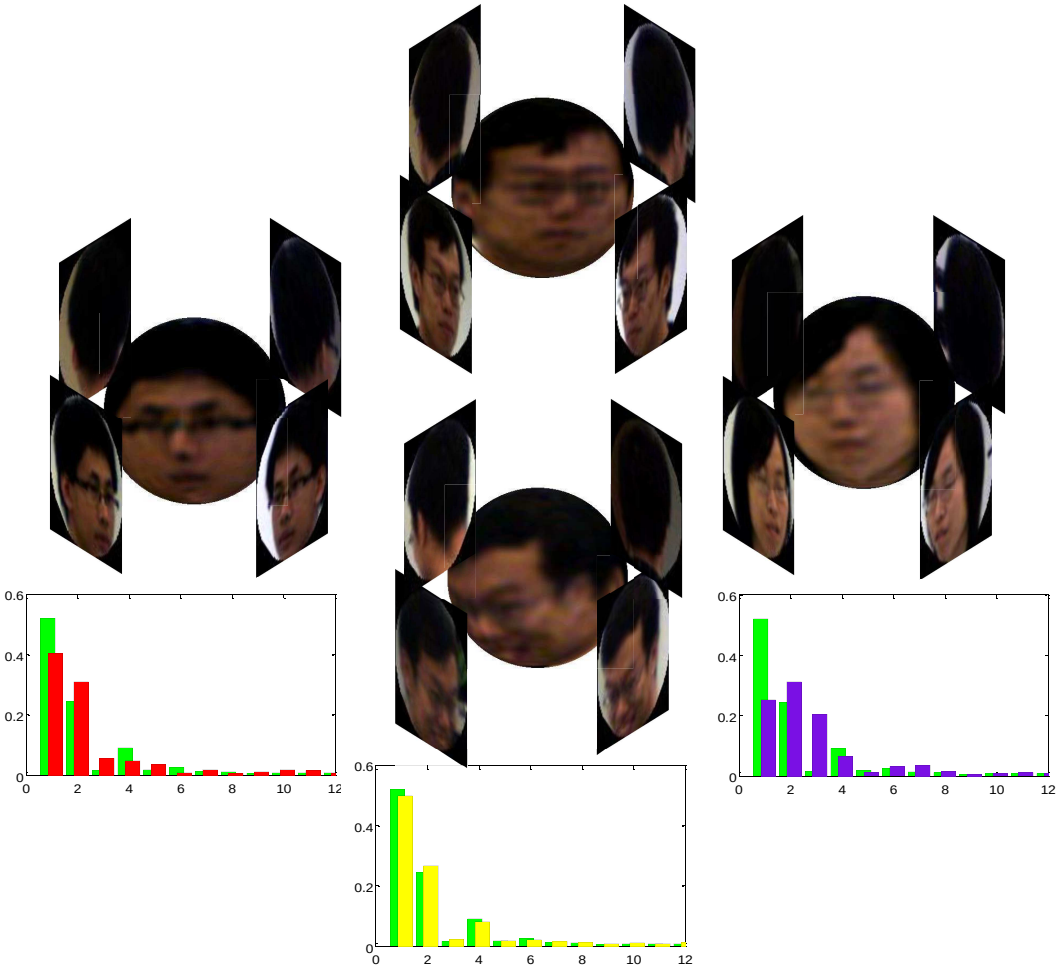


Figure 5.2: **Robust features based on Spherical Harmonics.** The texture of each model is constructed from multi-view images captured by four synchronized cameras. The top and bottom models correspond to the same subject, but the capture time of the two sets of images are separated by a time span of more than 6 months. Note that we intentionally rotate the bottom model by 180° so that readers can see that it is the same subject as in the top one. Therefore their actual pose difference is even larger than the one shown. The green bars in the three bar graphs are the same feature vector extracted from the top model. For visualization considerations, only the first 12 elements of the feature vector are plotted here.

Figure 5.3: **Comparison of the Reconstruction Qualities with SH Coefficients** The images from left to right are: the original 3D head texture map, the texture map reconstructed from 40-degree, 30-degree and 20-degree SH coefficients, respectively. Note that we interpolated the surface points for a better visualization quality.

ization of the face and has direct impact on the performance of the recognition algorithm. Recall that the proposed SH spectrum feature is extracted from the texture map of the face under a spherical head model. The tracking module, together with a texture mapping step, describes the entire feature extraction process (see Fig. 5.4).

5.4.1 Multi-View Tracking

To fully describe the position and pose of a rigid 3D object, we usually need a 6-D representation ($\mathbb{R}^3 \times SO(3)$), where the 3-D real vector space is used to represent the object's location, and $SO(3)$ is used to represent the object's rotation. It is well known that higher the dimensionality of the state space is, the harder the tracking problem becomes. This is especially true for search-algorithms like SIR since the number of particles typically grows dramatically for high-dimensional state spaces.

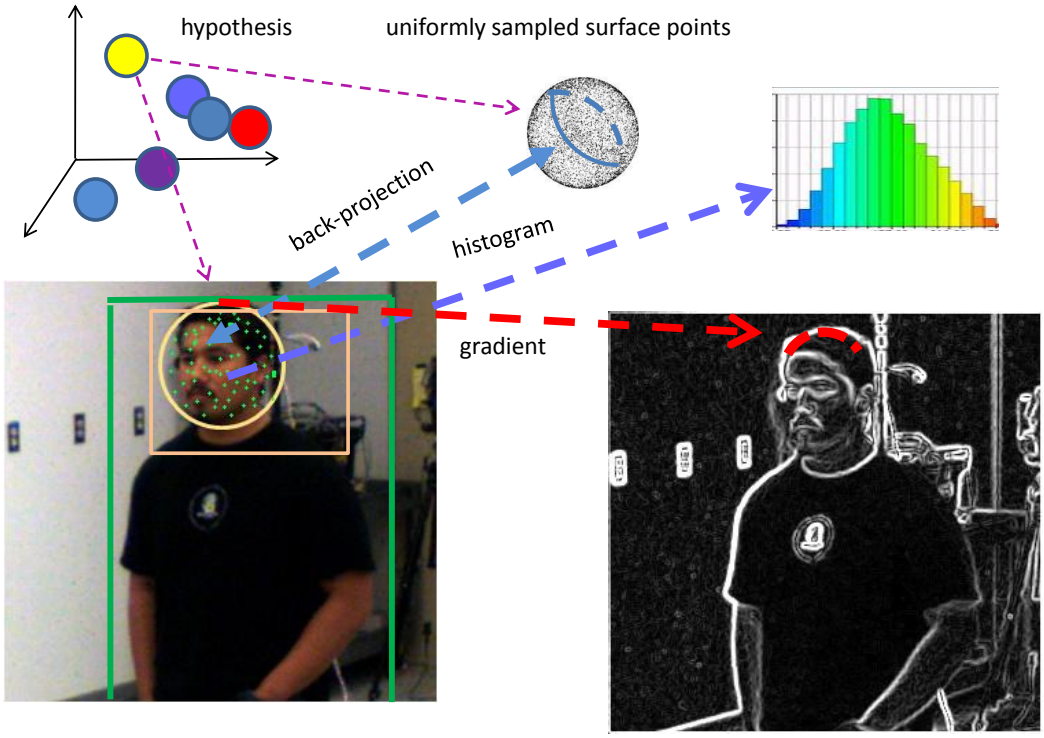


Figure 5.4: **The Multi-Cue Tracking Algorithm and Back-Projection.** The yellow circle is the boundary of the head's image for a certain hypothesis state vector. The green and orange rectangles mark the human body detection result and the estimated range of head center's projection, respectively. Green dots are the projections of model's surface points. The navy-blue curve on the sphere highlights the boundary of the visible hemisphere. Note that we draw tracking and back-projection together just for illustration. In actual case, only the MAP estimate of the state vector will be back-projected to construct the texture map.

However, given that our eventual recognition framework is built on the robust feature derived using SH representation under the diffuse lighting assumption, it suffices that we track only the location of the head in 3D. Hence, the state space for tracking $\mathbf{s} = (x, y, z)$ represents only the position of a sphere's center, disregarding any orientation information. Initialization of the tracker can be solved through face detection (For example, the cascaded Haar-feature detector in [102]) applied to the first frame and followed by multi-view triangulation.

The state transition model $P(\mathbf{s}_t|\mathbf{s}_{t-1})$ is modeled as a Gaussian distribution $\mathcal{N}(\mathbf{s}_t|\mathbf{s}_{t-1}, \sigma^2\mathbf{I})$. We found that the tracking result is relatively insensitive to the specific value of σ and have fixed it in all of our experiments.

The observation model $P(O_t|\mathbf{s}_t)$ of the tracker is based on multiple cues such as a histogram, the gradient map and a geometric constraint.

Histogram: To evaluate the image likelihood for a hypothesized state vector $\mathbf{s}_t^i$, we assume a weak-perspective camera model and calculate the image of the spherical model on the jth camera's image plane, which is a disk-like region E_j^i (We shall use the subscript j to indicate the jth view). A normalized 3D histogram in RGB space is built from this image region. Its difference with the template, which is set up at the first frame through the same procedure and subject to adaptive update thereafter, is measured by the Bhattacharyya distance. This defines the first cue matching function $\phi(O_t, \mathbf{s}_t^i)$.

Gradient map: On the circular perimeter of the model's image, we select the 90°

arc segment on the top, superimposing it on the horizontal and vertical gradient map of $I_{t,j}$. Despite various shapes of human heads, this part of the boundary turns out to reliably coincide with an arc. Therefore, if the state vector is a good match to the ground truth, we expect the magnitude of the image gradient response along this arc segment to be strong and its direction to be perpendicular to the tangent directions [190]. Consequently, we formulate the second cue matching score as:

$$\varphi(O_t, \mathbf{s}_t^i) = \frac{1}{r_j^i} \sum_{m=1}^{M} |\mathbf{n}_m \cdot \bigtriangledown \mathbf{I}_m|, \tag{5.15}$$

where r_j^i is the radius of E_j^i measured in number of pixels, $\mathbf{n}_m$ is the normal vector of the m-th pixel on the arc, and $\bigtriangledown \mathbf{I}_m$ is the image gradient at this pixel.

