Towards Realistic Human Analytics

LIU LUOQI

A THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE

2015

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety.

I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Liu Luoqi

July 2015

Acknowledgements

After four years' living and studying in Singapore, my PhD career will end soon. This period will be the most valuable treasure in my whole life. The starting two years of study are really tough. But fortunately, with the help and support of my supervisor and lab mates, soon I fit into the right routine quickly and achieve satisfying academic performance.

Firstly, I would like to give my special gratitude to my supervisor Prof. Yan Shuicheng. It is a great honor for me to pursue my PhD under his supervision. His ambition, diligence and conscientiousness teach me how to become an excellent individual; his innovation, broad vision and solid theory foundation tell me the way to do world-class research; his kindness and patience make him become a perfect work mate. The valuable instructions of him will benefit my career throughout my life.

I also would like to express my thanks to my seniors: Li Annan, Niu Zhiheng, Xing Junliang, Liu Si, Feng Jiashi, Tam V. Nguyen and Cheng Bin. I can not accomplish most of my research works without the great endeavor of them. Their advice and encouragement help me go through some hard time.

Furthermore, I would like to thank my previous roommates. Thank Huang Junshi, Dong Jian, Xia Wei, Xiong Chao and Wei Yunchao for their care for my living. Thank Ms. Fu Quanhong for her valuable and crucial efforts in revising my draft. I also thank all other members in Learning and Vision Group. It is great fortune and memory to work with all of them.

Last but not least, I would like to thank my parents, grandparents and wife for

their all-enduring and selfless love all these years. Their patience and companionship has been the most indispensable support during the period of my PhD life. This dissertation is dedicated to them.

Contents

Summary

Human analytics is one of the fundamental research directions in the realm of computer vision. It includes various human-oriented vision tasks, such as the analysis, recognition and synthesis of face and human body. Unlike most previous works under the constrained environment, applications in the realistic environment suffer from scale issues, complex environment and aging of the human. In spite of the enormous efforts within the field, these problems have not been well addressed. There is still a considerable gap between current academic progress and practical applications in the realistic uncontrolled environments.

In this thesis, we focus on addressing some of the key challenges towards realistic human analytics, including datasets in the unconstrained environment, low quality surveillance, scale issue, temporal change caused aging and artificial makeover. To be more specific, several "in the wild" datasets are constructed, which are carefully designed in consideration of these issues. These datasets are considered as benchmarks and made publicly available to boost the research in these directions. Several algorithms are also proposed and evaluated on these benchmarks which provide considerate improvements over the state-of-the-art approaches.

List of Tables

List of Figures

Chapter 1

Introduction

Human analytics is one of the fundamental directions in the research field of computer vision. With the growing popularity of digital devices, there has been a great interest of human-oriented applications in biometric authentication, surveillance, robotics, health care, human-computer interaction, and multimedia analysis. Due to the complexity and flexibility of the real world environment, there is still a considerate gap between industrial applications and research in laboratories. To bridge the gap, in this thesis, We aim to address several specific problems in the scope of human analytics, to acquire a better understanding of the challenges imposed in real-world environment, and explore possible solutions.

In the literature of computer vision and pattern recognition, human analytics is closely related to various topics targeting at the analysis, recognition and synthesis of human characteristics in image and video content, such as detecting the occurrence of human [38], recognizing human identities [45], parsing body items [31, 32, 99, 89] and synthesizing human appearances [136]. In order to reach these goals, several biometrics with regard to human biological, physical and behavioral characteristics are exploited, such as human face, palm print, fingerprint, retina, veins, DNA, iris, voice, signature and gait. The related technologies have a huge potential of applications in the security field, such as ATM, CCTV monitoring and entrance access control systems, and Internet products, such as face tagging in photo albums and content based image retrieval.

Due to the diversity and complexity of related technologies and applications, researches within the field are prone to study each problems individually, instead of merging all the topics into one unified framework. Among all these problems, face and human body usually serve as the primary biometric characteristics, and receive extensive attention from both academic and industrial communities.

Face and body biometrics characteristics can be roughly categorized into intrinsic and extrinsic traits. In the analytics of intrinsic traits, face and human recognition is one of the most fundamental problems, and has been throughly studied for decades [138, 11, 59, 167, 10, 142, 37, 118, 51]. Face recognition can be categorized into two sub-problems, face identification and face verification. Face identification is aimed to determine the identity of an unknown input face, while face verification is to verify the authenticity of previously claimed identity. Human recognition generally refers to recognition using face and body information. Here, we restrict human recognition to a more specific setting, where human recognition is conducted in the non-overlapped multi-camera environment and face appearance information can not be used for recognition. In this setting, the human recognition problem is often called person re-identification.

Besides of recognition based on face and human intrinsic characteristics, analytics of extrinsic traits, such as clothes, makeup and hairstyle, receives more research attention. Researches towards these topics are directly driven by fashion industry demand, and reveal very large market potential.

Previous works on human analytics related topics are often developed under the laboratory settings, while ignore the influence of variations presented in the real-world scenario. It is common that a model trained on the data set collected in well-controlled laboratory conditions is likely to experience a dramatic performance drop, if directly applied to real industrial applications. The main challenges of realistic human analytics lie in the large intra-class distance brought by variations in terms of scalability, aging, pose, occlusion, illumination, expression and resolution. The diversity of such local variations usually require very large datasets to train well-generalized models. But this kind of datasets rarely exist in this research area.

In spite of various previous efforts in addressing aforementioned issues, real world human analytics still remains a yet-to-solve topic. In particular, most publicly available data sets only have thousands to tens of thousands of images, which is far from enough to evaluate realistic systems. It is common that faces and human appearances will change as time lapses, but a very limited number of systems involve a robust strategy with the consideration of temporal changes. Pose, occlusion, complex illumination and low resolution can cause misalignment issues and appearance changes, which lead to a low discriminative ability.

Due to the challenges discussed above, how to properly handle these realistic problems towards robust human analytics is what to be studied in this thesis.

1.1 Background and Related Works

In this section, a literature review is presented for human analytics. Instead of listing all the related topics, my focus will be placed on several important problems of the primary human characteristics, such as face and human body. After introducing each related topic, several challenges for realistic human analytics will be discussed.

1.1.1 Classical Tasks for Human Analytics

Face Recognition

Face recognition aims at recognizing identities from human faces in images and videos automatically. It usually performs in two different settings: face verification and face identification. Face verification performs in the scenario where a query image/video clip matches one-to-one to a template image/video clip to verify the authenticity of the pre-claimed identity. With the popularity of the Labeled Faces in the Wild (LFW) [65] dataset and YouTube Faces dataset [148], plenty of research works have emerged for the face verification problem. Simonyan *et al.* [127] proposed Fisher vectors on densely sampled SIFT [105] features and the high dimensional features were reduced by discriminative metric learning. Chen *et al.* [21] increased LBP [3] features into high dimension and achieved similar performance with Fisher

vector. With the recent breakthrough of deep learning [74], many deep networks have been applied to the face verification problem. Huang *et al.* [64] used a local convolutional Restricted Boltzman Machine (RBM) [60] and achieved comparable performance with Fisher vector and high-dimensional LBP features. Chopra *et al.* [24] proposed the Siamese structure [159, 100] where input face pairs were mapped into a semantic space to approximate the distance metrics in the original space. With hundreds of thousands of extra labeled face data, several works claimed that they reduced the gap or even surpassed the human-level performance on LFW dataset. With 3D face alignment and millions of faces from 4,000 identities, Taigman *et al.* [134] reached 97.35% which is very close to human-level performance. Sun *et al.* [129] exploited the GooLeNet structure [133] and joint identification-verification loss which achieved 99.53%, surpassing human-level performance.

Face identification performs in the other scenario where a query image/video clip matches one-to-many against all the template images/video clips to determine the identity of the query face. The traditional face identification assumes faces lie in the linear or nonlinear subspace, and several subspace learning methods are proposed such as Eigen face [138], Fisher face [11], Laplacian face [59], and independent component analysis [10]. When face identification is applied to videos, temporal coherence information may be used as identification cues. Zhou *et al.* [171] proposed a state space model to characterize face movements to enhance face identification. Liu and Chen [104] tried to learn pose changes and head motion for identification using hidden Markov models, while some researchers [78, 93] constructed pose manifold to learn the transition matrix, and then used this transition information for identification.

Person Re-identification

Following the categorization in [142], research works on person re-identification can be roughly divided into two categories, i.e. *feature* and *learning*. Gheissari *et al.* [45] proposed to use local motion features to re-identify people across camera views. In this approach, correspondence between body parts of different persons is obtained

through space-time segmentation. Color and edge histograms are extracted on these body parts. Person re-identification is performed by matching the body parts based on the features and correspondence. Wang *et al.* [143] proposed shape and appearance context for person re-identification. The shape and appearance context was utilized to compute the co-occurrence of shape words and visual words. Farenzena *et al.* [37] represented the appearance of a pedestrian by combining three kinds of features, i.e., weighted color histograms, maximally stable color regions and recurrent highly structured patches respectively. The above mentioned features are sampled according to the symmetry and asymmetry axes obtained from silhouette segmentation.

Besides exploring better hand crafted features, learning discriminant models on low-level visual features is another popular way to tackle the problem of person re-identification. Gray and Tao [51] used AdaBoost to select an optimal ensemble of localized features for pedestrian recognition. They claimed that the learned feature is robust to viewpoint changes. Schwartz and Davis [123] used partial least squares to perform person re-identification. In this work, high dimensional original features are projected into a low dimensional discriminant latent subspace learned by Partial Least Squares. Person re-identification was performed in the latent subspace. Prosser *et al.* [118] treated person re-identification as a pair-wise ranking problem. And they used ranking SVM to learn the ranking model. In recent years, metric learning becomes popular in person re-identification. Zheng *et al.* [169] proposed a probabilistic relative distance comparison model. The proposed model maximizes the probability that the distance between a pair of true match is smaller than that between an incorrect match pair. Therefore the learned metric can achieve good results in person re-identification. Besides the above mentioned methods, Zheng *et al.* [170] extended the person re-identification approach in [118, 169] to set-based verification by transfer learning. Hirzer *et al.* [63] proposed a more efficient metric learning approach for person re-identification. In this approach, a discriminative Mahalanobis metric learning model was used. With some relaxations, this model can be efficient, and faster than previous approaches.

Facial Beautification

Facial beautification [101, 102] includes beautification of facial geometry, appearance and hairstyle. Facial geometry beautification mainly aims to enhance the attractiveness of facial shapes and components. Leyvand *et al.* [83] proposed a data-driven approach to optimize facial shapes. A beautification engine was proposed to predict the attractiveness score, and beautification was applied by calculating a 2D warp field mapping from the original face to the beautified face according to the corresponding landmark locations. Facial appearance is often beautified with cosmetics, such as foundation, lip gloss and eye shadow. Most of the studies utilize image pairs with before-and-after makeup effects. Tong *et al.* [136] extracted makeup from before-and-after training image pairs, and transfered the makeup effect defined as ratios to a new testing image. Scherbaum *et al.* [122] used 3D image pairs of before-and-after makeup, and modelled makeup as the ratio of appearance. Guo and Sim [55] considered makeup effects existing in two layers of the three-layer facial decomposition result, and makeup effects of a reference image were transferred to the target image.

1.1.2 Challenges of Realistic Human Analytics

There are large gaps between the laboratory environment and realistic conditions. Many algorithms designed in the well-controlled environment do not have good generalization ability when applied to less-constrained datasets. In this section, we review challenges of realistic human analytics and how they are handled by current research works.

Datasets in the Wild

Most traditional datasets for human analytics are collected in the laboratory condition and limited in capacity. AR [108], Yale [11], PIE [126] and FERET [52] for face recognition, were collected in a short time and in a particular laboratory location. The advantages of this type of datasets lie in the parameters, such as pose directions

and illumination intensity which can be easily controlled. Researchers can easily figure out how some factors influence designed algorithms. However, because faces are sampled from a relatively narrow distribution, the trained models cannot generalize well in the realistic environment.

Besides, most of these datasets have a quite small scale. For face recognition, VidTIMIT database [121] contains 43 subjects reciting short sentences. There are 10 sentences for each subject. Honda/UCSD database [79, 80] contains two databases of 20 and 15 subjects, respectively. CMU MoBo database [53] contains 24 walking people with 96 sequences. NRC-IIT Face Video database [48] contains 22 video sequences of 11 subjects. Similarly, in person re-identification, VIPeR [50], i-LIDS [168] and ETHZ [123] are the most frequently used datasets. They are limited in sample numbers and camera views. For example, VIPeR only consists of 632 image pairs captured from 2 cameras.

To bridge the gap between laboratory condition and realistic environment, Labeled Faces in the Wild (LFW) [65] dataset was proposed for face recognition in the unconstrained conditions. This dataset contains $13,233$ face images from $5,749$ different individuals. It is collected from daily photos, which is full of variability in pose, lighting, focus, resolution, expression, age, gender, race, accessories, makeup, occlusions, background and photo quality. Similar to LFW dataset, YouTube faces dataset [149] focuses on face recognition in the unconstrained videos. It consists of $1,595$ human subjects and $3,425$ videos collected from Youtube.com. Celebrities on the Web (CFW) [166] database is the only truly large scale database in the wild. It contains 2.45 million images of $421,436$ celebrities, which is much larger than any other previously released face database.

Scalability

When scaling from thousands to millions of variables, the scalability becomes a big challenge for most algorithms [6, 124]. Based on this observation, some technologies have been proposed to scale up existing recognition methods. Yuan *et al.* [164] combined rejection classifiers into a cascade to speed up nearest neighbor search

for face identification. Searching in Kd-tree [113, 81] only need $O(\log N)$ time cost, where N is the number of variables. Locality sensitive hashing (LSH) [7] constructed several hash functions to control the collision probability of samples based on their distances. Wu *et al.* [154] built a scalable face image retrieval system. In the indexing stage, face features were quantized into visual words and encoded into a very small hamming signature. This signature is as small as 40KB for $1,000$ images, so it is quite suitable for large scale retrieval. However, these strategies can only speed up nearest neighbor search. Though nearest neighbor search is low in the computation complexity, it hardly achieves high performance.

Sparse representation based classification (SRC) methods [152, 165, 34, 112, 158, 155] have proved their effectiveness for a variety of problems, but they are only workable for small or medium scale problems. For large scale problems, even the calculation of the first order gradient with respect to all variables is a heavy burden. The sparsity of a solution provides a strong prior on the solution, and some recent methods have partially utilized the prior. The feature-sign search algorithm [77] to solve the basis pursuit denoising problem (BPDN) [23] and grafting algorithm [117] divide the problem into a much simpler subproblem at each iteration, but the gradient with respect to all variables still need be calculated. These methods do not scale well on large-scale problems.

Aging

The research literature on aging-invariant human analytics is quite limited over the past decades. Aging causes large facial appearance change, and thus results in performance drop for face related problems. FG-NET [2] and MORPH [119] databases are the most widely used databases for this problem. The FG-NET database contains only $1,002$ images of 82 subjects from age 0 to 69. The MORPH database contains two subsets: MORPH album 1 and MORPH album 2. MORPH album 1 contains $1,690$ images of 625 subjects, and MORPH album 2 contains $15,204$ images of $4,039$ subjects. However, each subject in the MORPH database only has averagely about three images with a small age gap, which makes it inappropriate

for modeling the aging process for aging face recognition with large age gaps.

Several research works are proposed for this problem. Geng *et al.* [44] modeled the face aging patterns. Park *et al.* [116] proposed a 3D aging model, which can capture the aging pattern in the 3D domain. Wu *et al.* [153] used a parametric craniofacial growth model to model the facial shape change. These methods can model the aging process of the face shape or texture, but are weak in the discriminative capacity. Li *et al.* [88] proposed a discriminative model for age-invariant face recognition. They used scale invariant feature transform (SIFT) and multi-scale local binary patterns (MLBP) as local descriptors. Multi-feature discriminant analysis (MFDA) was proposed to process the two local feature spaces in a unified framework.

1.2 Thesis Focus

As mentioned in literature review, most research works on human analytics failed to address realistic situation. These realistic issues, such as scalability, aging, in-the-wild environment and clothing variance, do not largely influence performances in the constrained environment, but in the realistic environment the performance of applications are largely harmed. To fill the gap between laboratory and realistic applications, we will focus on some of the essential problems and their challenges in human analytics. More specifically, we study the impact of large-scale and in-the-wild issue on video-based face identification; how aging affecting face verification; how to use clothing information to assist person re-identification in low quality surveillance environment; how to achieve realistic makeover. These problems and issues are studied as follows.

- Large Population Face identification in Unconstrained Videos. we investigate large-scale face identification in unconstrained videos with one thousand subjects. This problem is very challenging, and until now most studies have only considered the scenarios with a small number of subjects and videos captured in controlled laboratory environments. In contrast, we firstly set up a

large-scale video database in unconstrained environment, Celebrity-1000, for face identification research. Moreover, a sparsity-induced scalable optimization method (SISO) is presented, which solves the large-scale Multi-Task Joint Sparse Representation (MTJSR) problem by sequentially solving a series of smaller-scale subproblems with theoretically guaranteed convergency [97].

- Deep Aging Face Recognition with Large Gaps. Along with the long-time evolution of popular social networks, e.g. Facebook, it inevitably comes to the era to consider face/user recognition with large age gaps. However, related research with adequate subjects and large age gaps is surprisingly rare. In this work, a so-called Cross-Age FacE (CAFE) dataset is collected, with more than 900 celebrities. The face images of each subject are captured with large age gaps, ranging from child, young, adult, to old groups. Then, a novel framework is proposed, called Deep Aging Face Recognition (DAFR), for this challenging task. DAFR includes two modules, aging pattern synthesis and aging face verification. The aging pattern synthesis module synthesizes the faces of all age groups, and the core structure is a deep aging-aware denoising auto-encoder (a^2-DAE) with multiple outputs of different age groups. The aging face verification module then takes the synthesized aging patterns of a face pair as the input, and each pair of synthesized images of the same age group is fed into a parallel CNN, and finally all parallel CNN outputs are fused to provide similar/dissimilar prediction.

- Clothing Attributes Assisted Person Re-identification. Due to the realistic surveillance environment, variety of human pose, low video quality, occlusion and missing of identifiable faces make the recognition problem really tough. In this situation, clothing appearance becomes the main cue for identification purpose. We present a comprehensive study on clothing attributes assisted person re-identification. First, the body parts and their local features are extracted for alleviating the pose-misalignment issue. A latent SVM based person re-identification approach is proposed to describe the relations among

the low-level part features, middle-level clothing attributes, and high-level re-identification labels of person pairs. Motivated by the uncertainties of clothing attributes, they are treated as real-value variables instead of using them as discrete variables. Moreover, a large-scale real-world dataset with ten camera views and about 200 subjects is collected and thoroughly annotated for this study [85, 84].

- Makeup and Hairstyle Recommendation and Synthesis. We propose a so called Beauty e-Experts system [96, 94], for realistic hairstyle and facial makeup recommendation and synthesis. Given a user-provided frontal face image with short/bound hair and no/light makeup, the Beauty e-Experts system can not only recommend the most suitable hairdo and makeup, but also show the synthetic effects. Two problems are considered for the Beauty e-Experts system: what to recommend and how to wear, which describe a similar process of selecting hairstyle and cosmetics in our daily life.

1.3 Thesis Overview

In Chapter 2, we study into large-population face identification in unconstrained videos. Then in Chapter 3, a deep learning based approach is proposed for face verification with large age gaps. In Chapter 4, clothing attributes assisted person re-identification is proposed. Finally, the Beauty e-Experts system is introduced for facial makeup and hairstyle synthesis in Chapter 5.

Chapter 2

Large Population Face Identification in Unconstrained Videos

In this chapter, large-population face identification is investigated in unconstrained videos [97]. Video based face identification is an important problem, but a large-scale realistic dataset is lacked for research in this direction. In this work, a challenging Celebrity-1000 dataset is constructed from realistic environment, which contains 1000 subjects and millions of video frames. The faces in the dataset are captured with the presence of various pose, illumination, expression and resolution change. We apply multi-task joint sparse representation (MTJSR) algorithm and accelerate it with sparsity induced scalable optimization (SISO), to effectively address this problem.

2.1 Introduction

Face identification, one of the most practical problems in human analytics, has attracted much attention in the past decades [138, 11, 59, 167, 10]. Unlike face verification [149, 65], which is to verify the authenticity of previously claimed identities, face identification need determine the identity of an unknown input face. There

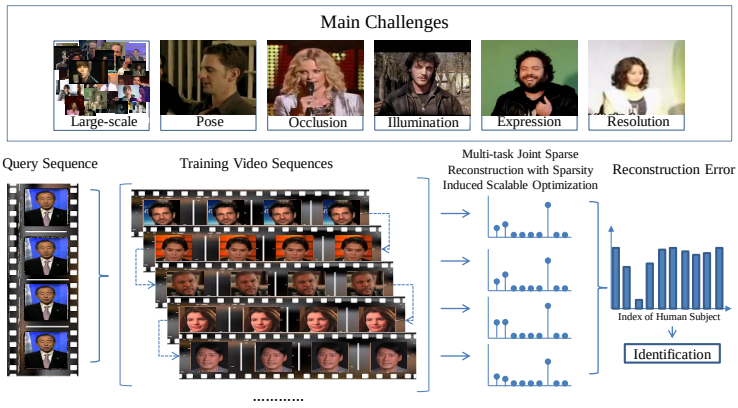


Figure 2.1: We construct the Celebrity-1000 database, containing $1,000$ celebrities with $\sim 160K$ tracking sequences, for face identification. The task is challenging due to the involved occlusions, various poses, illuminations, expressions and image resolutions. The training-free Multi-Task Joint Sparse Representations (MTJSR) algorithm is used owing to its natural capability in integrating the information from multiple frames for collaborative inference, and this work mainly focuses on accelerating the MTJSR algorithm for large-scale video based face identification.

exist two categories of face identification tasks: single image based and video (i.e. image set) based. Most traditional methods [138, 11, 59, 10] focus on single image based face identification. However, with the increase of affordable video cameras and large capacity storage, face identification can be performed with enriched information. Compared with single image based face identification, video based face identification has at least two advantages: 1) each additional frame adds extra information to the video, which may make identification more accurate and robust, and 2) dynamic information in the video may be useful for identification. Up to now, many methods for video based face identification have been proposed, but most of them are conducted on small or moderated databases due to the deficiency of large-scale unconstrained databases. For example, Wong *et al.* [151] placed three cameras above some natural choke points to capture pedestrians walking through portals in real world surveillance conditions, but there were only 54 videos in total. Wolf *et al.* [149] constructed a large 'YouTube Faces' database consisting of $1,595$ human subjects and $3,425$ videos, but their main purpose was face verification. Therefore, a large-scale unconstrained video database and studies on it are quite desirable. Our

work is motivated by this observation.

In this work, we firstly construct a large-scale unconstrained video database, Celebrity-1000, consisting of $1,000$ human subjects and $7,021$ video clips, downloaded from YouTube and Youku. We apply a multi-view face detector and tracker on these videos to retrieve continuous face sequences. The identity of each tracking sequence is manually confirmed. Then we evaluate the performances of the state-of-the-art face identification algorithms in terms of accuracy and scalability. Finally we focus on multi-task joint sparse representation (MTJSR) algorithm, which is training-free and can naturally make use of the contexts among all the frames. The joint sparsity of the linear representations of all frames within a tracking sequence is achieved by penalizing the sum of ℓ_2-norms of the blocks of coefficients [165]. We propose a sparsity induced scalable optimization (SISO) method to boost the efficiency, and make it scalable for experiments on such a large-scale database. The idea of SISO comes from the strong prior of sparsity: when reaching the optimum, the values of most variables are zeros. We maintain an active variable set, and automatically expand and shrink this set. In this way, some variables can be simply instantiated by zeros and the original problem can be simplified into a sequence of much smaller-scale subproblems, which can be solved efficiently by any proper off-the-shelf group sparsity optimization method, e.g. Accelerated Proximal Gradient (APG) method [137].