Geometric constraint: We impose geometric constraints to the state vector by applying the part-based human body detector as proposed in [191]. The detector is based on the histogram of gradients (HOG) feature. We further apply body size constraints to filter out potential background human subjects, and then pick the detection result with highest confidence value among the remaining ones. A reliable head region R_j^i with respect to the detected human body area is then selected. Note this cue forms a hard constraint for the state vector:

$$\psi(O_t, \mathbf{s}_t^i) = \begin{cases} 0 & \text{if } E_j^i \subset R_j^i = \emptyset \\ 1 & \text{otherwise} \end{cases} \tag{5.16}$$

The overall image likelihood can be calculated as:

$$P(O_t|\mathbf{s}_t^i) \propto \ln \psi(O_t, \mathbf{s}_t^i) + \lambda_1 \ln \phi(O_t, \mathbf{s}_t^i) + \lambda_2 \ln \varphi(O_t, \mathbf{s}_t^i), \tag{5.17}$$

where λ_1 and λ_2 are determined by applying a logistic regression-like algorithm to independent data. We determine the location of the head in 3D space as:

$$\begin{aligned} \mathbf{s}_t &= \underset{\mathbf{s}_t^i}{\text{argmax}} P(\mathbf{s}_t^i | O_t) \\ &= \underset{\mathbf{s}_t^i}{\text{argmax}} P(O_t | \mathbf{s}_t^i) P(\mathbf{s}_t^i | \mathbf{s}_{t-1}^i) \end{aligned} \tag{5.18}$$

Fig. 5.5 shows the result of our multi-view tracking algorithm. The tracker is able to track all the 500 frames without failure. Note that the video contains significant head motions in terms of rotation, translation and scaling. It is also subject to interruptions when the head moves out of the field of view. The second video example shown in Fig. 5.6 was captured when Baratunde Thurston, a technology-loving humorist and host of the Science Channel, visited the Biomotion laboratory at University of Maryland. Our multi-view tracking algorithm accurately locates the subject's head in spite of his dramatic motion. (Both videos are provided as supplementary materials.) Though in real-world surveillance videos subjects usually do not perform such extreme motions as in the example videos, the results clearly illustrate the robustness of our algorithm. The tracker also successfully handles all the videos in our database.

5.4.2 Texture Mapping

Once the MAP estimate of the head center is obtained, we are ready to obtain the surface texture map for the model. First, we uniformly sample the sphere's surface according to the following procedure:

1. Uniformly sample within the range $[-R,\ R]$, where R is the radius of the

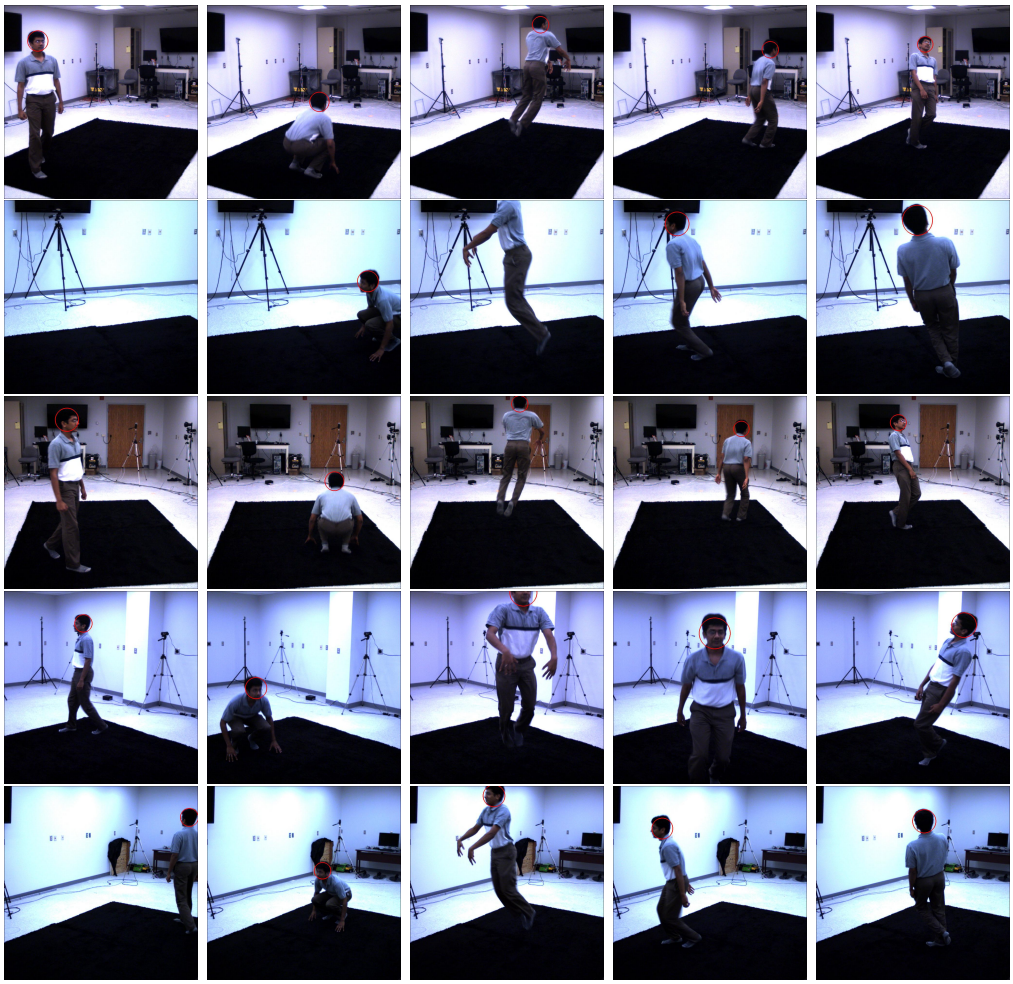


Figure 5.5: **Sample Tracking Results** Tracking results for a 500-frame multi-view video sequence. 5 views are shown here. Each row of images is captured by the same camera. Each column of images is captured at the same time.

Figure 5.6: **Sample Tracking Results** Tracking results for a 200-frame multi-view video sequence. The subject performs dramatic dancing motions. Five views are shown here. Each row of images is captured by the same camera. Each column of images is captured at the same time.

sphere, to get z_n, $n = 1, 2, ..., N$.

2. Uniformly sample α_n within the range $[0, 2\pi]$, and independent of z_n.

3. $x_n = \sqrt{R^2 - z_n^2}\cos\alpha_n$, $y_n = \sqrt{R^2 - z_n^2}\sin\alpha_n$.

Then, we perform a coordinate transformation for these sample points. Assume that their original world coordinates are $\{(x_n, y_n, z_n), n = 1, 2, ..., N\}$. After the transformation, we obtain $\{(x'_{n,j}, y'_{n,j}, z'_{n,j})\}$, which are their coordinates in the jth camera coordinate reference frame. We determine their visibility to camera j by examining $(x'_{n,j}, y'_{n,j}, z'_{n,j})$. Only an un-occluded point, i.e. which satisfies $z'_{n,j} \leq z'_{0,j}$, can contribute to an image on the jth camera's image plane. Here, $z'_{0,j}$ is the distance from the head center to the jth camera center. It is said that a back-projection link is created between a sample point on the model's surface and a pixel in a frame I_j if the former's world coordinates (x_n, y_n, z_n) and the latter's image coordinates $(x''_{n,j}, y''_{n,j})$ can be related under the weak-perspective projection assumption.

We denote the texture map for the jth camera view obtained by using such a back-projection approach as T^j. Note that when we iterate the procedure over all the cameras in the network, some model points will correspond to pixels from multiple views, because these cameras have overlapped field of views. For sample points in the overlapped region, we adopted a weighted fusion strategy, i.e., we assign weight $w_{n,j}$ to a pixel with image coordinate $\mathbf{p}_{n,j}$:

$$w_{n,j} = \exp(-\|\mathbf{p}_{n,j} - \mathbf{p}_{0,j}\|/r_j^i), \tag{5.19}$$

where $\mathbf{p}_{0,j}$ is the image coordinates of the pixel back-projected by the head model, and thus roughly the center of all the projections for camera j. Intuitively, the closer a pixel is to this center, the larger its contribution to the texture map should be. On the rim of a sphere a large number of sample points tend to project to the same pixel, and hence are not suitable for back-projection. The texture of the model point with world coordinates (x_n, y_n, z_n) is determined by:

$$T(x_n, y_n, z_n) = T^{j_0}(x_n, y_n, z_n), \tag{5.20}$$

where

$$j_0 = \arg\max_j \ w_{n,j}, \ \ j = 1, 2, ..., K. \tag{5.21}$$

This weighting scheme is illustrated in Fig. 5.7. Note that in our multi-view face recognition algorithm, T is in fact the function $f(\theta, \phi)$ that is subject to decomposition, as described in Section 5.3.