2.2 Related Work

Video based face database: Though there are already many video based face databases [149, 151, 121, 110, 79, 80, 53, 71, 48], none of them focuses on the large-scale face identification problem in unconstrained videos. VidTIMIT database [121] contains 43 subjects reciting short sentences. There are 10 sentences for each subject. Honda/UCSD database [79, 80] contains two databases of 20 and 15 subjects, respectively. CMU MoBo database [53] contains 24 walking people with 96 sequences. NRC-IIT Face Video database [48] contains 22 video sequences of 11

subjects. These databases are designed for the small-scale face recognition, and most of them are collected in the indoor environment. XM2VTSDB [110] database contains 295 subjects of totally 1,180 video sequences, and each sequence is about 5 seconds long, but it is also collected for constrained face recognition. ChokePoint database [151] is designed for the face verification task under the surveillance environment. Three cameras were placed above choke points to capture pedestrians walking through portals, which is still limited in camera views and semi-indoor environment. The database size is also limited. It contains 48 video sequences of 25 subjects in portal 1 and 29 in portal 2. YouTube Faces database [149] is the only one focusing on large-scale face recognition in the unconstrained environment. It consists of 1,595 human subjects and 3,425 videos collected from Youtube.com. However, it is designed for face verification, but not suitable for face identification. It only has 2.15 videos per subject on average, and 591 of 1,595 subjects only have one video.

It is worth noting that there are also some image based face databases in the wild. Labeled Faces in the Wild (LFW) database [65] provides a standard testing benchmark and attracts the interest of thousands of researchers to test their algorithms on it. However, this database has only 13,000 images, so it is not a large scale database and is designed only for face verification. Celebrities on the Web (CFW) [166] database is a truly large scale database in the wild. It contains 2.45 million images of 421,436 celebrities, which is much larger than any other previously released face database, but it is still image based. Therefore, there is still an urgent need for a large scale video based database in face identification research.

Video based face recognition algorithms: There are many works on video based identification in recent years [125, 171, 104, 78, 147, 93, 141, 9, 19]. On one hand, many algorithms tried to learn temporal coherence as an identification cue for face identification. Zhou *et al.* [171] proposed a state space model to characterize face movement to enhance face identification. Liu and Chen [104] tried to learn pose change and head motion for identification using Hidden Markov Models, while some researchers [78, 93] constructed pose manifold to learn transition matrix, and

then used this transition information for identification. All these works were computationally expensive to extract the dynamic information, which made them not suitable for large-scale databases. Besides, their performance strongly relied on stable temporal coherence in training and testing. On the other hand, models without learning temporal coherence were also widely studied. Shakhnarovich *et al.* [125] assumed that the multiple frames for each human subject followed the Gaussian distribution, and they used Kullback-Leibler divergence to measure the distance between distributions. Their assumption was too strong and their performance relied on the accurate distribution estimation. Wang *et al.* [141] regarded each video sequence as a manifold, and sought to learn an embedding space, where manifolds of different subjects were better separated. This method required a large number of frames to build a stable manifold for each human subject. If the manifold was not built well, the performance would drop dramatically. Wolf and Shashua [147] calculated a new positive definite kernel for each pair of image sets based on principal angles between two linear subspaces, and then trained SVM for identification. All aforementioned algorithms suffered from heavy off-line training load, and their experiments were only conducted on small or moderated scaled databases.

Technologies for scalable object/face recognition: When scaling from thousands to millions of variables, the scalability becomes a big challenge for most object/face recognition algorithms [6, 124]. Based on this observation, some technologies have been proposed to scale up existing recognition methods. Yuan *et al.* [164] combined rejection classifiers into a cascade to speed up nearest neighbor search for face identification. Searching in Kd-tree [113, 81] only needs $O(\log N)$ time cost, where N is the number of variables. Locality sensitive hashing (LSH) [7] constructs several hash functions to control the collision probability of samples based on their distances. Wu *et al.* [154] built a scalable face image retrieval system. In the indexing stage, face features are quantized into visual words and encoded into a very small hamming signature. This signature is as small as 40KB for $1,000$ images, so it is quite suitable for large scale retrieval. In the retrieval stage, candidate images are retrieved and re-ranked based on this signature. However, these strate-

gies can only speed up nearest neighbor search. Though nearest neighbor search is low in the computation complexity, it hardly achieves high performance in the face identification problem.

Sparse representation based classification (SRC) methods [152, 68, 165] have proved their effectiveness for a variety of problems, but they are only workable for small or medium scale problems. For large scale problems, even the calculation of the first order gradient with respect to all variables is a heavy burden. The sparsity of a solution provides a strong prior on the solution, and some recent methods have partially utilized the prior. The feature-sign search algorithm [77] to solve the basis pursuit denoising problem (BPDN) [23] and grafting algorithm [117] divide the problem into a much simpler subproblem at each iteration, but the gradient with respect to all variables still need be calculated. These methods do not scale well on large-scale problems. Besides, it is hard to apply these methods for multi-task joint sparse representation problem.

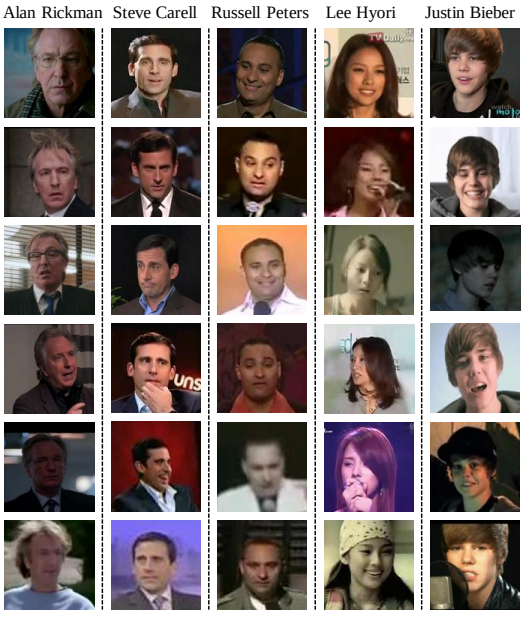


Figure 2.2: Some exemplar cropped frames in the constructed Celebrity-1000 database. From left to right, the celebrities are Alan Rickman (United Kingdom), Steve Carell (United States), Russell Peters (Canada), Lee Hyori (South Korea) and Justin Bieber (United States). Note that the difficulties caused by pose, occlusion, illumination, expression and low resolution are generally encountered in this database.

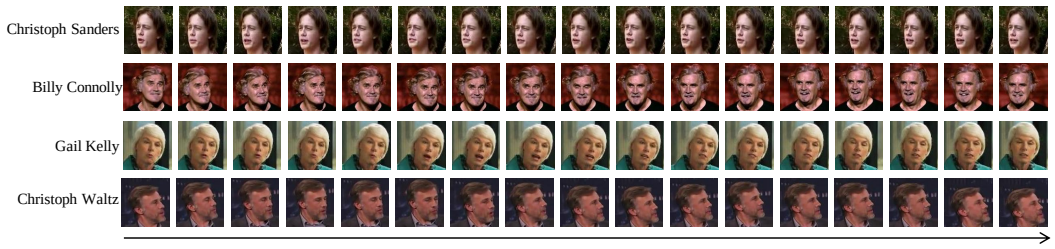


Figure 2.3: Some exemplar sequences in the constructed Celebrity-1000 database. From the first to last row, the celebrities are Christoph Sanders (United Kingdom), Billy Connolly (Scotland), Gail Kelly (Australia), and Christoph Waltz (Germany, Austria).

2.3 Celebrity-1000 Database

We collect a list of celebrities' names, and then crawl a video database of $1,000$ celebrities using the name list as queries on YouTube and Youku. The celebrities' identities vary from Hollywood stars, Asian movie stars, singers to politicians. The video scenes include interviews, concerts, speeches and news broadcasts. Video resolution varies from 320p to 720p. Some cropped faces are presented in Figure 2.2. Some statistics are showed in Table 2.1. As far as we know, it is the first large-scale unconstrained video database for face identification.

Table 2.1: Statistics of Celebrity-1000 database.

# Total celebrities	$1,000$
# Total videos	$7,021$
# Total sequences	$159,726$
# Total frames	$2,405,379$

Table 2.2: Exemplar names of celebrities and their corresponding numbers of videos downloaded and numbers of tracking sequences.

Query	#vid	#seq	Query	#vid	#seq
Aaron Staton	5	97	Bill Mumy	3	50
Aasif Mandvi	6	337	Bon Jovi	5	72
Adam Yauch	5	196	Brian Cox	6	102
Adrien Brody	10	474	David Pyott	4	165
$\cdots$	$\cdots$	$\cdots$	Total	$7,021$	$\sim 160K$

After crawling the videos, face detection [1] and tracking are performed on each image frame to segment videos into tracking face sequences. When doing face tracking, two faces in the neighboring frames are considered in the same sequence if their overlapping region is larger than a predefined threshold. The threshold is measured by the intersection of union (IOU) of two face regions. The minimum detected face size is 20 by 20 pixels. We develop an annotation tool to manually confirm whether the tracking sequences coincide with corresponding celebrity names. The mis-tracked sequences are removed. There are $\sim 160K$ tracking sequences retained. The entire process takes more than $1,500$ hours. Finally the retained tracking sequences are sampled every 3 frames to at most 17 frames each, as shown in Figure 2.3. Since the frame rate of the original videos is 15 or 25 fps, the retained sequences are at the length of $2 \sim 3$ seconds of the original videos, which is sufficient to capture single expression or pose change. Some exemplar celebrities and numbers of videos as well as numbers of sequences are showed in Table 2.2. Celebrity-1000 database covers a wide range of poses, illumination conditions, expressions and resolutions. It contains human subjects with different age, gender, social group and ethnicity. All these differences make the identification very challenging.

We provide two types of protocols: Open-set and Close-set. Open-set is for the purpose of investigating the generalization ability of algorithms. In this test, the database is divided into training, gallery and probe subsets. The generic training set includes 200 subjects and algorithms should be trained on this set. The probe set and the gallery set are used in the testing stage. The identities of frames in the gallery set are known, and the frames in the probe set are matched with those in the gallery set for identification. The probe and gallery set include 800 subjects, which do not overlap with those in the training set. They are further divided into 4 scales[2]: 100, 200, 400, and 800. In Close-set test, the dataset is divided into training and testing subsets. The training and testing subsets contain the same

[1] OMRON, OKAO vision.
http://www.omron.com/r_d/coretech/vision/okao.html

[2] We use different scales since many algorithms can only work on a subset of our original database. In the following explanation, We use scale to represent the number of celebrities in the subset of the original database.

identities and are further divided into 4 scales: 100, 200, 500, and 1000. In both of the two test protocols, nearly 70% tracking sequences are randomly selected as training/gallery and the remaining 30% for testing/probe. The sequences from the same video only appear in either training/gallery or testing/probe set. Identification performance is reported on cumulative match characteristic curve (CMC) [111]. CMC curve tells the cumulative accuracy within the top k ranks. Further details of suggested test protocols and database can be referred and downloaded from website "http://www.lv-nus.org/facedb/".

2.4 Methodology

Though various methods have been exploited to address video based face identification, most of them cannot handle the large-scale database efficiently. Statistical model based methods [104, 125, 171] often encounter heavy-load parameter estimation process. Some subspace based methods like Laplacian faces [59] require a huge similarity matrix, and easily reach the memory limit. Besides, some methods, such as the one proposed by Shakhnarovich *et al.* [125], simply use the voting strategy to combine single image based face identification results, and therefore they cannot make use of the available information in videos. In consideration of the drawbacks encountered by these methods, the multi-task joint sparse representation (MTJSR) [165] is an appropriate choice to reach a good balance between accuracy and scalability. It naturally integrates all the frames in a query tracking sequence for contextual and robust inference. Different video frames are correlated with each other, which are treated as multi-task in a one-step recognition fashion. It is different from those recognition methods using independent frames.