5.5 Video-Based Recognition

Video-based face recognition has some advantages. First, video offers data redundancy, which can be exploited to improve the robustness of a recognition algorithm. It has been reported in the literature that video-based algorithms in general achieve better performance than image-based ones. Second, by performing video tracking we can automate feature acquisition. Although it is always possible to extend the frame-based recognition result to a video-based one via simple fusion rules such as majority voting, a principled approach that exploits data's underlying

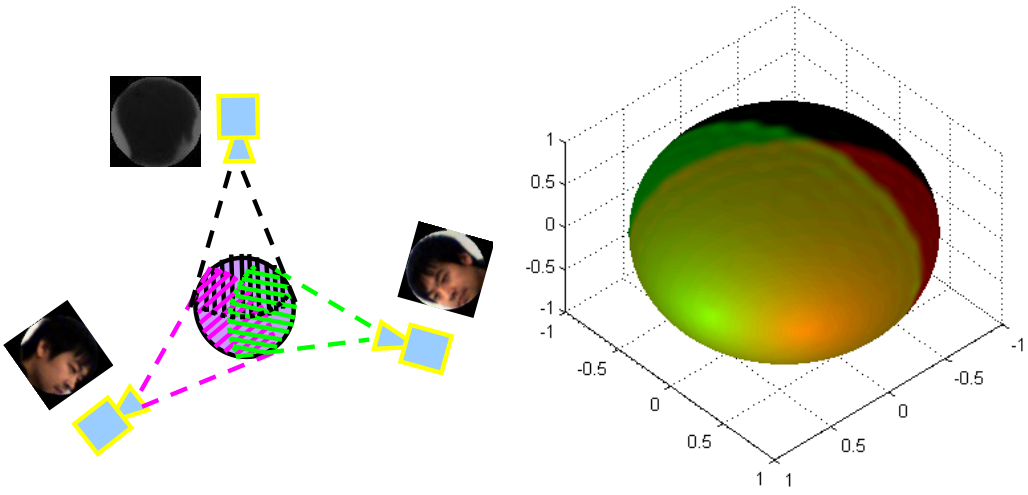


Figure 5.7: **Weighted Texture Mapping.** In multi-view texture mapping, the field of views of different cameras in a network often have overlap. The red (green) region on the sphere model represents the targeting back-projection area for the first (second) camera. The redness (greenness) at a certain point is proportional to its texture mapping weight with regard to the first (second) camera. In their overlapping region, whether a point is more red or more green determines which camera's image the texture map at that point should be based on.

structure is often more desirable for performance reason.

Given two multi-view video sequences with m and n (note that in general $m \neq n$) multi-view frames (a multi-view frame refers to the group of K frames synchronously captured by K cameras), respectively, two sets of feature vectors can be extracted. We look into their projections in the reproducing kernel Hilbert space (RKHS). The projection is indirectly performed via an Radial Basis Function (RBF) kernel. It is known that the kernel trick induces nonlinear feature mapping, which often leads to easier separation in RKHS. We treat each instance of feature vector as a sample from its class-conditional probability distribution. Therefore, the similarity of the two ensembles of features can be measured using the distance between the two class-conditional probability distributions in RKHS. By assuming that these distributions are Gaussian, analytical form of several different distance measures are derived in [48]. We follow [48] to calculate the limiting Bhattacharyya distance. To this end, the rank-deficient covariance matrix (since the dimensionality of RKHS is much higher than the number of data samples) involved in calculating the Bhattacharyya distance is replaced by an invertible approximation $\mathbf{C}$, which preserves the dominant eigenvalues and eigenvectors. The limiting Bhattacharyya distance in this case is:

$$D = \frac{1}{8}(\alpha_{11} + \alpha_{22} - 2\alpha_{12}), \tag{5.22}$$

where

$$\alpha_{ij} = \mu_i^T \left(\frac{1}{2}\mathbf{C}_i + \frac{1}{2}\mathbf{C}_j\right)^{-1} \mu_j^T. \tag{5.23}$$

We now show the steps to calculate (5.23) from the Gram matrix. Denote the Gram matrix as $\mathbf{K}_{ij}$, where $i, j \in \{1, 2\}$ are the indices of ensembles. The $\mathbf{K}_{11}$ and $\mathbf{K}_{22}$ are then centered:

$$\mathbf{K}'_{ii} = \mathbf{J}_i^T \mathbf{K}_{ii} \mathbf{J}_i, \ \mathbf{J}_i = N_i^{-1/2}(I_N - \mathbf{s}\mathbf{1}^T) \tag{5.24}$$

where $\mathbf{s} = N_i^{-1}\mathbf{1}$, $\mathbf{1}$ is a $N_i \times 1$ vector of 1s and N_i is the number of vectors in ensemble i. Let $\mathbf{V}_i$ be the matrix which stores the first r eigenvectors of $\mathbf{K}'_{ii}$ (i.e. corresponding to the r largest eigenvalues). Define:

$$\mathbf{P} = \begin{pmatrix} \sqrt{\frac{1}{2}}\mathbf{J}_1\mathbf{V}_1 & 0 \\ 0 & \sqrt{\frac{1}{2}}\mathbf{J}_2\mathbf{V}_2 \end{pmatrix}, \tag{5.25}$$

then it can be verified that

$$\left(\frac{1}{2}\mathbf{C}_i + \frac{1}{2}\mathbf{C}_j\right)^{-1} = \mathbf{I}_f - \begin{pmatrix} \mathbf{\Phi}_1 & \mathbf{\Phi}_2 \end{pmatrix} \mathbf{B} \begin{pmatrix} \mathbf{\Phi}_1^T \\ \mathbf{\Phi}_2^T \end{pmatrix}. \tag{5.26}$$

$\mathbf{I}_f$ is $f \times f$ identity matrix, where f is the dimensionality of the RKHS. And $\mathbf{\Phi}$ is the matrix of nonlinearly-mapped data in RKHS, which is not explicitly available to us. Matrix $\mathbf{B}$ can be computed from the Gram matrix:

$$\mathbf{B} = \mathbf{P}\mathbf{L}^{-1}\mathbf{P}^T, \ \mathbf{L} = \mathbf{P}^T \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix} \mathbf{P}. \tag{5.27}$$

By combining (5.23) and (5.26), we have:

$$\alpha_{ij} = \mathbf{s}_i^T \mathbf{K}_{ij} \mathbf{s}_j - \mathbf{s}_i^T \begin{pmatrix} \mathbf{K}_{i1} & \mathbf{K}_{i2} \end{pmatrix} \mathbf{B} \begin{pmatrix} \mathbf{K}_{1j} \\ \mathbf{K}_{2j} \end{pmatrix} \mathbf{s}_j. \tag{5.28}$$

5.6 Experiments

5.6.1 Database

As mentioned in Section 5.2, there are very few works addressing the multi-view face recognition problem. We exhaustively searched for a public multi-view video-based face database. It seems that a database which contains videos captured by multiple **synchronized** cameras is not available yet. Therefore, we collected a multi-view video database using an indoor camera network. The database has 40 subjects. The videos were collected at four different sessions and are 100 to 200 frames in length. Most of the subjects have 3 videos and some of them have 2 or 4 videos. We use one as gallery and the rest as probes. This database is double the size of its previous versions [192][193] in terms of the number of videos. To test the robustness of our recognition system, we have arranged the time span that separated the sessions to be up to 6 months. The appearance of many subjects has changed significantly between the sessions. Such a dataset well serves our purpose of simulating a practical surveillance environment and poses great challenges to multi-view face recognition algorithms. Fig. 5.8 shows some example frames from gallery and probe video sequences.

5.6.2 Feature Comparison

As the proposed feature can work for a single multi-view frame as well as video sequences, we first associate four different kinds of features with different

Figure 5.8: **Example of Gallery and Probe Video Frames.** Shown in the first row are examples of gallery frames and the second row are examples of probe frames.

classifiers to compare their performance in image-based face recognition settings. By "image-based face recognition" we mean that each frame (a multi-view frame for the SH spectrum feature and a single-view frame for other features.) is treated as a gallery or probe individually without concerning which video it comes from. We use one multi-view video of each subject as the gallery and the remaining videos as probe. We pick every 10th frame in this experiment. The four features are: Locality Preserving Projection (LPP) [194] and LDA in the original image space, SH raw coefficients with PCA, and the proposed SH spectrum feature. For the first two features, we use the face image that is automatically cropped by a circular mask as a result of tracking, and normalize it to the size 50×50. For LDA, we first train a PCA projection matrix from all the gallery images to reduce the dimension of the original image feature, in order to avoid the intra-class scatter matrix's rank deficiency issue. As in the conventional LDA formulation, the criterion we choose to optimize is $\det(WS_bW^T)/\det(WS_wW^T)$, where W is the projection matrix, and S_b and S_w are the between-class/within-class scatter matrices, respectively. For LPP, we utilize label information in the gallery by setting the weights between inter-class samples to be 0. We also use cross-validation to determine the optimal scale constant which is defined in the weight matrix of LPP. The experiment runs in a single-view vs. single-view mode for the LPP and LDA case, and in a multi-view vs. multi-view mode for the SH+PCA and SH spectrum feature case. The results are shown in Table 5.1. Due to the incompatibility of the nature of single-view features with the special structure of multi-view image data, the performance of the proposed feature exceeds them by a large margin in all cases.

Table 5.1: Comparison of Recognition Performance

Feature	NN	KDE	SVM-Linear	SVM-RBF
LPP	56.1%	42.7%	58.8%	65.9%
LDA	51.3%	34.8%	40.6%	47.4%
SH PCA	40.7%	36.4%	39.3%	52.2%
Proposed	**65.3%**	**65.1%**	**79.0%**	**87.3%**

To quantitatively verify the proposed feature's discriminant power, we then conducted the following experiment. We calculate distances for each unordered pair of feature vectors $\{\mathbf{x}_i, \mathbf{x}_j\}$ in the gallery. If $\{\mathbf{x}_i, \mathbf{x}_j\}$ belongs to the same subject, then the distance is categorized as being *in-class*. Otherwise, the distance is categorized as being *between-class*. We approximate the distribution of the two kinds of distances as histograms. Intuitively, if a feature has good discrimination power, then the in-class distances evaluated using that feature tends to be smaller compared to the between-class distances, and hence the distributions of the two distances should exhibit large divergence. We use the symmetric KL divergence ($KL(p\|q) + KL(q\|p)$) to evaluate the difference between the two distributions. We summarize the results for the four features in Table 5.2 and plot three of them in Fig. 5.9. The in-class distances for the SH spectrum feature are concentrated in the low value bins, while its between-class distance tends to have higher values. Their modes are obviously separated. For the other features, the between-class distance tend to mix with the in-class distance.