2.4.1 Multi-task Joint Sparse Representation

Assuming that the gallery face set $X \in \mathbb{R}^{d \times p}$ contains M groups, where d is the dimension of the feature representation and p is the total number of gallery frames. A group is a subset of image frames within each subject, and a subject means the

identity of a celebrity. These groups are clustered by k-means clustering within each subject with cluster size ~ 100 frames. $X_m \in \mathbb{R}^{d \times p_m}$ indicates the image frames of the mth group, where p_m is the number of frames. $\sum_{m=1}^{M} p_m = p$. Given a test tracking sequence as an ensemble of L image frames, each probe face y^l can be formulated as linear representation over the gallery images:

$$y^l = \sum_{m=1}^{M} X_m w_m^l + \varepsilon^l, l = 1, \ldots, L, \tag{2.1}$$

where $w_m^l \in \mathbb{R}^{p_m}$ is a reconstruction coefficient vector associated with the mth group, and $\varepsilon^l \in \mathbb{R}^d$ is the residual term.

Denote $w_m = [w_m^1, \ldots, w_m^L]$ as the representation vector from the mth group across different testing frames and $W = [w_m^l]_{l,m}$. The multi-task joint sparse representation problem is formulated as multi-task least square regressions with $l_{2,1}$-norm regularization:

$$\begin{aligned} \min_W F(W) &= f(W) + \psi(W) \\ &= \frac{1}{2} \sum_{l=1}^{L} \|y^l - \sum_{m=1}^{M} X_m w_m^l\|_2^2 + \lambda \sum_{m=1}^{M} \|w_m\|_2, \end{aligned} \tag{2.2}$$

where the first term $f(W)$ is quadratic while the second term $\psi(W)$ is the non-smooth yet convex regularization term.

2.4.2 Sparsity Induced Scalable Optimization

Celebrity-1000 database contains $1,000$ celebrities, and each celebrity has $\sim 2,400$ image frames on average. When querying with a test tracking sequence of average length ~ 15, the variables in (2.2) will reach up to $1000 \times 2400 \times 15 \approx 36M$. Clearly, it is time prohibitive to directly solve (2.2) with any off-the-shelf group sparsity optimization methods.

The solution of (2.2), denoted by W^*, is usually very sparse in group level. That is, there are many groups w_m^* with all variables equal to zeros. For any W, its *group support* is defined as $\sigma(W) = \{m | w_m \neq \mathbf{0}\}$, where $w_m \neq \mathbf{0}$ means that there are

nonzero variables in w_m, and $w_m = \mathbf{0}$ means that all variables in w_m are zeros. The group level sparsity makes it possible to solve (2.2) efficiently.

Denote the original optimization problem (2.2) as P_I, where $I = \{1, \cdots, M\}$. For any $C \subseteq I$, a subproblem P_C can be defined, which maintains $|C|$ groups of variables and set all variables in other groups to be zeros. Since the optimal solution W^* is often very sparse in group level, that is, $|\sigma(W^*)| \ll M$, in fact the solution of (2.2) can be obtained by solving a proper subproblem. The following two theorems lay the groundwork.

Lemma 1. For any set $C \subseteq I$, the subproblem P_C is also convex.

This lemma can be easily proved based on the definition of convexity and the proof is ignored here.

For the solution W^*, if it is sparse, then the corresponding W^*_C (obtained by copying entries whose indices in C from W^* and setting other entries to be zeros) is also the solution to the subproblem P_C.

Lemma 2. If W^* is the solution to the problem P_I and $\sigma(W^*) \subseteq C$, then W^*_C is also the solution to the subproblem P_C.

Since $|\sigma(W^*)| \ll M$, Theorem 2 in fact points out that the solution to (2.2) can be found by solving a proper subproblem P_C which contains a much smaller number of groups. The difficulty lies in how to determine the set C.

Obviously, it is hard to obtain a proper C directly. Since group sparsity optimization problems usually have no analytic solutions, nearly all existing group sparsity optimization methods are founded on the gradient descent strategy, that is, starting from an initialization $W(0)$, moving along a path $W(0), W(1), \cdots, W(t), W(t+1), \cdots, W^*$ with $F(W)$ monotonically decreasing. To be efficient, we need to move along a path with $C(t) = \sigma(W(t))$ which is always small, that is, always dealing with a small set of groups. The sparsity property of W^* actually makes this possible. The set $C(t)$ is called *active group set*, which dynamically changes as the optimization process proceeds.

Suppose $W = W(t)$ and the current active group set $C(t)$ is small, we need to find a nearby $W(t+1)$ with $F(W(t+1)) < F(W(t))$ and $C(t+1)$ is also small.

There are two situations:

1) If $W(t)$ is not the solution to the subproblem $P_{C(t)}$, then solve the subproblem $P_{C(t)}$ first. When solving the subproblem $P_{C(t)}$, the initialization is usually set as $W(t)$ for efficiency. In the optimization process, W moves along a segment of the path with $|\sigma(W)| \leq |C(t)|$ and $F(W)$ is monotonically decreasing, some groups in $\{w_m, m \in C(t)\}$ may become all zeros and the active group set then shrinks, and thus this step is called *shrinkage phase*.

2) If $W(t)$ is the solution to the subproblem $P_{C(t)}$, then we need to judge whether $W(t)$ is already the solution to (2.2). If yes, we have the conclusion that $W^* = W(t)$. If not, we need to automatically select another proper active group set $C(t+1)$. To further optimize (2.2), a proper selection is to ensure $C(t) \subseteq C(t+1)$ and some new active groups are added into $C(t+1)$. So this step is called *expansion phase*.

Denote the Lagrangian function of the problem P_C by $L_C(W_C)$. According to the generalized KKT condition [62], if W_C^* is the solution of P_C, then

$$0 \in \frac{\partial}{\partial W_C} L_C(W_C^*). \tag{2.3}$$

(2.3) offers us a tool to judge whether the solution to the subproblem P_C is a solution to (2.2) or not. If not, (2.3) also guides us on how to expand the active group set C.

If W_C^* is the solution to the subproblem P_C, then $0 \in \frac{\partial}{\partial W_C} L_C(W_C^*)$, that is,

$$0 \in \frac{\partial}{\partial w_m} L_C(W_C^*), \text{if } m \in C. \tag{2.4}$$

By adding zeros, we can transform W_C^* into W^*.

Note that if W^* is the solution to (2.2), according to the generalized KKT condition, we have

$$0 \in \frac{\partial}{\partial w_m} L(W^*), \text{if } m = 1 \cdots M, \tag{2.5}$$

with $L(W)$ being the Lagrangian function of (2.2).

If $m \in C$, then $\frac{\partial}{\partial w_m} L(W^*) = \frac{\partial}{\partial w_m} L_C(W_C^*) = 0$, thus, we only need to consider

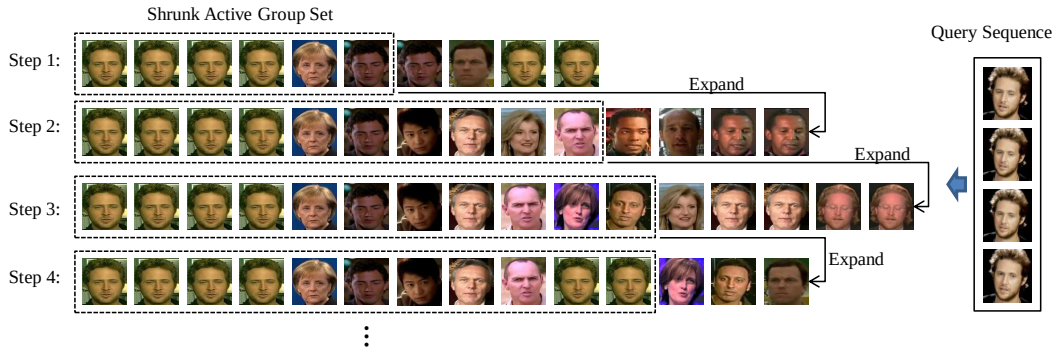


Figure 2.4: An illustration of the shrinkage-expansion process. The dotted rectangle contains active groups set after shrinkage. In each step, there are two phases: shrinkage phase and expansion phase. In shrinkage phase, groups with all zeros are excluded from the active group set. For example, in Step 1 four groups outside the dotted rectangle are excluded. In expansion phase, active group set is expanded by automatically adding more active groups.

the partial derivatives with respect to w_m, $m \notin C$. Let $B = \{m|0 \notin \frac{\partial}{\partial w_m} L(W^*)\}$, which is called *expansion set* and plays a key role in our proposed algorithm. Since for (2.2), the generalized KKT condition is also sufficient, then we can judge whether W^* is the solution to (2.2) or not according to the expansion set B. If B is empty, then W^* is already the solution to (2.2); otherwise not. At the same time, when B is not empty, in the expansion phase, we only need to add groups whose indices belong to B to the active group set C. Since $0 \notin \frac{\partial}{\partial w_m} L(W^*)$ if $m \in B$, the obtained new subproblem can definitely be further optimized. To control the complexity, when $|B|$ is too large, we only add part of groups in B into C. According to our experiments, as $F(W)$ decreases, the size of B decreases very quickly, thus this strategy only functions in the first few expansion phases.

In conclusion, in the shrinkage phase, a subproblem is solved which contains a much smaller number of groups (thus a much smaller number of variables), and for the derived solution to the subproblem, some active groups may become inactive, so the active group set shrinks; while in the expansion phase, the active group set is expanded, which then defines the new subproblem for the next shrinkage phase. The iteration of these two phases leads to an efficient iterative procedure, called *sparsity induced scalable optimization (SISO)*, as summarized in Algorithm 1. An

shrinkage-expansion process on real data is illustrated in Figure 2.4. Each line shows active group set of current step. In shrinkage phase, inactive groups are excluded; while in expansion phase, active groups will be added into the active group set.

To derive expansion set B, we need to compute the differential $\frac{\partial}{\partial w_m}L(W)$ for $m = 1, \cdots, M$, where $L(W)$ denotes the Lagrangian function of (2.2) as

$$\begin{aligned} L(W) &= f(W) + \psi(W), \\ f(W) &= \frac{1}{2}\sum_{l=1}^{L} \|y^l - \sum_{m=1}^{M} X_m w_m^l\|_2^2, \\ \psi(W) &= \lambda \sum_{m=1}^{M} \|w_m\|_2. \end{aligned} \tag{2.6}$$

$\frac{\partial}{\partial w_m}L(W)$ can be calculated as

$$\begin{aligned} \frac{\partial}{\partial w_m}L(W) &= \frac{\partial}{\partial w_m}f(W) + \frac{\partial}{\partial w_m}\psi(W), \\ \frac{\partial}{\partial w_m^l}f(w_m^l) &= X_m^T(\sum_{n=1}^{M} X_n w_n^l - y^l), \\ \frac{\partial}{\partial w_m^l}\psi(w_m^l) &= \lambda \frac{w_m^l}{\|w_m\|_2},\ m = 1 \cdots M,\ l = 1 \cdots L. \end{aligned} \tag{2.7}$$

Note that in Algorithm 1, the subproblem can be solved by any proper off-the-shelf group sparsity optimization methods, such as the APG method [137], and because the subproblem usually contains a small set of groups, most group sparsity optimization methods can solve this problem efficiently. Thus Algorithm 1 essentially provides a very general framework to accelerate group sparsity optimization on large-scale problems.