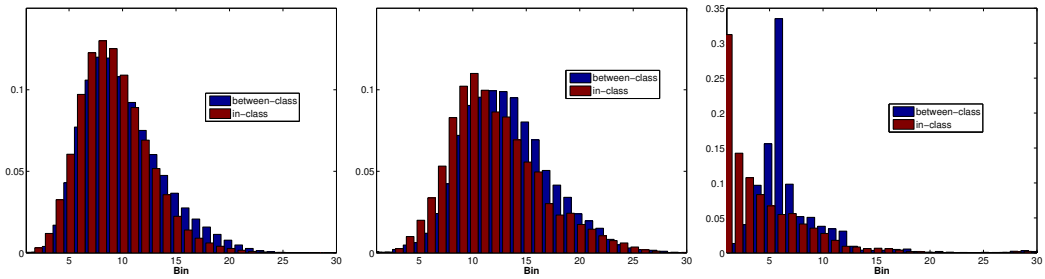


Figure 5.9: **Comparison of the Discriminant Power** Histograms of between-class distance distribution (blue) and in-class distance distribution (red) of the LDA feature (left), LPP feature (center) and the SH spectrum feature (right) are presented above. Number of bins is 30.

Table 5.2: KL divergence of in-class and between-class distances for different features

LPP	LDA	SH+PCA	SH Spectrum
0.3511	0.2709	0.2866	**1.3141**

The symmetric KL-divergence also suggests the same phenomenon.

5.6.3 Video-Based Recognition

The algorithm we use for video-level recognition is the one as described in Section 5.5. We compare the performance of our video recognition algorithm with five other ones: (1) Ensemble-similarity-based algorithm directly applied to the raw image. Inputs are the head images which are tracked in a video and scaled to size 50 by 50. The kernel is RBF. (2) View-selection-based algorithm. We use a Viola-Jones frontal face detector [102] to select frontal-view face images from both gallery and probe multi-view videos. The chosen frames from a subject's gallery

video are then used to construct the personal frontal-view face PCA subspace. The frontal-view frames from the probe videos are fitted to the personal PCA subspaces for recognition. Video-level decision is made through majority voting. (3) The probabilistic appearance manifold algorithm proposed in [32]. We use 8 planes for the local manifold model and set the probability of remaining the same pose to be 0.7 in the pose transition probability matrix. We first use this algorithm to process each camera view of a probe video. To fuse results of different camera views we use majority voting. If there is a tie in views' voting, we pick the one with smaller Hausdorff distance. (4) Image-based recognition with SH spectrum feature and majority voting for video-level fusion. We use SVM with RBF kernel for every multi-view frame recognition. Note however that the recognition accuracies in this case should not be compared to the previous experiment's result to draw misleading conclusions[2]. (5) The Manifold-Manifold Distance (MMD) algorithm presented in [39]. We use the author's code and parameter settings. When comparing two multi-view videos, we first calculate the MMD between the sequence pairs of the same view, and then use the minimum MMD across views as the distance measure. We also tried with average MMD across views, which yielded similar results.

[2]The numbers in the two cases are not convertible to each other, as in the previous image-based recognition experiment we did not fuse results with respect to video. Think of two extreme situations: (1) For each video of the probe set, 51% frames are individually correctly recognized. (2) For half of the probe videos, 100% frames are individually correctly recognized and for the remaining half only 49% frames are correctly recognized. The overall image recognition rate and majority-voting-based video recognition rate are respectively 51% and 100% in the former case, and 74.5% and 50% in the latter one.

We plot the cumulative recognition rate curve in Fig. 5.10. The view-selection method heavily relies on the availability of frontal-view face images, however, in the camera network case, the frontal pose may not appear in any view of the cameras. As a result, it does not perform well in this experiment. The manifold-based algorithm, the MMD-based algorithm and the image-ensemble-based algorithm use more principled strategies than voting to combine classification results of individual frames. Moreover, they both have certain ability to handle pose variations, especially the two algorithms based on manifold. However, because they are designed to work with a single camera, they are single-view in nature. Repeating these algorithms for each view does not fully utilize the multi-view information. For example, we found in our experiments that mismatches made by the MMD algorithm often happens when the minimum MMD is produced between the back-of-head clusters, which have similar appearance representations even for different subjects. In contrast, the proposed method based on a robust feature performs noticeably better in this experiment. An additional advantage of the algorithm is that it requires no pose estimation or model registration. Comparison between the ensemble matching algorithm and the majority voting method which both use the proposed feature demonstrates the superiority of a systematic fusion strategy to an ad-hoc one.

5.7 Conclusion

In this chapter, we proposed a multi-view face recognition algorithm. The most noteworthy feature of the algorithm is that it does not require any pose es-

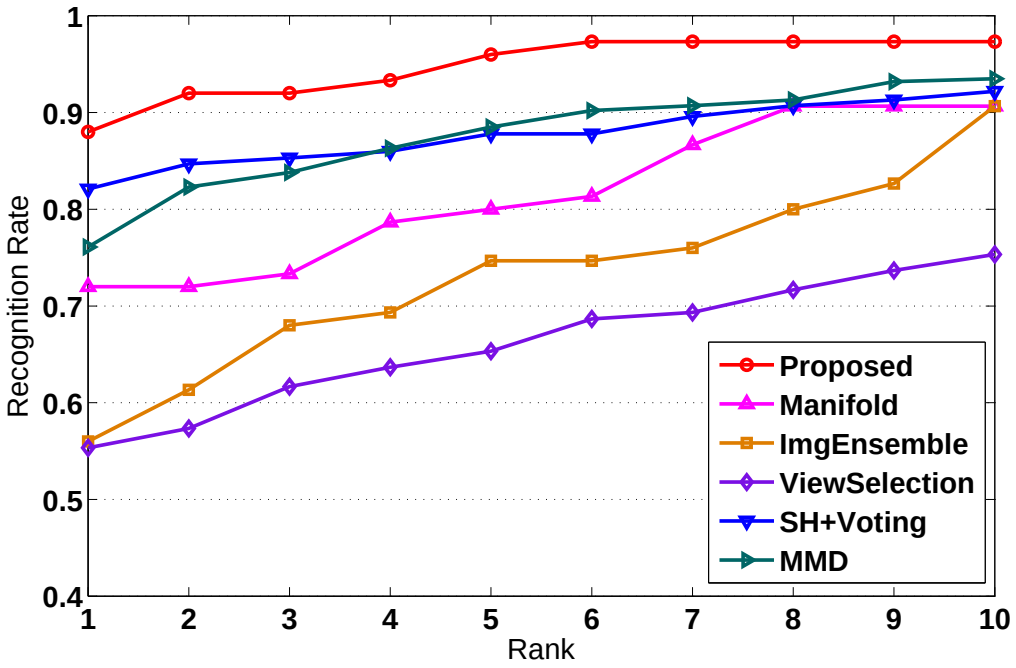


Figure 5.10: **Video Face Recognition Results** Cumulative recognition rate of the video-based face recognition algorithms.

timation or model registration step. Under the normal diffuse lighting condition, we present a robust feature by exploring the fact that the subspace spanned by Spherical Harmonics is an irreducible representations for the SO(3) group. We also proposed a multi-view video tracking algorithm to automate the feature acquisition in a camera network setting. We modeled the video-based recognition problem as one of measuring ensemble similarities in RKHS. We demonstrated the performance of our method on a relatively uncontrolled multi-view video database.

Limitations One limitation of our method is that the pose insensitivity property of the SH representation relies on the assumption that the spherical function remains unchanged other than a rotation, i.e.: $f(\theta, \phi, t_1) = f(R(\theta, \phi), t_2)$. In practice, this could always be affected by real-world lighting conditions. Under normal

lighting conditions, this assumption is reasonable, and as we mentioned, even global illumination variation can be partially compensated for by the energy normalization step in feature extraction. However, extreme lighting conditions can render the assumption invalid. This could happen when, for example, there are anisotropic illumination variations, or strong directional light is casted onto the face from the side. Such situations will result in large fluctuation in the features and cause the recognition performance to degrade. There are some possible solutions to this problem. For example, we could use the self-quotient method to preprocess video frames, or we could figure out a way to integrate the algorithm in [184] for a uniform modeling of both lighting conditions and face appearance. This will be one of our future research directions. Our algorithm also relies on the assumption that human head can be approximated by a sphere. While this approximation may be reasonable, model fitting errors due to the non-spherical nature of human heads do exist and can become evident in certain cases. Moreover, because we treat the texture map as a spherical function, unavoidably there will be quantization error caused by the discrete pixel value. Finally, calibration of camera network could be a source of error, too.

Chapter 6

Future Works

In this dissertation, we studied the video-based face recognition problem in three different aspects, i.e. face extraction from videos, recognition using single-view videos and recognition using multi-view videos. Our research has shown that by leveraging the information contained in videos, we can achieve encouraging results in all of these problems in spite of the challenging data used in our experiments. However, we realized that limitations exist and robust face recognition from completely unconstrained videos is still an open problem. We have also noticed that pose variations, especially the out-of-plane ones, remains the main issue to be addressed by a VFR algorithm. In the following, we would like to outline several possible directions to pursue in the future.

6.1 Deep Learning

Deep learning [195] has triggered a revolution in the computer vision community by reshaping the way that objects are represented. Traditionally, most of the successful visual features used in object detection or recognition, such as SIFT [196], Histogram of Gradients [197], Histogram of Gabor Phase Pattern (HGPP) [198] etc., almost all rely on hand-crafting to capture the essence of different visual patterns. Deep learning techniques allow us to learn the features from data, and hence can be

tied directly to our final goal. The application of deep learning to face processing has led to promising results. In [199], Osardchy et al. developed a real-time face detection algorithm based on convolutional neural networks. Recently, the DeepFace system proposed by Taigman et al. [200] has achieved near-human performance on the challenging LFW database. By applying the learned representation of face to the association or recognition framework presented in this dissertation, we expect to obtain substantial performance boost.

6.2 Cross-Scene Face Association

The face association framework presented in this dissertation is concerned with video frames of the same scene. In many applications, it is often desirable to further group the faces across scenes, forming larger clusters. As a result, there will be more face images per subject at our disposal, which is potentially advantageous for the following name labeling or recognition task. For cross-scene clustering we will need to define a proper distance measure and we are interested in applying metric learning techniques [155]. In terms of clustering, one can adopt the Dirichlet process mixture model used in Chapter 4 to extract representative frames.