For efficiency, the size of the support of the initialization $W(0)$, namely, $|\sigma(W(0))|$, should be small. In our experiments, we simply set $W(0) = 0$. The parameter K is usually set to ensure that the subproblem can be solved very efficiently, such as $K = 20$. Algorithm 1 is an EM-style procedure, and the expansion phase expands the active group set C, thus provides a smaller upper bound of $F(W)$, which corresponds to the minimum of the subproblem P_C; while the shrinkage phase evolves

Algorithm 1 Sparsity Induced Scalable Optimization

Input: The large-scale multi-task sparse representation problem (2.2), the initialization W(0) and the parameter K.
Set $W(t) = W(0)$, k=0, and $C = \sigma(W(0))$;
while $k < k_{max}$ (set as 20 in this work) **do**
  Compute the differential $\frac{\partial}{\partial w_m} L(W(t))$ for $m = 1, \cdots, M$;
  Build the expansion set B;
  if B is empty **then**
    $W(t)$ is already the solution to (2.2), break;
  else
    If $\|B\| > K$, select K active groups with the largest average absolute gradients in B and add them into the active group set C. If $\|B\| \leq K$, add all active groups in B into C; {Expansion};
  end if
  Solve the subproblem P_C with the initialization $W(t)$ and get $W(t+1)$;
  Set $C = \sigma(W(t+1))$; {Shrinkage}
  k=k+1;
end while
Output: The solution to (2.2).

towards this upper bound, and guarantees to reach this upper bound. These two phases iterate until the global optimum of (2.2) is reached.

Since the subproblems can be solved efficiently, the main computation burden of Algorithm 1 is the calculation of sub-differential with respect to all variables which occur in the expansion phase. However, the expansion phase is called only when needed, that is, only when the optimal solution on the current active group set is obtained. According to our experiments, usually only several rounds of expansion phases are needed. At the same time, in the proposed framework, W is always sparse and the change from $W(t)$ to $W(t+1)$ only occurs at several groups.

2.4.3 Classification Rule

For each test tracking sequence of L images, the probe image y^l can be represented by groups coefficients associated with the sth subject: $y^l = \sum_{m \in \pi_s} X_m w_m^l$, where π_s is the index set for the groups belonging to the sth subject. The decision is defined as the subject with the lowest reconstruction square loss accumulated over all the

L tasks:

$$s^* = \arg\min_s \sum_{l=1}^{L} \|y^l - \sum_{m \in \pi_s} X_m w_m^l\|_2^2 \tag{2.8}$$

2.5 Experiments

In this section, the proposed MTJSR algorithm is systematically evaluated in terms of accuracy and efficiency. All the experiments are conducted on Celebrity-1000 database, and conducted on a cluster with 14 workstations. These workstations run on Linux operating system, each of which has 8 CPU cores (3GHz) and 48G memory.

2.5.1 Experiment Configurations

In all the experiments, the faces are aligned using the eye positions and resized to 80×64 pixels. Histogram equalization is conducted as pre-processing step. The features used include Local Binary Patterns [3] and Gabor features, motivated by the success achieved by Wolf *et al.* [150] for face verification task. For LBP, the face image is divided into 10×4 blocks and then extract LBP descriptors in each block. All the descriptors are then combined into a single LBP feature vector with $2,360$ dimension. For Gabor features, each image is convolved with Gabor filters with 5 scales and 8 orientations. Then 40 convolved images are down-sampled to 320 dimension each and concatenated into vector with $12,800$ dimension. The LBP and Gabor feature vectors are concatenated after separate ℓ_2-normalization. Principal Component Analysis (PCA) [70] is trained on the training set to reduce dimensions of features. 95% energy is maintained in reduced dimensions. So the reduced dimensions are $1,651$, $1,790$, $1,815$ and $1,854$ on scale 100, 200, 500 and $1,000$ of Close-set, and $1,772$ on Open-set.

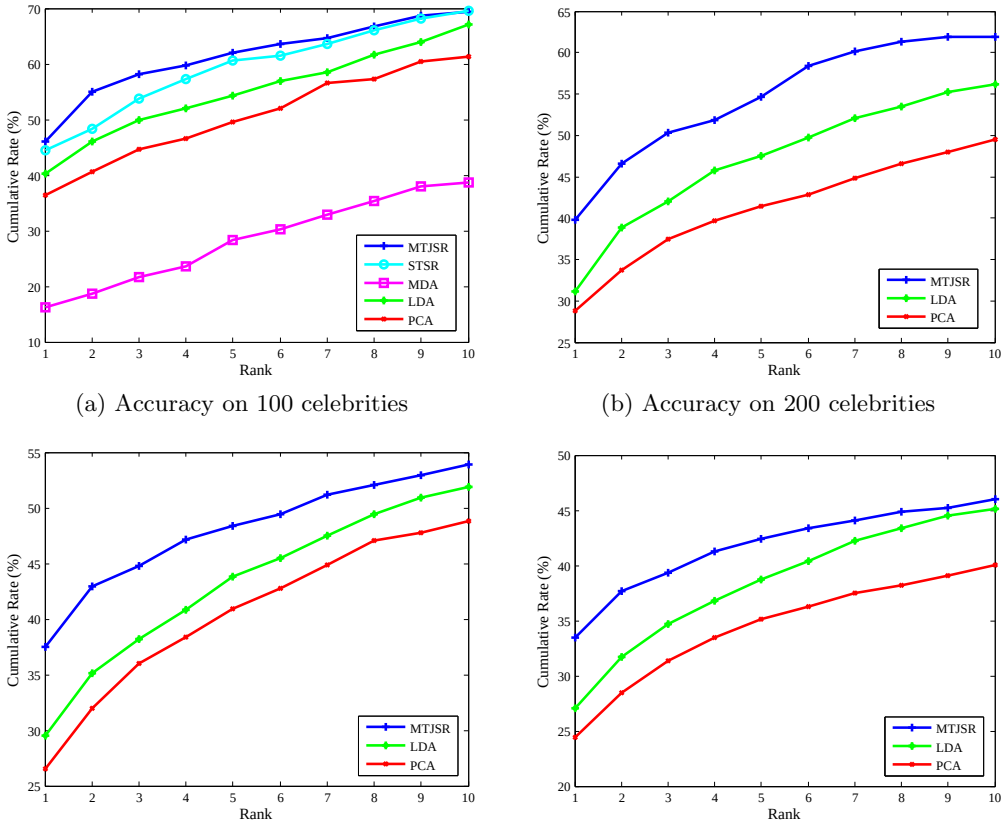
The proposed MTJSR algorithm is compared with several baseline algorithms, including Principal Component Analysis (PCA) [70], Linear Discriminative Analysis (LDA) [43], Linear Support Vector Machine (SVM) [42], Multi-instance SVM (MI-SVM) [8], Manifold Discriminant Analysis (MDA) [141] and Single Task Sparse

Representation (STSR). In STSR, each frame in the testing sequence is identified from the gallery set separately, and then the identification scores are fused to obtain the final ranking. Note that all the algorithms are based on the LBP + Gabor features reduced by PCA, so PCA is omitted here for the ease of explanation. For example, LDA actually means PCA+LDA, and SVM means PCA+SVM. PCA and LDA are followed by Nearest Neighbor (NN) for classification. The linear SVM is solved by Optimized Cutting Plane Algorithm (OCAS) [42], which significantly outperforms other SVM solvers on large-scale data. We only run MI-SVM on scale-100, since the machine will run out of memory for higher scales. For MDA, we also only show the results on scale-100, since its performance is not satisfactory on this database. MDA requires a large number of face images for each subject to build the manifold, while for Celebrity-1000, the number of selected frames for each tracking sequence is relatively small due to the storage issue. For STSR, we only run the experiments at scale-100 for efficiency. Even for scale-100 only, the total experimental time is already more than 120 hours.

For MTJSR, the regularization parameter λ is explored within $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$, and finally set to 0.1. For the accelerated MTJSR, at most $K = 20$ groups are allowed to be added into the active group set in each iteration.

2.5.2 Accuracy Evaluation: MTJSR vs. Baselines

Figure 2.5 and Figure 2.6 summarize the face identification accuracies of all evaluated algorithms on Open-set and Close-set of different scales. From the figures, we observe that MTJSR generally outperforms STSR, MDA, SVM, MI-SVM, LDA and PCA at all scales of both Open-set and Close-set. MTJSR has no requirement on the number of frames for each tracking sequence, and is thus more flexible than MDA. On scale-100, MDA shows extremely bad performance due to the small number of frames available for building the manifold within each tracking sequence. MTJSR can naturally integrate the information from multiple frames within a query tracking sequence for collaborative inference, and thus outperforms MDA. The performance of MI-SVM is not as high as MTJSR and SVM in our experiments. The underlying



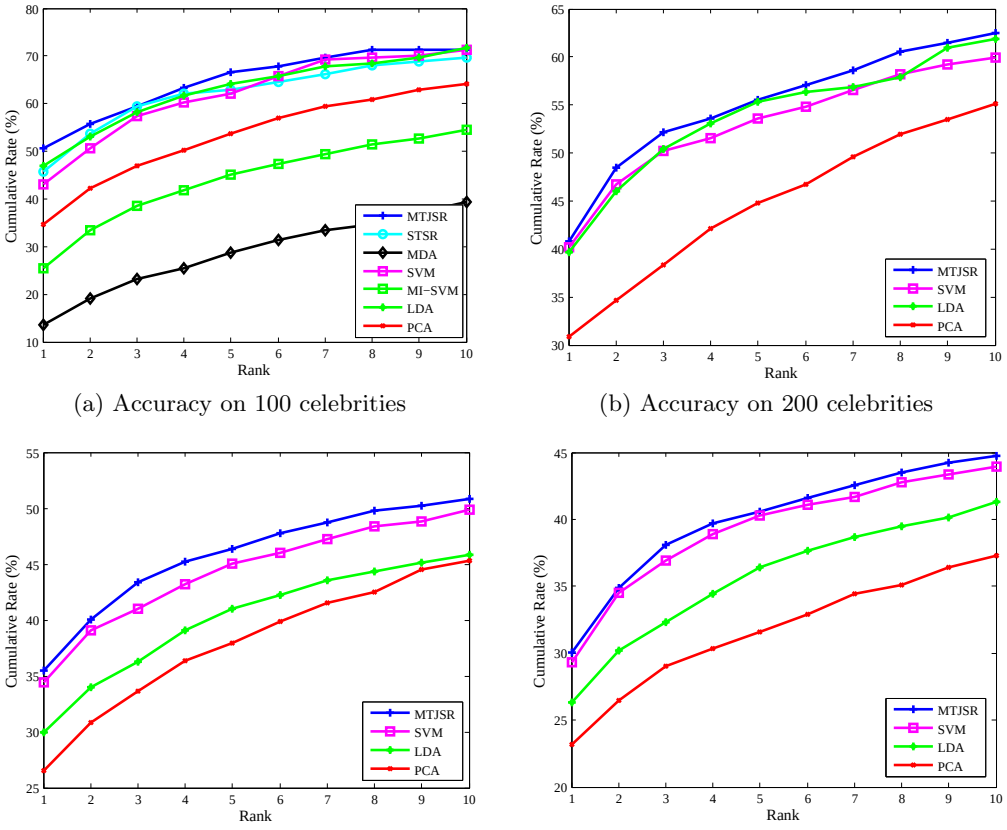
(a) Accuracy on 100 celebrities

(b) Accuracy on 200 celebrities

(c) Accuracy on 400 celebrities

(d) Accuracy on 800 celebrities

Figure 2.5: Face identification accuracy comparison between accelerated MTJSR and other applicable methods in Open-set test. Four different scales of experiments are conducted, with 100, 200, 400, 800 subjects respectively. On scale 100, we compare MTJSR with PCA, LDA, MDA and STSR. On scale 200, 400, 800, either due to memory limit or efficiency issue for other baseline algorithms, we only compare accelerated MTJSR with PCA and LDA.