6.3 Adaptive Face Association

The parameters of our CRF and M^3 networks remain fixed after having been learned from the training data. However, one set of parameters is not necessarily suitable for processing all videos. We would like to infer the parameters at run-time,

adapting to the test videos. To this end, we could take an iterative optimization strategy, which is similar to the generalized-EM algorithm. Specifically, one could maximize the lower bound of the data log likelihood:

$$Q(\mathbf{w}, q) = -KL(q(\mathbf{y})||p(\mathbf{y}|\mathbf{x}, \mathbf{w})) + \ln Z(\mathbf{x}, \mathbf{w}) \tag{6.1}$$

by alternately fixing the parameters $\mathbf{w}$ and the labels $\mathbf{y}$. The $q(\mathbf{y})$ in the equation is an approximation to the posterior distribution and is in the form of a product of marginal distributions. Scene descriptive features may be used as priors for the parameters to improve convergence rate.

6.4 Joint Framework for Face Alignment and Video-Based Face Recognition

In Chapter 4, the face alignment problem was handled by a structural SVM and recognition was based on intra-personal/extra-personal dictionary learning. These are two decoupled optimization processes. Being aware of the existing work on batch alignment of face images by sparse and low-rank decomposition [201], we are curious to ask the question whether it is possible to integrate alignment and recognition into a unified framework. We anticipate the joint method to work more effectively for each individual task than currently independently optimized schemes.

6.5 Still-To-Video Face Recognition Problem

Though video-based face recognition has received more and more interest, not many of them are devoted to the still-to-video case. The approach proposed in [15] is one of the only few algorithms that systematically address this issue in a probabilistic framework. However, the dense sampling strategy adopted in the algorithm has serious limitations in its applicability to practical databases. We are interested in developing an efficient scheme to intelligently sample the joint state space of pose and identity parameters. To be specific, we would like to transform the space of identity variable to a well-behaved one such that it possess certain "continuity". We expect the manifold-based algorithms and the metric learning methods to play an important role in defining such a transform. Joint tracking and recognition approaches based on such a transformed state space could potentially run very efficiently and be applicable to practical scenarios.

6.6 Spherical Harmonics Based Head Pose Estimation

In Chapter 5, we have mentioned that after the pose of a head changes, the SH coefficients of the texture map at a certain degree will be linearly related to the original ones. This property can be utilized to estimate the head pose. As the pose parameters are coupled in a nonlinear fashion in the transformation matrix, stochastic optimization algorithms are required to find the solution. We have verified the idea on toy problems. However, to apply it to a head pose estimation

problem, there are many practical issues to be addressed. We also plan to extend our work to more complicated conditions, such as outdoor environments, less stringent calibration requirements etc.

Appendix A

Structural SVM

As an extensively adopted supervised classifier, the Support Vector Machine (SVM) is well known for its theoretically guaranteed generalization error bound and its easy integration with kernels. While the traditional SVM is appropriate for single-output classification tasks, the Structural SVM (SSVM) [160] generalizes the max margin principle to the vector output case. It is also know as max margin Markov networks (or M^3nets), as we can also view it as a result of applying the max margin learning rule to replace the maximum likelihood one in a Hidden Markov Model. Relationships between some common, single-output and structural-output classifiers are listed in Table (A.1):

Learning Rule	**Single Output**	**Structural Output**
Maximum Joint Likelihood	Naive Bayesian Classifier	Hidden Markov Model
Maximum Conditional Likelihood	Logistic Regression	Conditional Random Field
Maximum Margin	SVM	Structural SVM (M^3nets)

Table A.1: Relationships between single-output and structural-output classifiers

A.1 Problem Formulation

Suppose we have a feature function $\phi(\mathbf{x}, \mathbf{y})$ which measures the fitness of input $\mathbf{x}$ with structural output $\mathbf{y}$. As in the case of SVM and other generalized linear models, we calculate the linear filter response $f(\mathbf{x}, \mathbf{y}) = \mathbf{w}^T \phi(\mathbf{x}, \mathbf{y})$ and take $\mathbf{y}^* = \underset{\mathbf{y} \in \mathcal{Y}}{\operatorname{argmax}} f(\mathbf{x}, \mathbf{y})$ as the prediction output. Ideally, we hope we could find filter coefficients $\mathbf{w}$ such that the following condition holds:

$$\mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i^*) \geq \max_{\mathbf{y} \in \mathcal{Y} \backslash \mathbf{y}_i^*} \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}) \ \forall i \tag{A.1}$$

. Among all the solutions satisfying the condition, we are particularly interested in the one that maximizes the margin, which is defined as:

$$\eta = \min_i [\mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \max_{\mathbf{y} \in \mathcal{Y} \backslash \mathbf{y}_i} \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y})] \tag{A.2}$$

, i.e. the smallest difference between the filter response of the ground truth and that of the second optimal solution, across all training samples. However, maximizing (A.2) would yield unbounded solution unless we enforce constraint on the scale of $\mathbf{w}$. For convenience, we let $\|\mathbf{w}\| = 1$ and try to solve the optimization problem:

$$\max_{\mathbf{w}: \|\mathbf{w}\|=1} \eta \ s.t. \ \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}) \geq \eta \ \forall i, \ \forall \mathbf{y} \in \mathcal{Y} \backslash \mathbf{y}_i \tag{A.3}$$

, or equivalently:

$$\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 \ s.t. \ \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}) \geq 1 \ \forall i, \ \forall \mathbf{y} \in \mathcal{Y} \backslash \mathbf{y}_i \tag{A.4}$$

Just as in the SVM case, we can define slack variables to tolerate some errors for the training samples. The optimization problem is then converted to:

$$\min_{\mathbf{w}, \xi_i \geq 0} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i \ s.t. \ \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}) \geq 1 - \xi_i \ \forall i, \ \forall \mathbf{y} \in \mathcal{Y} \backslash \mathbf{y}_i \tag{A.5}$$

In the SVM case, since the output is a scalar representing class label, usually all errors are treated equally. However, when we are concerned with structural output, it is intuitive to define a loss function $\Delta(\mathbf{y}, \mathbf{y}_i)$ to penalize constraint violations differently. Accordingly, stringent slack variables will be assigned to training samples that cause more serious violation of constraints. There are two different schemes to achieve this. The first is slack-rescaling SSVM:

$$\min_{\mathbf{w}, \xi_i \geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i \ s.t.\ \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}) \geq 1 - \frac{\xi_i}{\Delta(\mathbf{y}, \mathbf{y}_i)}\ \forall i,\ \forall \mathbf{y} \in \mathcal{Y}\backslash\mathbf{y}_i \tag{A.6}$$

. The second is margin-rescaling SSVM:

$$\min_{\mathbf{w}, \xi_i \geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i \ s.t.\ \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}) \geq \Delta(\mathbf{y}, \mathbf{y}_i) - \xi_i\ \forall i,\ \forall \mathbf{y} \in \mathcal{Y}\backslash\mathbf{y}_i \tag{A.7}$$

. Note that in the optimization problems above, we have $|\mathcal{Y}| - 1$ constraints for each training sample. Therefore the total number of constraints is $N|\mathcal{Y}| - N$, determined by both the size of the training database and the cardinality of configuration space. Equivalently, we can rewrite the formulation using $|\mathcal{Y}| - 1$ constraints:

$$\min_{\mathbf{w}, \xi_i \geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i \ s.t.\ \max_{\mathbf{y} \in \mathcal{Y}}[\Delta(\mathbf{y}, \mathbf{y}_i) + \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y})] - \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}_i) \leq \xi_i\ \forall i,\ \forall \mathbf{y} \in \mathcal{Y}\backslash\mathbf{y}_i \tag{A.8}$$

. Finally, we obtain the following optimization problem:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \max_{\mathbf{y} \in \mathcal{Y}}[\Delta(\mathbf{y}, \mathbf{y}_i) + \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y})] - \mathbf{w}^T\phi(\mathbf{x}_i, \mathbf{y}_i) \tag{A.9}$$

A.2 Learning SSVM

A.2.1 Subgradient Method

Subgradient of a convex function $f : \mathbf{R}^n \rightarrow \mathbf{R}$ at a point $\mathbf{w}_0$ is defined as a vector $\mathbf{v}$ such that:

$$f(\mathbf{w}) - f(\mathbf{w}_0) \geq \mathbf{v} \cdot (\mathbf{w} - \mathbf{w}_0) \tag{A.10}$$

. Obviously, subgradients coincide with traditional gradients at any differentiable points of the function, but become a set at those non-differentiable ones. The subgradient method is an iterative procedure to optimize a convex objective function, in which we move along a negative subgradient direction at each iteration:

$$\mathbf{w}_k = \mathbf{w}_{k-1} - \alpha_k \mathbf{w}_k \tag{A.11}$$

, where α_k is a learning rate following certain shrinking scheme. We summarize the subgradient algorithm for SSVM training in Algorithm 3.

The subgradient algorithm has a slow convergence rate of $O(\sqrt{\epsilon})$, which means that it requires $O(1/\epsilon^2)$ iterations to reduce the distance to the optimal solution by a factor of ϵ [202]. However, the bundle version of the algorithm can achieve faster convergence.

A.2.2 Cutting Plane Algorithm

Cutting plane method [203] is a delayed constraint generation technique. The procedure is initialized with an empty set of active constraints. Then at each iteration, it solves the quadratic programming problem (A.8) with constraints from the

Algorithm 3: The subgradient Structural SVM training algorithm.

input : N labeled training samples$\{\mathbf{x}_n, \mathbf{y}_n\}$

number of iterations K

learning rate α

output: Optimal parameters $\mathbf{w}^*$

Initialization: $\mathbf{w}_0 = 0$

for $k = 1 \rightarrow K$ **do**

  for $n = 1 \rightarrow N$ **do**

    $\mathbf{y}'_n = \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}} \Delta(\mathbf{y}, \mathbf{y}_n) + \mathbf{w}_{k-1}^T \phi(\mathbf{x}_n, \mathbf{y})$

  $g = \mathbf{w}_{k-1} + C \sum_{n=1}^{N} [\phi(\mathbf{x}_n, \mathbf{y}_n) - \phi(\mathbf{x}_n, \mathbf{y}'_n)]$

  $\mathbf{w}_k = \mathbf{w}_{k-1} - \frac{\alpha}{K} g$

$\mathbf{w}^* = \mathbf{w}_K$

current set of active constraints. Using the solved parameters and slack variables, it then searches for the most violated constraints and adds it to the working set. The iteration continues until the set of active constraints no longer changes. The method is summarized in Algorithm 4.