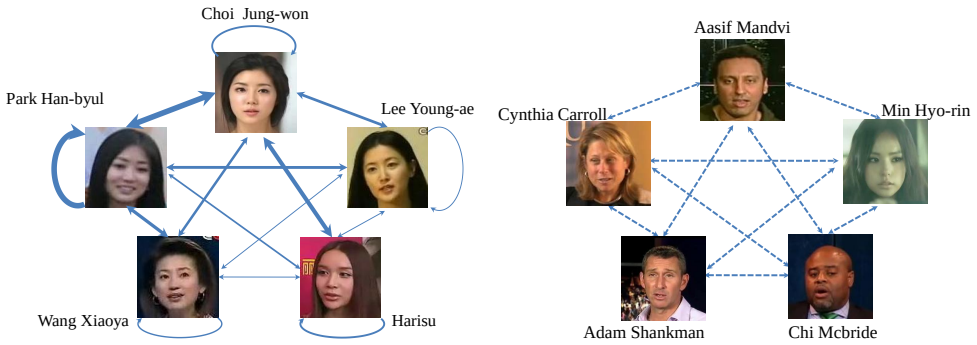
(a) Accuracy on 100 celebrities

(b) Accuracy on 200 celebrities

(c) Accuracy on 500 celebrities

(d) Accuracy on 1000 celebrities

Figure 2.6: Face identification accuracy comparison between accelerated MTJSR and other applicable methods in Close-set test. Four different scales of experiments are conducted, with 100, 200, 500, 1000 subjects respectively. On scale 100, we compare MTJSR with PCA, LDA, MDA, SVM, MI-SVM and STSR. On scale 200, 500, 1000, either due to memory limit or efficiency issue for other baseline algorithms, we only compare accelerated MTJSR with PCA and LDA.



(a) Similar faces (b) Dissimilar faces

Figure 2.7: An illustration of similar and dissimilar faces. The thickness of the edge indicates the probability to mutually classify one subject as another. The dashed line of the dissimilar faces means that their similarities are quite small.

philosophy of MI-SVM is that a bag is labeled positive if at least one instance in the bag is positive, and a bag is negative if all the instances in it are negative. It is unsuitable for the video based face identification scenario, when considering an image sequence as one bag. On scale-100, MTJSR outperforms the STSR, which validates the necessity for collaborative inference instead of simple late confidence fusion.

To better understand which subjects are easy to be mutually mis-classified, we construct a graph based on the predicted rank list from MTJSR algorithm, and the top ranked subjects are assigned larger weights to connect the true subjects. Then, we run the dense subgraph detection algorithm proposed by Liu *et al.* [91, 35, 30] to detect similar faces. Figure 2.7 shows two examples of similar faces and dissimilar faces. The similar faces are within the same dense group and are easy to be mutually mis-classified. It can be seen that within this group, the visual appearances of all subjects are quite similar. The dissimilar faces are from different dense subgraphs and not easy to be mis-classified.

2.5.3 Running Time Evaluation: MTJSR vs. Baselines

We list the average running time and memory usage of MTJSR and other baselines on scale-1000 of Close-set in Table 2.3. The testing time is averaged over the first 100 testing sequences. Here "MTJSR+SISO" means MTJSR algorithms accelerated by

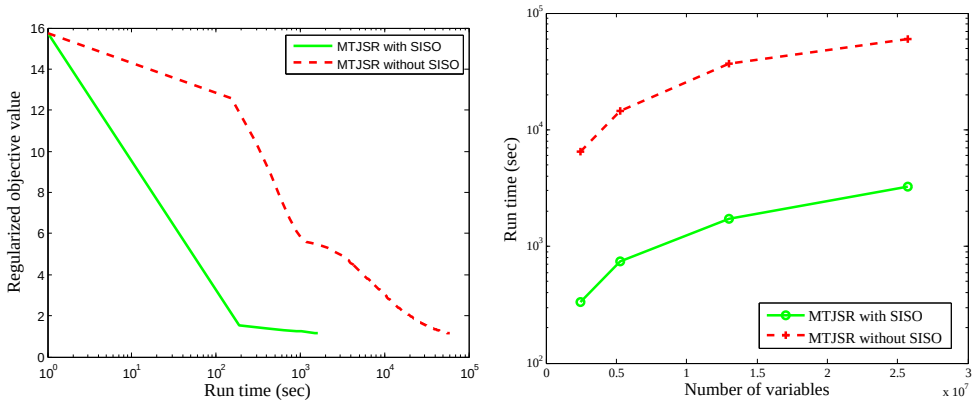


Figure 2.8: The curves for MTJSR with/without SISO are plotted in green solid and red dotted curves, respectively. (a) Objective values as functions of run time. (b) Average run times for different numbers of variables.

SISO, while only "MTJSR" means no SISO speedup. Because MTJSR(+SISO) and STSR are training-free, the training time is 0. More computation cost of MTJSR(+SISO) and STSR exists in the testing stage, so more testing time is needed than the baselines. It can be seen SISO achieves $\sim$ 18 times faster than the ordinary MTJSR witout SISO. It should be noted that our current APG solver to solve the MTJSR and MTJSR+SISO is not the fastest, and the testing time for MTJSR and MTJSR+SISO will be further reduced when a faster solver is applied. MI-SVM and MDA do not scale well in the large scale settings. More than 48 GB memory is used on scale-1000, which is far too more than we can bear. So we do not report the performance of MI-SVM and MDA on scale-1000.

Table 2.3: Training/testing time and memory usage of MTJSR and other baselines on scale-1000 of Close-set.

	Training Time (sec)	Testing Time (sec)	Memory (GB)
MTJSR+SISO	0	3,254	12.2
MTJSR	0	59,505	11.5
STSR	0	40,851	11.0
SVM	130,773	0.1	13.5
LDA	296	115	34.9
MI-SVM	–	–	>48.0
MDA	–	–	>48.0

2.5.4 Speedup Evaluation on Accelerated MTJSR

We evaluate the efficiency of MTJSR with/without SISO. To demonstrate that the efficiency of the accelerated version is better than the original version, we select one test sequence as an example, and plot the value of object function as a function of run time in Figure 2.8a. The original APG algorithm [115] costs about 6×10^4 seconds to reach the optimum, while the accelerated version only needs about 1.6×10^3 seconds.

We further plot the average running time for different numbers of variables in Figure 2.8b. The numbers of variables are counted from scale-100 to scale-1000. The time of the non-accelerated version grows quickly compared with the accelerated version.

2.6 Chapter Summary

In this chapter, we construct a large-population unconstrained video database, Celebrity-1000, from two popular video sharing websites, YouTube and Youku. This dataset is very challenging, which is designed for the large-scale problem and in-the-wild environment. With this large-scale testbed, we dedicate our efforts to boost the efficiency of the Multi-Task Joint Sparse Representation (MTJSR) algorithm. We propose a sparsity-induced scalable optimization method (SISO) to solve the scalability problem in MTJSR. The extensive experiments show the encouraging speedup and also the satisfying performance of the accelerated MTJSR algorithm.

Chapter 3

Deep Aging Face Recognition with Large Gaps

Aging can cause slow but obvious appearance change on human face. It will result in performance drop in most face recognition systems. In this chapter, we use deep learning framework for face recognition with large age gaps [95]. Our method significantly improves the recognition accuracy, while is capable of synthesizing the faces in different age groups as a byproduct at the same time.

3.1 Introduction

With the growing popularity of digital devices, it has been increasingly convenient for people to share photos on various websites, such as Facebook and Flickr. These photos function as a way of connection with other people, and many of them may contain human faces. These considerable number of face images provide rich research material for multimedia studies and benefit many valuable applications, such as face recognition, similar face matching, face naming and face retrieval. To date, many photo sharing websites have been working for quite a long time, and may continue to provide services in an expected long term. The recognition of the user's faces with large age gaps has become a great challenge for most existing applications. When a person grows old, his/her face appearance may change a lot, which makes

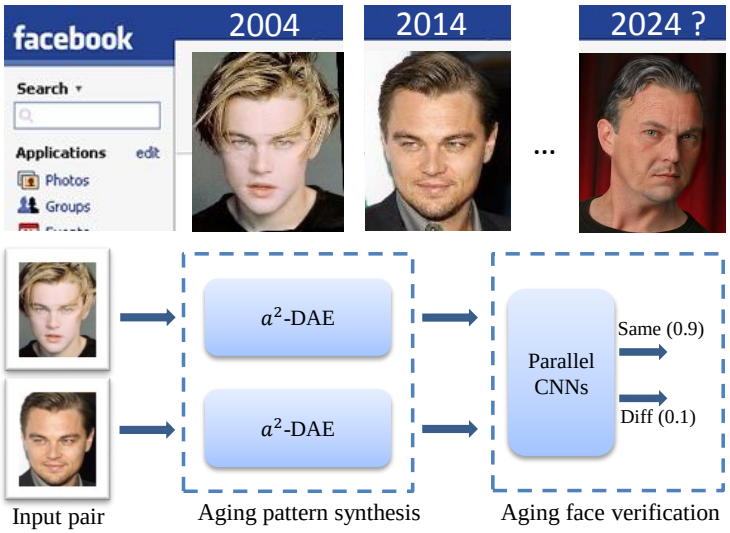


Figure 3.1: Illustration of our two-stage deep learning system for aging face recognition with large gaps. Given a face pair as the input, our system will synthesize the faces of all the age groups, and then verify whether they belong to the same person.

it difficult to correctly recognize his/her photos captured at early ages from the photo album or social network. To address this problem, we build a novel system as shown in Figure 3.1, which can recognize face images across large age gaps. Given a pair of face images from possibly different age groups as the input, our system will synthesize faces of all the age groups for each of them. Then, an aging face verification process is followed to recognize whether the input pair is from the same identity. Since the two steps are both built on deep learning [60, 74], the framework can be called as Deep Aging Face Recognition (DAFR).

The difficulty in developing such a system mainly lies in the facial appearance change during the aging process. Here, the facial appearance includes the facial shape and texture. Facial aging is a very complex process, which involves changes in both the facial shape and texture. As a person grows from young to old, the facial shape will alter as the skull grows, and the facial texture will also show wrinkles gradually. Without taking the facial appearance change across ages into consideration, the performance of current face recognition systems may be degraded.

Despite its practical significance, face recognition with large age gaps is a very challenging problem which has rarely been studied. Till now there has been no sat-

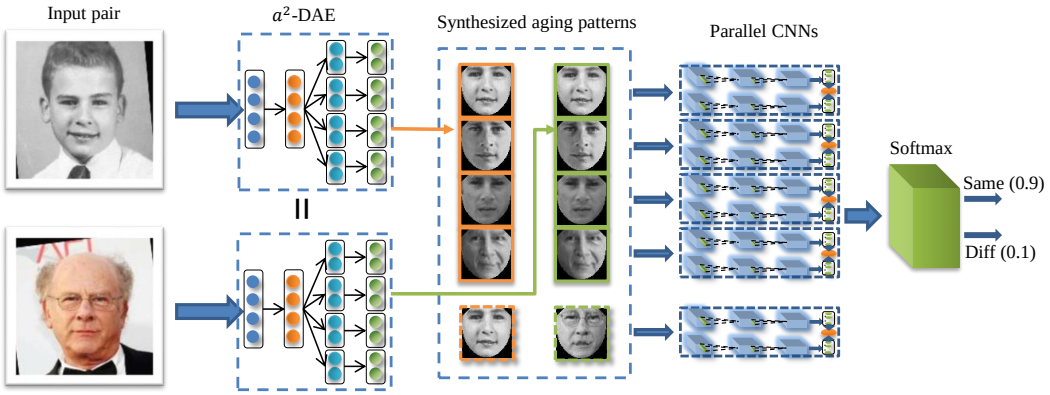


Figure 3.2: The flowchart of the DAFR architecture. It includes two modules: the aging pattern synthesis module and the aging face verification module. In the aging pattern synthesis module, a novel aging-aware DAE (a^2-DAE) is proposed to synthesize the faces of all the age groups. In the aging face verification module, parallel CNNs are trained based on the synthesized faces and the original faces to predict the verification score.

isfactory dataset for the research on this problem. Two widely used databases for cross-age face recognition research, i.e., FG-NET [2] and MORPH [119], both have limitations. FG-NET [2] only contains 82 subjects, which is limited in diversity in terms of human race, gender, living environment, etc. As for MORPH [119], the age gap of the photos from the same person is quite small, which is not suitable to study the long time span aging process. Therefore, a new database which can cover both large human identities and large age gaps is desired in this research area. Besides, existing research work, which is not much, mainly tackles the cross-age face recognition problem from either of two directions. The first group of researchers, such as Li *et al.* [88], focus on designing new features, which possess a high discriminative ability to recognize different people and robustness against age changes. The other group of researchers [116, 153] model the face appearance change caused by face aging, but do not well explore the discriminative features.