Algorithm 4: The cutting plane Structural SVM training algorithm.

input : N labeled training samples$\{\mathbf{x}_n, \mathbf{y}_n\}$

Preset threshold ϵ

output: Optimal parameters $\mathbf{w}^*$

Initialization: $S = \emptyset$

repeat

Obtain $(\mathbf{w}, \xi)$ by solving the constrained quadratic programming problem

(A.8), where constraints are from S

for $n = 1 \rightarrow N$ **do**

$\mathbf{y}'_n = \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}} \Delta(\mathbf{y}, \mathbf{y}_n) + \mathbf{w}^T \phi(\mathbf{x}_n, \mathbf{y})$

if $\Delta(\mathbf{y}'_n, \mathbf{y}_n) + \mathbf{w}^T \phi(\mathbf{x}_n, \mathbf{y}'_n) - \mathbf{w}^T \phi(\mathbf{x}_n, \mathbf{y}_n) - \xi_n > \epsilon$ **then**

$S = S \cup \{(\mathbf{x}_n, \mathbf{y}'_n)\}$

until S *does not change*;

$\mathbf{w}^* = \mathbf{w}$

Bibliography

[1] W. Y. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face recognition: A literature survey," *ACM Computing Survey*, vol. 35, pp. 399–458, 2003.

[2] B. Moghaddam, "Principal manifolds and probabilistic subspaces for visual recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 780–788, 2002.

[3] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent k-svd: learning a discriminative dictionary for recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, November 2013.

[4] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," in *ICCV*, vol. 2, July 2001, pp. 383–390.

[5] H. Wang, Y. Wang, and Y. Cao, "Video-based face recognition: a survey," *World Academy of Science, Engineering and Technology*, pp. 293–302, 2009.

[6] J. Steffens, E. Elagin, and H. Neven, "Personspotter - fast and robust system for human detection, tracking and recognition," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 1998, pp. 516–521.

[7] S. A. Berrani and C. Garcia, "Enhancing face recognition from video sequences using robust statistics," in *Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance*, September 2005, pp. 324–329.

[8] X. Liu, T. Chen, and S. M. Thornton, "Eigenspace updating for non-stationary process and its application to face recognition," *Pattern Recognition*, vol. 36, pp. 1945–1959, 2003.

[9] Y. Zhang and A. M. Martnez, "A weighted probabilistic approach to face recognition from multiple images and video sequences," *Image and Vision Computing*, vol. 24, no. 6, pp. 626–638, 2006.

[10] U. Park, A. K. Jain, and A. Ross, "Face recognition in video: adaptive fusion of multiple matchers," in *Proceedings of IEEE Computer Society Workshop on Biometrics (in conjunction with CVPR)*, June 2007, pp. 1–8.

[11] J. Stallkamp, H. K. Ekenel, and R. Stiefelhagen, "Video-based face recognition on real-world data," in *IEEE International Conference on Computer Vision*, October 2007, pp. 1–8.

[12] B. Li and R. Chellappa, "Face verification through tracking facial features," *Journal of the Optical Society of America A*, vol. 18, no. 12, pp. 2969–2981, 2001.

[13] M. Isard and A. Blake, "Condensation /conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, pp. 5–28, August 1998.

[14] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002.

[15] S. Zhou, V. Krueger, and R. Chellappa, "Probabilistic recognition of human faces from video," *Computer Vision and Image Understanding*, vol. 91, pp. 214–245, 2003.

[16] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 696–710, 1997.

[17] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. on Image Processing*, 2004.

[18] S. Mckenna, S. Gong, and Y. Raja, "Face recognition in dynamic scenes," in *British Machine Vision Conference*, September 1997, pp. 140–151.

[19] L. Torres and J. Vila, "Automatic face recognition for video indexing applications," *Pattern Recognition*, vol. 35, no. 3, pp. 615–625, 2002.

[20] S. Satoh, "Comparative evaluation of face sequence matching for content-based video access," in *IEEE International Conference on Automatic Face and Gesture Recognition*, March 2000, pp. 163–168.

[21] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, December 1936.

[22] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition using temporal image sequence," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 1998, pp. 318–323.

[23] K. Fukui and O. Yamaguchi, "Face recognition using multi-viewpoint patterns for robot vision," in *International Symposium of Robotics Research*, Octobor 2003, pp. 192–201.

[24] M. Nishiyama, O. Yamaguchi, and K. Fukui, "Face recognition using the multiple constrained mutual subspace mehod," *IEICE Transactions on Information and Systems*, vol. 88, no. 8, pp. 1339–1348, 2005.

[25] L. Wolf and A. Shashua, "Kernel principal angles for classification machines with applications to image sequence interpretation," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2003, pp. 635–642.

[26] T. K. Kim, J. Kittler, and R. Cipolla, "Discriminative learning and recognition of image set classes using canonical correlations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1005–1018, 2007.

[27] H. Cevikalp and B. Triggs, "Face recognition based on image sets," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2567–2573.

[28] Y. Hu, A. S. Mian, and R. Owens, "Sparse approximated nearest points for image set classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 121–128.

[29] T. K. Kim, O. Arandjelovic, and R. Cipolla, "Boosted manifold principal angles for image set-based recognition," *Pattern Recognition*, vol. 40, no. 9, pp. 2475–2484, 2007.

[30] Y. Li, S. Gong, and H. Liddell, "Video-based online face recognition using identity surfaces," in *Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, July 2001, pp. 40–46.

[31] E. Kokiopoulou and P. Frossard, "Video face recognition with graph-based semi-supervised learning," in *ICME*, June/July 2009, pp. 1564–1565.

[32] K. C. Lee, J. Ho, M. H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2003, pp. 313–320.

[33] K.-C. Lee, J. Ho, M.-H. Yang, and D. J. Kriegman, "Visual tracking and recognition using probabilistic appearance manifolds," *Computer Vision and Image Understanding*, vol. 99, no. 3, pp. 303–331, 2005.

[34] O. Arandjelovic and R. Cipolla, "Face recognition from video using the generic shape-illumination manifold," in *European Conference on Computer Vision*, May 2006, pp. 27–40.

[35] ——, "A pose-wise linear illumination manifold model for face recognition using video," *Computer Vision and Image Understanding*, vol. 113, no. 1, pp. 113–125, 2009.

[36] W. Fan and D.-Y. Yeung, "Locally linear models on face appearance manifolds with application to dual-subspace based classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2006, pp. 1384–1390.

[37] ——, "Face recognition with image sets using hierarchically extracted exemplars from appearance manifolds," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 2006, pp. 1384–1390.

[38] W. Liu, Z. Li, and X. Tang, "Spatio-temporal embedding for statistical face recognition from video." in *European Conference on Computer Vision*, May 2006, pp. 374–388.

[39] R. Wang, S. Shan, X. Chen, and G. Wen, "Manifold-manifold distance with application to face recognition based on image set," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[40] Y. Zhao, S. Xu, and Y. Jia, "Discriminant clustering embedding for face recognition with image sets," in *Asian Conference on Computer Vision*, November 2007, pp. 641–650.

[41] R. Wang and X. Chen, "Manifold discriminant analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 429–436.

[42] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[43] Y.-C. Chen, V. M. Patel, and R. Chellappa, "Dictionary-based face recognition from video," in *European Conference on Computer Vision*, October 2012, pp. 766–779.

[44] Y.-C. Chen, V. Patel, S. Shekhar, R. Chellappa, and P. Phillips, "Video-based face recognition via joint sparse representation," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 2013, pp. 1–8.

[45] E. G. Ortiz, A. Wright, and M. Shah, "Face recognition in movie trailers via mean sequence sparse representation-based classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 3531–3538.

[46] G. Shakhnarovich, J. W. Fisher, and T. Darrell, "Face recognition from long-term observations," in *European Conference on Computer Vision*, May/June 2002, pp. 851–868.

[47] O. Arandjelovic and R. Cipolla, "Face recognition from face motion manifolds using robust kernel resistor-average distance," in *Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop*, vol. 5, June 2004, pp. 88–93.

[48] S. K. Zhou and R. Chellappa, "From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel Hilbert space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 917–929, 2006.

[49] X. Liu and T. Chen, "Video-based face recognition using adaptive hidden Markov models," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003.

[50] A. Hadid and M. Pietikainen, "From still image to video-based face recognition: an experimental analysis," in *IEEE International Conference on Automatic Face and Gesture Recognition*, May 2004, pp. 17–19.

[51] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[52] M. Tistarelli, M. Bicego, and E. Grosso, "A weighted probabilistic approach to face recognition from multiple images and video sequences," *Image and Vision Computing*, vol. 27, no. 3, pp. 222–232, 2009.

[53] G. Aggarwal, A. K. Roy-Chowdhury, and R. Chellappa, "A system identification approach for video-based face recognition," in *International Conference on Pattern Recognition*, August 2004, pp. 175–178.

[54] K. D. Cock and D. B. Moor, "Subspace angles and distances between arma models," in *Proceedings of International Symposium of Mathematical Theory of Networks and System*, August 2000.

[55] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999, pp. 187–194.

[56] ——, "Face recognition based on fitting a 3d morphable model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.

[57] A. K. Roy-Chowdhury and R. Chellappa, "Face reconstruction from monocular video using uncertainty analysis and a generic model," *Computer Vision and Image Understanding*, vol. 91, pp. 188–213, 2003.

[58] Z. Zhang, Z. Liu, D. Adler, M. F. Cohen, E. Hanson, and Y. Shan, "Robust and rapid generation of animated faces from video images: a model-based modeling approach," *International Journal of Computer Vision*, vol. 58, no. 2, pp. 93–119, 2004.