Based on these observations, we address the cross-age face recognition problem from two perspectives. Firstly, we build the Cross-Age FacE (CAFE) dataset, which contains 901 male and female celebrities from a variety of races, careers and nationalities. Each person in this dataset has photos across relatively large age gaps, from child, young, adult to old age.

Secondly, we propose a new deep learning architecture, called Deep Aging Face Recognition (DAFR), to model the aging process and then extract strong discriminative features for the verification task. The flowchart of this architecture is illustrated in Figure 3.2. Two modules are included in this architecture: the aging pattern synthesis module and the aging face verification module. For the aging pattern synthesis module, we use a novel deep aging-aware denoising auto-encoder (a^2-DAE) to synthesize the face appearances of all the age groups for the input face of an arbitrary age. Different from the ordinary DAE, the aging-aware a^2-DAE has multiple branches in the decoding layer. The reconstructed face of a certain age group can be output from the corresponding branch. Then the faces for all the age groups are synthesized for an input image pair to the verification module, the following verification can be conducted by directly comparing the faces from the same age group. In this way, the age difference of a face pair is well dismissed. For the aging face verification module, with the synthesized aging patterns of the face pair taken as the input, each pair of synthesized faces from the same age group is fed into a parallel convolutional neural network (CNN) [74, 145] to learn discriminative features and do the face verification. Unlike the traditional methods, the convolutional neural network can directly learn strong discriminative features supervised by the pair labels. In the parallel CNN, a discriminative space is learned where the similarity of the face pair from the same identity is maximized, while that of the pair from different identities is minimized. These parallel CNNs trained based on the synthesized faces of each age group are then jointly fine-tuned to achieve a high discriminative capacity.

However, the DAFR architecture can easily suffer from overfitting in the aging pattern synthesis module. In the aging pattern synthesis module, the objective function values of the a^2-DAEs can be low in the training set, while those are generally higher in the testing set. This will lead to "perfect" reconstruction of the aging patterns in the training set, but will cause large reconstruction errors in the testing set. This mis-match will result in the bad generalization capability of the trained parallel CNNs which are based on the synthesized faces. To avoid this

problem, we train the aging pattern synthesis module in a cross-validation fashion, such that the training and testing reconstruction errors are well balanced. Then, the synthesized aging patterns are fed into the aging face verification module, which have the similar reconstruction errors from the training and testing sets.

The contributions of this work are summarized as follows:

- A large Cross-Age FacE (CAFE) dataset is constructed, including $4,650$ face images of 901 celebrities covering large age gaps, which can serve as a new and comprehensive benchmark for the research community to study the aging face recognition problem.
- The Deep Aging Face Recognition (DAFR) architecture is proposed, including two modules, i.e., aging pattern synthesis module and aging face verification module.
- A novel training strategy is exploited to produce error-aware outputs based on the cross-validation strategy for the aging synthesis module, such that the whole framework can less suffer from overfitting.

3.2 Related Work

There have been very few datasets for the cross-age face recognition research. FG-NET [2] and MORPH [119] databases are the most widely used face databases, which serve as evaluation benchmarks for cross-age face recognition methods [116, 88, 153]. The FG-NET database contains only $1,002$ images of 82 subjects from age 0 to 69. The relatively small size of the database makes it inappropriate for the real applications. The MORPH database contains two subsets: MORPH album 1 and MORPH album 2. MORPH album 1 contains $1,690$ images of 625 subjects, and MORPH album 2 contains $15,204$ images of $4,039$ subjects. However, each subject in the MORPH database only has averagely about three images with a small age gap, which makes it inappropriate for modeling the aging process for aging face recognition with large age gaps.

The research literature on cross-age face recognition is also quite limited over

the past decades. Geng *et al.* [44] modeled the face aging patterns. The face aging pattern is defined as a sequence of face images from the same person sorted in the time order. A principal component space of aging pattens is constructed to model the correlation of faces from different age groups. The faces at different ages of the testing face can be reconstructed by projecting the testing face into the subspace. Park *et al.* [116] proposed a 3D aging model, which can capture the aging pattern in the 3D domain. They first converted 2D images into 3D ones by a 3D morphable model [15], and then the facial shape and texture changes are modeled separately in the Principal Component Analysis (PCA) [70] subspace. The missing samples in the training set will be generated by interpolating from the samples of the nearest ages. Wu *et al.* [153] used a parametric craniofacial growth model to model the facial shape change. These methods can model the aging process of the face shape or texture, but are weak in the discriminative capacity. Li *et al.* [88] proposed a discriminative model for age-invariant face recognition. They used scale invariant feature transform (SIFT) and multi-scale local binary patterns (MLBP) as local descriptors. To avoid overfitting, multi-feature discriminant analysis (MFDA) was proposed to process the two local feature spaces in a unified framework. It focused on highly discriminative features but failed to model the aging process. With the help of deep learning methods, our proposed framework can not only model and synthesize the aging process, but also learn discriminative features to achieve high performance.

There have been many works exploiting deep learning technology for face analysis/recognition problem. Based on deep belief networks, Luo *et al.* [106] propose a novel face parse, which can hierarchically parse faces into parts, components and pixel-wise labels. Taigman *et al.* [134] and Sun *et al.* [130, 128] use convolutional neural networks (CNN) [74] based methods for face verification problem. The performance of their works already reaches or surpasses human's performance on the widely used labeled face in the wild (LFW) dataset [66]. Zhu *et al.* [172] propose a novel multi-view perception network (MVP), which can reconstruct a full spectrum of views based on a single 2D face. However, all these works have not taken cross-age

face recognition problem into consideration, which is the main problem I want to handle in our work.

3.3 The Cross-Age FacE (CAFE) Dataset

The Cross-Age FacE (CAFE) dataset is constructed with photos of 901 celebrities. We first collect a list of celebrities' names and crawl images from the Internet. These celebrities include actors, singers and politicians. The genders are roughly balanced. Then face bounding boxes and 68 landmark points in the faces are detected and located by a commercial facial analysis toolbox [1]. Faces are aligned by similarity transform according to the centers of two eyes and that of the mouth. The distance of two eye centers is set to 32 pixels. The photos are cropped with enlarged face bounding boxes of size 160×160, and then saved. The images which contain non-frontal faces are removed, and the remaining photos have only near frontal faces. we finally collect $4,659$ photos of 901 celebrities. Based on the photos' taken date stored in the metadata and the celebrities' years of birth, we divide the photos into four age groups: child (0$\sim$12 years old), young (13$\sim$25 years old), adult (26$\sim$50 years old) and old age (>50 years old). Some celebrities have photos of all the four age groups, while some have photos of two or three age groups. Our dataset contains more subjects than the FG-NET dataset and has much larger age gaps for each subject than the MORPH dataset. Some exemplar faces of the celebrities from the CAFE dataset are shown in Figure 3.3.

3.4 Deep Aging Face Recognition

3.4.1 Framework Overview

Our proposed whole framework for cross-age face recognition includes the following steps:

- **Preprocessing: shape and texture separation**. Faces are preprocessed to extract shape and shape-free texture, as described in Section 3.4.2.

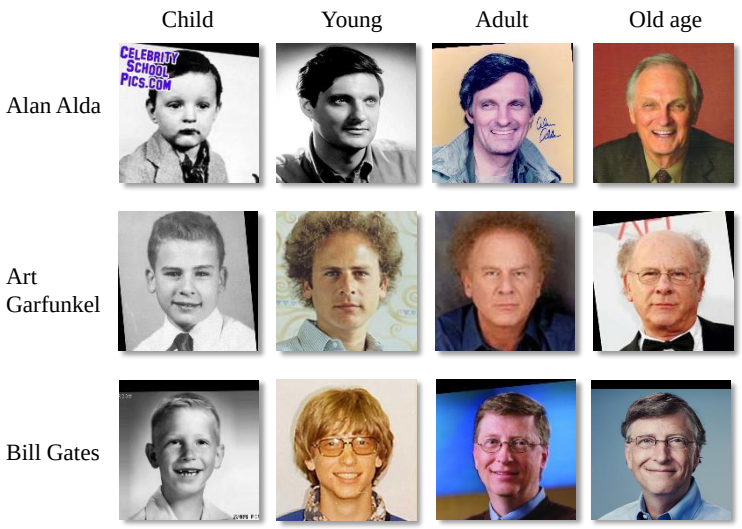


Figure 3.3: Some exemplar faces in the CAFE dataset. From top to bottom, the celebrities are Alan Alda, Art Garfunkel and Bill Gates. The photos are shown in four age groups: child, young, adult and old age, from the first column to the last column.

- **Aging pattern synthesis module**. The deep aging-aware denoising auto-encoder (a^2-DAE) is learned to synthesize the faces at all the age groups for the input face. The details will be described in Section 3.4.3.

- **Aging face verification module**. Given aging pattern pair as input, the parallel convolutional neural network is exploited to learn a discriminative space for the verification task.

3.4.2 Preprocessing: Shape & Texture Separation

Both shape and texture of a face contain important information about human age and identity. The cranial size of a face increases quickly as a person grows until 19 years old. After that, the facial texture change becomes the dominant factor for human aging [5]. Wrinkles are deepened at the sides of the eyes, and freckles and aging spots occur on the face skin. However, shape and texture correlate with each other deeply on the face, and are also influenced by other factors, such as pose and illumination. This phenomenon makes cross-age face recognition an even more challenging problem.

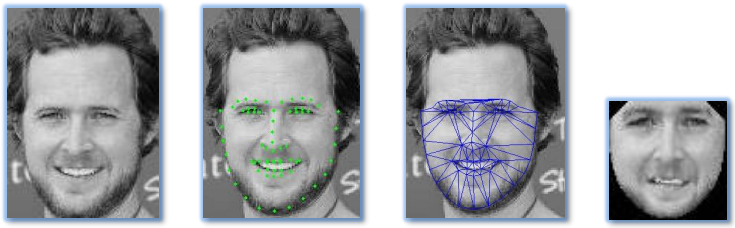


Figure 3.4: An example of the shape-free texture extraction process: a) the original face, b) the detected face landmark points, c) the Delaunay triangulation, and d) the warped face.

Based on the above observations, we extract shape and texture from the faces and model them separately. 68 face landmark points are located by the OMRON face alignment algorithm [1]. Faces are aligned according to the centers of two eyes and that of the mouth. The shape information is represented by the normalized coordinates of landmark points on the aligned faces.

To extract shape-free faces, we first calculate the mean shape of all the training images from the dataset. Delaunay triangulation [29] is computed on the mean face and each face image in the dataset to obtain 111 triangles. Piecewise linear affine transformation [49] is applied within the corresponding triangles between the face image and the mean face to obtain the warped face. Figure 3.4 shows the shape-free texture extraction process. Figure 3.4(a) is the original face image. Figure 3.4(b) shows the detected 68 landmark points. The extracted 111 triangles are shown in Figure 3.4(c). Figure 3.4(d) illustrates the warped face after piecewise linear affine transformation, which represents the shape-free facial appearance.

3.4.3 Aging Pattern Synthesis Module

Motivations

Facial appearance (shape and texture) changes dramatically along with the human aging process, which poses a great challenge to current face recognition systems. For example, if we directly compare two face images of the same person, one for the childhood and the other for the adult, due to the changes in face shape and texture over such a large time span caused by the environment, genes, and other social

factors, the similarity between the two faces in the feature space may be low. Most current face recognition systems may fail in such a case. Thus, modeling the face appearance change over the time is a necessary step for cross-age face recognition.