[59] P. Breuer, K.-I. Kim, W. Kienzle, B. Scholkopf, and V. Blanz, "Automatic 3d face reconstruction from single images or video," in *IEEE International Conference on Automatic Face and Gesture Recognition*, September 2008, pp. 1 –8.

[60] T. Heseltine, N. Pears, and J. Austin, "Three-dimensional face recognition using combinations of surface feature map subspace components," *Image Vision Computing*, vol. 26, pp. 382–396, March 2008.

[61] L. Yin and M. T. Yourst, "3d face recognition based on high-resolution 3d face modeling from frontal and profile views," in *Proceedings of the 2003 ACM*

SIGMM workshop on Biometrics methods and applications, November 2003, pp. 1–8.

[62] J. Huang, B. Heisele, and V. Blanz, "Component-based face recognition with 3d morphable models," in *International Conference, Audio- and Video- Based Biometrics Person Authentication*, June 2003, pp. 27–34.

[63] D. Jiang, Y. Hu, S. Yan, L. Zhang, H. Zhang, and W. Gao, "Efficient 3D reconstruction for face recognition," *Pattern Recognition*, vol. 38, pp. 787–798, June 2005.

[64] G. J. Edwards, T. C. J., and C. T. F., "Improving identification performance by integrating evidence from sequences," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 1999, pp. 486–491.

[65] X. Tang and Z. Li, "Video based face recognition using mutiple classifiers," in *IEEE International Conference on Automatic Face and Gesture Recognition*, May 2004, pp. 345–349.

[66] X. Wang and X. Tang, "Unified subspace analysis for face recognition," in *IEEE International Conference on Computer Vision*, vol. 1, October 2003, pp. 679–686.

[67] S. Chen, S. Mau, M. T. Harandi, C. Sanderson, B. Abbas, and B. C. Lovell, "Face recognition from still images to video sequences: a local-feature-based framework," *EURASIP*, vol. 2011, no. 1, pp. 1–14, 2011.

[68] N. Ye and T. Sim, "Towards general motion-based face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2598–2605.

[69] T. L. Berg, B. A. C., J. Edwards, M. Maire, R. White, Y.-W. Teh, E. L.-M., and D. A. Forsyth, "Names and faces in the news," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, June/July 2004, pp. 848–854.

[70] B. Raytchev and H. Murase, "Unsupervised recognition of multi-view face sequences based on pairwise clustering with attraction and repulsion," *Computer Vision and Image Understanding*, vol. 91, pp. 22–52, July 2003.

[71] O. Arandjelovic and A. Zisserman, "Automatic face recognition for film character retrieval in feature-length films," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 860–867.

[72] A. W. Fitzgibbon and A. Zisserman, "On affine invariant clustering and automatic cast listing in movies," in *European Conference on Computer Vision*, vol. 3, May/June 2002, pp. 304–320.

[73] ——, "Joint manifold distance: a new approach to appearance based clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2003, pp. 26–33.

[74] J. Sivic, M. Everingham, and A. Zisserman, "Person spotting: video shot retrieval for face sets," in *Proceedings of International Conference on Image and Video Retrieval*, July 2005, pp. 226–236.

[75] M. Everingham, J. Sivic, and A. Zisserman, ""hello! my name is... buffy" – automatic naming of characters in tv video," in *British Machine Vision Conference*, vol. 3, September 2006, pp. 899–908.

[76] J. Sivic, M. Everingham, and A. Zisserman, ""who are you?" – learning person specific classifiers from video," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 1145–1152.

[77] N. E. Apostoloff and A. Zisserman, "Who are you? – real-time person identification," in *British Machine Vision Conference*, September 2007, pp. 509–518.

[78] D. Ramanan, S. Baker, and S. Kakade, "Leveraging archival video for building face datasets," in *IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007, pp. 1–8.

[79] M. Tapaswi, M. Bauml, and R. Stiefelhagen, ""knock! knock! who is it?" probabilistic person identification in tv-series," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2658–2665.

[80] M. Bauml, M. Tapaswi, and R. Stiefelhagen, "Semi-supervised learning with constraints for person identification in multimedia data," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 3602–3609.

[81] D. Beymer and T. Poggio, "Face recognition from one example view," in *IEEE International Conference on Computer Vision*, June 1995, pp. 500–507.

[82] X. Chai, S. Shan, X. Chen, and W. Gao, "Locally linear regression for pose-invariant face recognition," *IEEE Trans. on Image Processing*, vol. 16, pp. 1716–1725, July 2007.

[83] H. S. Lee and D. Kim, "Generating frontal view face image for pose invariant face recognition," *Pattern Recognition Letters*, vol. 27, pp. 747–754, May 2006.

[84] V. Blanz, T. Grother, P. J. Phillips, and T. Vetter, "Face recognition based on frontal views generated from non-frontal images," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 454–461.

[85] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 1994, pp. 84–91.

[86] ——, "Multi-view face recognition by nonlinear tensor," in *International Conference on Pattern Recognition*, December 2008, pp. 1–4.

[87] T. Kanade and A. Yamada, "Multi-subregion based probabilistic approach towards pose-invariant face recognition," in *IEEE International Symposium on Computational Intelligence in Robotics Automation*, vol. 2, July 2003, pp. 954–959.

[88] S. Lucey and T. Chen, "Learning patch dependencies for improved pose mismatched face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 909–915.

[89] J. J. Yokono and T. Poggio, "A multiview face identification model with no geometric constraints," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 2006, pp. 493–498.

[90] A. B. Ashraf, S. Lucey, and T. Chen, "Learning patch correspondences for improved viewpoint invariant face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2008, pp. 1–8.

[91] S. J. D. Prince, J. H. Elder, J. Warrell, and F. M. Felisberti, "Tied factor analysis for face recognition across large pose differences," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 970–984, June 2008.

[92] Y. Li, S. Gong, J. Sherrah, and H. Liddell, "Support vector machine based multi-view face detection and recognition," *Image and Vision Computing*, vol. 22, pp. 413–427, 2004.

[93] I. Kotsia, N. Nikolaidis, and I. Pitas, "Frontal view recognition in multiview video sequences," in *International Conference on Multimedia and Expo*, June 2009, pp. 702 –705.

[94] A. Pnevmatikakis and L. Polymenakos, *Far-field, multi-camera, video-to-video face recognition.* InTech, 2007, pp. 468–486.

[95] M. Nishiyama, M. Yuasa, T. Shibata, T. Wakasugi, T. Kawahara, and O. Yamaguchi, "Recognizing faces of moving people by hierarchical image-set matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[96] 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*, vol. 76, pp. 183–204, February 2008.

[97] J. Yoder, H. Medeiros, J. Park, and A. C. Kak, "Cluster-based distributed face tracking in camera networks," *IEEE Trans. on Image Processing*, vol. 19, pp. 2551–2563, October 2010.

[98] X. Liu and T. Chen, "Pose-robust face recognition using geometry assisted probabilistic modeling," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 502–509.

[99] ——, "Face mosaicing for pose robust video-based recognition." in *Asian Conference on Computer Vision*, vol. 2, November 2007, pp. 662–671.

[100] T. Choudhury, B. Clarkson, T. Jebara, and A. Pentland, "Multimodal person recognition using unconstrained audio and video," in *Proceedings of International Conference on Audio- and Video-Based Person Authentication*, March 1999, pp. 176–181.

[101] X. Zhou and B. Bhanu, "Integrating face and gait for human recognition at a distance in video," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 5, pp. 1119–1137, October 2007.

[102] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, pp. 137–154, May 2004.

[103] P. Felzenszwalb, D. McAllester, and D. Ramaman, "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[104] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2879–2886.

[105] K. Mikolajczyk, R. Choudhury, and C. Schmid, "Face detection in a video sequence - a temporal approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, December 2001, pp. 96–101.

[106] H. Schneiderman and T. Kanade, "A statistical method for 3d object detection applied to faces and cars," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2000, pp. 746–751.

[107] C. Garcia and M. Delakis, "Convolutional face finder: a neural architecture for fast and robust face detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1408–1423, November 2004.

[108] A. Hadid and M. Pietik?inen, "Color-based face detection using skin locus model and hierarchical filtering," in *International Conference on Pattern Recognition*, August 2002, pp. 196–200.

[109] P. Bojanowski, I. Laptev, J. Ponce, C. Schmid, and J. Sivic, "Finding actors and actions in movies," in *IEEE International Conference on Computer Vision*, December 2013, pp. 2280–2287.

[110] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network based face detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–28, January 1998.

[111] L. Wiskott, J. M. Fellous, M. Kruger, and C. Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, July 1997.

[112] T. F. Cootes, G. J. Edwords, and T. C. J., "Active appearance models," in *European Conference on Computer Vision*, vol. 2, June 1998, pp. 484–498.

[113] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, p. 2005, 2005.

[114] M. J. Black and A. D. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

[115] G. R. Bradski, "Computer video face tracking for use in a perceptual user interface," *Intel Technology Journal Q2*, vol. 2, no. 2, pp. 12–21, 1998.

[116] J. Sivic, F. Schaffalitzky, and A. Zisserman, "Object level grouping for video shots," in *European Conference on Computer Vision*, May 2004, pp. 85–98.

[117] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 1994, pp. 593–600.

[118] S. Mckenna, S. Gong, and J. J. Collins, "Face tracking and pose representation," in *British Machine Vision Conference*, September 1996, pp. 755–764.

[119] D. Ramanan, D. A. Forsyth, and A. Zisserman, "Strike a pose: tracking people by finding stylized poses," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 271–278.

[120] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, pp. 125–141, May 2008.

[121] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[122] H. Pirsiavash, D. Ramanan, and C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 1201–1208.

[123] B. Yang, C. Huang, and R. Nevatia, "Learning affinities and dependencies for multi-target tracking using a crf model," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 1233 –1240.

[124] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. J. V. Gool, "Robust tracking-by-detection using a detector confidence particle filter," in *IEEE International Conference on Computer Vision*, October 2009, pp. 1515–1522.

[125] Y. Cai, N. de Freitas, and J. J. Little, "Robust visual tracking for multiple targets," in *European Conference on Computer Vision*, May 2006, pp. 107–118.

[126] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[127] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *European Conference on Computer Vision*, Octobor 2008, pp. 788–801.

[128] B. Song, T. Jeng, E. Staudt, and A. K. Roy-chowdhury, "A stochastic graph evolution framework for robust multi-target tracking," in *European Conference on Computer Vision*, September 2010, pp. 605–619.

[129] J. Luo, M. Boutell, and C. Brown, "Pictures are not taken in a vacuum - an overview of exploiting context for semantic scene content understanding," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 101–114, March 2006.

[130] Y. Song and T. Leung, "Context-aided human recognition - clustering," in *European Conference on Computer Vision*, May 2006, pp. 382–395.

[131] J. Xing, H. Ai, and S. Lao, "Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 1200–1207.

[132] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Comput. Surv.*, vol. 38, December 2006.

[133] T. Zhao and R. Nevatia, "Tracking multiple humans in complex situations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 1208–1221, September 2004.

[134] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 685–692.

[135] C.-H. Kuo and R. Nevatia, "How does person identity recognition help multi-person tracking ?" in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 1217–1224.

[136] B. Yang and R. Nevatia, "Online learned discriminative partbased appearance models for multi-human tracking," in *European Conference on Computer Vision*, Octobor 2012, pp. 484–498.

[137] B. Yang and R. Navatia, "An online learned crf model for multi-target tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2034–2041.

[138] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B*, vol. 48, no. 3, pp. 259–302, 1986.

[139] A. C. Gallagher and T. Chen, "Using group prior to identify people in consumer images," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[140] D. Anguelov, K.-C. Lee, S. B. Gokturk, and B. Sumengen, "Contextual identity recognition in personal photo albums," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–7.

[141] D. Lin, A. Kapoor, G. Hua, and S. Baker, "Joint people, event, and location recognition in personal photo collections using cross-domain context," in *European Conference on Computer Vision*, vol. 1, September 2010, pp. 243–256.

[142] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1195–1209, July 2009.

[143] A. Jepson, D. Fleet, and E.-M. T.F., "Robust online appearance model for visual tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 415–422, 2001.

[144] L. Bourdev and J. Malik, "Poselets: body part detectors trained using 3d human pose annotations," in *IEEE International Conference on Computer Vision*, September 2009, pp. 1365–1372.

[145] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, November 1984.

[146] B. J. Frey and D. J. C. Mackay, "A revolution: Belief propagation in graphs with cycles," in *Neural and Information Processing Systems*, vol. 10, December 1998, pp. 479–485.

[147] L. K. Saul and M. I. Jordan, *A mean field learning algorithm for unsupervised neural networks*. MIT Press, 1999.

[148] M. Roth, M. Buml, R. Nevatia, and R. Stiefelhagen, "Robust multi-pose face tracking by multi-stage tracklet association," in *International Conference on Pattern Recognition*, November 2012, pp. 1012–1016.

[149] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro, "Particle phd filtering for multi-target visual tracking," in *International Conference on Acoustics, Speech and Signal Processing*, vol. 1, April 2007, pp. 1101–1104.

[150] M. Du and R. Chellappa, "Face association across unconstrained video frames using conditional random fields," in *European Conference on Computer Vision*, vol. 7, Octobor 2012, pp. 167–180.

[151] P. Bojanowski, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic, "Finding actors and actions in movies," in *IEEE International Conference on Computer Vision*, December 2013, pp. 2280–2287.

[152] T. K. Kim, J. Kittler, and R. Cipolla, "Discriminative learning and recognition of image set classes using canonical correlations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1005–1018, June 2007.

[153] R. Wang, H. Guo, L. Davis, and Q. Dai, "Covariance discriminative learning: a natural and efficient approach to image set classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2496–2503.

[154] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: a joint formulation," in *European Conference on Computer Vision*, October 2012, pp. 566–579.

[155] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? metric learning approaches for face identification," in *IEEE International Conference on Computer Vision*, September 2009, pp. 498–505.

[156] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, January 2001.

[157] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, June 2010.

[158] M. Yang, D. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *IEEE International Conference on Computer Vision*, Nov 2011, pp. 543–550.

[159] P. N. Belhumeur, D. W. Jacobs, D. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 545–552.

[160] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *Journal of Maching Learning Research*, vol. 6, pp. 1453–1484, December 2005.

[161] J. Wu and J. M. Rehg, "Centrist: A visual descriptor for scene categorization," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489–1501, 2011.

[162] K. Kurihara, M. Welling, and T. Y. W., "Collapsed variational dirichlet process mixture models," in *International Joint Conference on Artificial Intelligence*, January 2007, pp. 2796–2801.

[163] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.

[164] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: a large-scale, real-world database for facial landmark localization," in *IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, November 2011, pp. 2144–2151.

[165] Z. Cui, S. Shan, H. Zhang, S. Lao, and X. Chen, "Image sets alignment for video-based face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2626–2633.

[166] R. G. Cinbis, J. Verbeek, and C. Schmid, "Unsupervised metric learning for face identification in tv video," in *IEEE International Conference on Computer Vision*, November 2011, pp. 1559–1566.

[167] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: a database for dtudying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[168] S. Ba and J. M. Odobez, "Probabilistic head pose tracking evaluation in single and multiple camera setups," in *Multimodal Technologies for Perception of Humans*, 2008, pp. 276–286.

[169] Q. Cai, A. C. Sankaranarayanan, Q. Zhang, Z. Zhang, and Z. Liu, "Real time head pose tracking from multiple cameras with a generic model," in *CVPR Workshops*, June 2010, pp. 25–32.

[170] S. Li, X. Liu, X. Chai, H. Zhang, S. Lao, and S. Shan, "Morphable displacement field based image matching for face recognition across pose," in *European Conference on Computer Vision*, October 2012, pp. 102–115.

[171] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *European Conference on Computer Vision*, October 2012, pp. 808–821.

[172] A. Li, S. Shan, and W. Gao, "Coupled bias-variance tradeoff for cross-pose face recognition," *IEEE Trans. on Image Processing*, vol. 21, no. 1, pp. 305–315, 2012.

[173] C. D. Castillo and D. W. Jacobs, "Using stereo matching for 2-D face recognition across pose," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[174] ——, "Wide-baseline stereo for face recognition with large pose variation," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 537–544.

[175] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2707–2714.

[176] Q. Yin, X. Tang, and J. Sun, "An associate-predict model for face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 497–504.

[177] F. Schroff, T. Treibitz, D. Kriegman, and S. Belongie, "Pose, illumination and expression invariant pairwise face-similarity measure via Doppelgänger list comparison," in *IEEE International Conference on Computer Vision*, November 2011, pp. 2494–2501.

[178] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 1–8.

[179] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell, "Face recognition with image sets using manifold density divergence," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 581–588.

[180] A. C. Sankaranarayanan, A. Veeraraghavan, and R. Chellappa, "Object detection, tracking and recognition for multiple smart cameras," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1606–1624, 2008.

[181] B. Song and A. K. Roy-Chowdhury, "Robust tracking in a camera network: a multi-objective optimization framework," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 4, pp. 582–596, August 2008.

[182] B. Song, A. T. Kamal, C. Soto, C. Ding, J. A. Farrell, and A. K. Roy-Chowdhury, "Tracking and activity recognition through consensus in distributed camera networks," *IEEE Trans. on Image Processing*, vol. 19, no. 10, pp. 2564–2579, October 2010.

[183] A. K. Roy-Chowdhury and B. Song, *Camera Networks: The Acquisition and Analysis of Videos over Wide Areas*, ser. Synthesis Lectures on Computer Vision. Morgan & Claypool Publishers, 2012.

[184] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, 2003.

[185] R. Ramamoorthi, "Analytic PCA construction for theoretical analysis of lighting variability in images of a convex lambertian object," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 10, pp. 1322–1333, 2002.

[186] L. Zhang and D. Samaras, "Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 351 –363, March 2006.

[187] Z. Yue, W. Zhao, and R. Chellappa, "Pose-encoded spherical harmonics for face recognition and synthesis using a single image," *EURASIP Journal on Advances in Signal Process*, vol. 2008, pp. 1–18, January 2008.

[188] T. Brocker and T. Dieck, *Representations of Compact Lie Groups*. Springer, 2003.

[189] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, June 2003, pp. 156–164.

[190] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 1998, pp. 232–237.

[191] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[192] M. Du, A. C. Sankaranarayanan, and R. Chellappa, "Face tracking and recognition in a camera network," in *Multibiometrics for Human Identification*, B. Bhanu and V. Govindaraju, Eds. Cambridge University Press, 2011, pp. 235–257.

[193] R. Chellappa, M. Du, P. K. Turaga, and S. K. Zhou, "Face tracking and recognition in video," in *Handbook of Face Recognition, 2nd Edition*, S. Z. Li and A. K. Jain, Eds. Springer, 2011, pp. 323–351.

[194] X. He, D. Cai, and W. Min, "Statistical and computational analysis of locality preserving projection," in *International Conference on Machine Learning*, August 2005, pp. 281–288.

[195] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural and Information Processing Systems*, December 2012, pp. 1097–1105.

[196] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.

[197] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 886–893.

[198] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition," *IEEE Trans. on Image Processing*, vol. 16, no. 1, pp. 57–68, January 2007.

[199] M. Osadchy, Y. LeCun, and M. Miller, "Synergistic face detection and pose estimation with energy-based models," *Journal of Machine Learning Research*, vol. 8, pp. 1197–1215, May 2007.

[200] Y. Taigman, R. Yang, M., M. A., and L. Wolf, "Deepface: closing the gap to human-level performance in face verification," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2014.

[201] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2233–2246, November 2012.

[202] A. Nedic and D. Bertsekas, "Convergence rate of incremental subgradient algorithms," *Stochastic Optimization: Algorithms and Applications*, vol. 54, pp. 223–264, 2001.

[203] J. T., F. T., and C. J. Yu, "Cutting-plane training of structural svms," *Machine Learning*, vol. 77, no. 3, pp. 27–59, October 2009.