Unlike other factors such as gender or facial expression, face aging has its own characteristics. First of all, human aging is personalized. One's face appearance is determined by mainly two aspects: internal factors, i.e. genes, and external factors, such as one's living environment, lifestyle, etc. Genes determine the initial appearance of a person. As the person grows up, many external factors may impose their influence on what he/she looks like. For example, a man who has an unhealthy diet tends to have a fat face. Secondly, face aging is an irreversible sequential process. Every person, if no deathly accident or disease occurs, experiences the growing process from childhood, youth to adult and old age, in a temporal order. No one can go through the process the other way around. It is slow with decades of time, but irreversible.

Based on the characteristics, the face appearance change of each subject should be considered as a function of both identity and age. Each image I in the cross-age face dataset should have two labels: the identity label $id(I)$ and the age label $age(I)$. This is the difference between the ordinary face recognition problem and the cross-age face recognition problem. For the ordinary face verification problem, given two input faces I_a and I_b, the system verifies whether $id(I_a)$ equals $id(I_b)$. No age information is considered in the ordinary face system. In the cross-age face verification system, given two input faces I_a at $age(I_a)$ and I_b at $age(I_b)$, the system verifies whether $id(I_a)$ equals $id(I_b)$, but does not require $age(I_a) = age(I_b)$. If we can map I_a and I_b to the same age group as $\hat{I}_a$ and $\hat{I}_b$ where $age(\hat{I}_a) = age(\hat{I}_b)$, and directly verify whether $id(\hat{I}_a)$ equals $id(\hat{I}_b)$, the face recognition problem will be much easier.

We follow Geng *et al.* [44] to represent the face appearance change of the same person over the time as the *aging pattern*. An aging pattern is defined a sequence of face images of the same identity sorted in the temporal order [44]. The aging pattern is personalized and ordered by time. Here, we consider four time spans: child, young,

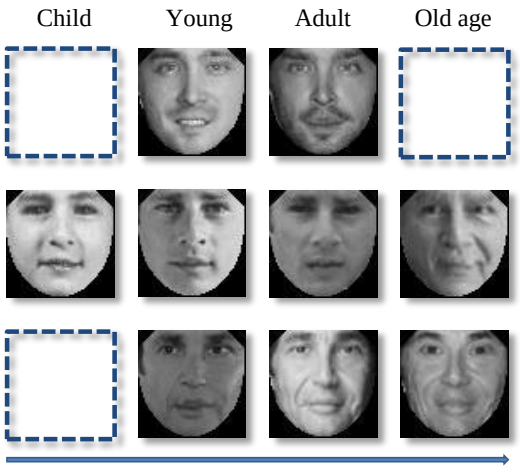


Figure 3.5: Examples of the aging pattern. Each row from left to right is the aging pattern sorted in the time order. The blank bounding box means the missing position in the aging pattern.

adult and old age. Figure 3.5 shows some examples of aging patterns. The aging pattern of one person is shown in one row. From the first to the last column, the aging pattern is sorted in the time order. Note that some positions in the aging patterns of the training samples are missing, so the aging model should have the ability to handle the missing aging patterns. In the testing stage, the aging pattern of the testing sample will be synthesized. Based on the synthesized aging pattern, the testing pair I_a and I_b are projected to $\hat{I}_a$ and $\hat{I}_b$, where $age(\hat{I}_a) = age(\hat{I}_b)$. Thus, the facial appearance change caused by aging is well eliminated.

The Deep Aging-aware Denoising Auto-encoder

We propose the deep aging-aware denoising auto-encoder (a^2-DAE) to learn the aging model. Using the aging patten of each face image as groundtruth, the a^2-DAE is trained in a supervised way. Given a testing image as the input, the aging model will predict the aging pattern of the testing image. We first review some basic concepts of the auto-encoder, and then go ahead to our formulation of the a^2-DAE.

Auto-encoder and Denoising Auto-encoder Given an image as the input, in which the pixel values are considered as the visible variables $\mathbf{v}$, the auto-encoder [13] first encodes it into a hidden representation $\mathbf{h}$ via a deterministic mapping $\mathbf{h} = \sigma(\mathbf{W}\mathbf{v} + \mathbf{b})$, where σ is the activation function, such as the *sigmoid* function. The

hidden representation $\mathbf{h}$ is then decoded back into $\mathbf{v}'$, the prediction of $\mathbf{v}$, through a similar mapping function $\mathbf{v}' = \sigma(\mathbf{W}'\mathbf{h} + \mathbf{b}')$. If the reverse weight parameter $\mathbf{W}'$ is constrained to $\mathbf{W}' = \mathbf{W}^{\mathbf{T}}$, then $\mathbf{W}'$ is called the *tied weight*. The reconstruction error can be measured in many ways, such as cross-entropy and squared loss. We use squared loss in our later formulation as the loss function, where $L(\mathbf{v}, \mathbf{v}') = \|\mathbf{v} - \mathbf{v}'\|^2$. The hidden representation $\mathbf{h}$ is viewed as a lossy compressed code of $\mathbf{v}$. If there is only one linear hidden layer and the squared loss is used as the cost function, the $k_{\mathbf{h}}$ hidden units of $\mathbf{h}$ can be viewed as the first $k_{\mathbf{h}}$ principal components of the training data. Then the auto-encoder can be viewed the same as PCA. When the nonlinear activation function σ, such as *sigmoid* or *tanh* is used, the auto-encoder will behave differently from PCA. To force the hidden layer $\mathbf{h}$ recovering more robust features and prevent it simply overfitting to the training data, the input data can be stochastically corrupted by noises. This corrupted version of auto-encoder is called **Denoising Auto-encoder**.

Stacked Denoising Auto-encoder The denoising auto-encoder can be stacked into multiple layers to form the stacked denoising auto-encoder. This network should firstly be pretrained layer by layer in an unsupervised manner. After the k-th layer is pretrained, then taking the hidden representation $\mathbf{h}_k$ of the k-th layer as the input, the $(k+1)$-th layer will be pretrained. The pretraining can be conducted by the auto-encoder or Restricted Boltzmann Machine (RBM) [13]. After the pretraining stage, a supervised fine-tuning process can be conducted to jointly train the whole model from the first layer to the last layer.

The deep aging-aware denoising auto-encoder (a^2-DAE) is shown in Figure 3.6. Denote the input image in the first layer as $\mathbf{v}_0 \in \mathbb{R}^d$ (or $\mathbf{h}_0$ for convenience) and the aging pattern as $\{\mathbf{v}_i \in \mathbb{R}^d | i = 1 \cdots 4\}$. The a^2-DAE aims to learn a mapping function $f(\mathbf{v}_0)$ to give the reconstruction of the aging pattern as $\{\mathbf{v}'_i \in \mathbb{R}^d | i = 1 \cdots 4\}$. $d = 64 \times 64 = 4,096$ is the dimension of the training and testing images. The

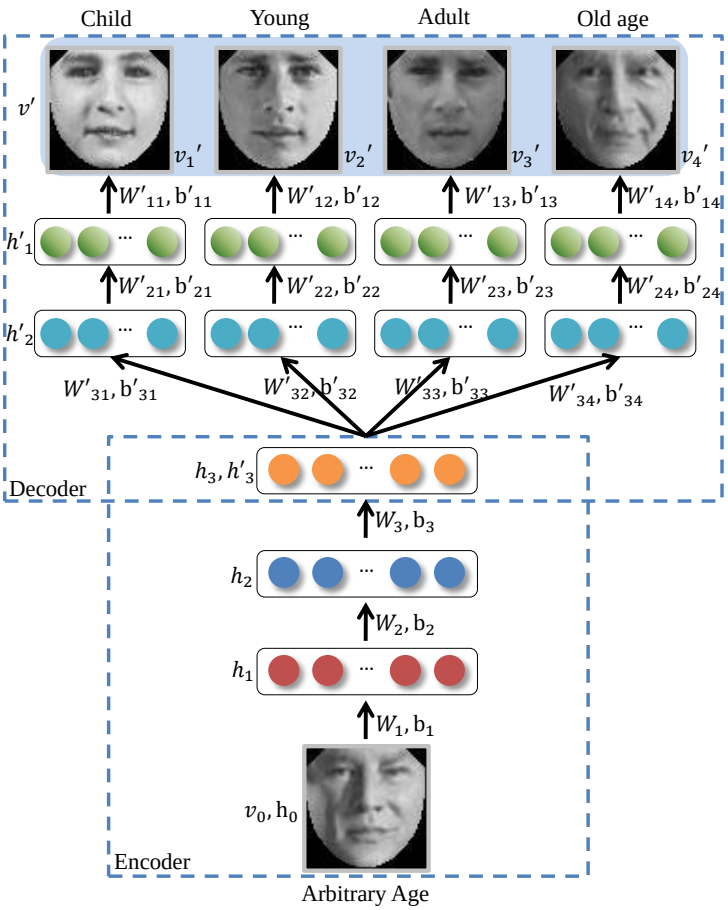


Figure 3.6: The deep aging-aware denoising auto-encoder (a^2-DAE).

mapping function $f(\mathbf{v}_0)$ can be decomposed as follows:

$$\begin{aligned}
\mathbf{h}_i =& \mathbf{W}_i\sigma(\mathbf{h}_{i-1}) + \mathbf{b}_i,\ i = 1, 2, 3, \\
\mathbf{h}'_{kj} =& \mathbf{W}'_{k+1,j}\sigma(\mathbf{h}'_{k+1}) + \mathbf{b}'_{k+1,j},\ k = 2, 1, j = 1 \cdots 4, \\
\mathbf{v}'_j =& \mathbf{W}'_{1j}\sigma(\mathbf{h}'_{1j}) + \mathbf{b}'_{1j},\ j = 1 \cdots 4.
\end{aligned} \tag{3.1}$$

We use the *sigmoid* function as the activation function $\sigma(h) = (1 + \exp(-h))^{-1}$. $\mathbf{W}$ and $\mathbf{b}$ are the weight matrices and bias vectors. $\mathbf{h}_i, i = 1, 2, 3$ is the hidden representation of the input data. The numbers of hidden units of $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$ are $2,500$, $1,000$ and 400, respectively. $\mathbf{h}'_{kj},\ k = 2, 1, j = 1 \cdots 4$ is the reconstructed hidden representation $\mathbf{h}_k$ at position j. $\mathbf{h}'_{kj}$ has the same number of hidden units with $\mathbf{h}_k$. $\mathbf{v}'_j$ is the reconstructed aging pattern at position j. $\mathbf{h}_3$ is a shared representation across all the modalities, which constructs an age-invariant space for all the input

images with different age labels.

To train the a^2-DAE, we minimize the square error loss function between the groundtruth and the reconstruction of the aging pattern:

$$\min_{\mathbf{W},\mathbf{b}} L(\mathbf{v}, \mathbf{v}') = \sum_{i=1}^{4} \|\mathbf{v}_i - \mathbf{v'}_i\|^2 + \epsilon_1 \|\mathbf{W}\|^2, \tag{3.2}$$

where ϵ_1 is the ℓ_2 weight decay coefficient for all the layers. Unfortunately, some images in one or more positions of the aging patterns are missing, thus we cannot minimize Eqn. 3.2 directly. However, the missing images in the aging patterns follow some statistical principles. For example, the missing image in the position "child" of the aging pattern should reflect the common traits of children. The children's skin is more smooth than that of the old people. It is very rare for children to have wrinkles on faces like the old people. Based on these statistical principles, the loss function can be further defined as follows:

$$\min_{\mathbf{W},\mathbf{b}} L(\mathbf{v}, \mathbf{v}') = \sum_{i=1}^{4} \Phi(\mathbf{v}_i, \mathbf{v'}_i) + \epsilon_1 \|\mathbf{W}\|^2, \tag{3.3}$$

where $\Phi(\mathbf{v}_i, \mathbf{v'}_i)$ equals

$$\begin{cases} \|\mathbf{v}_i - \mathbf{v'}_i\|^2, & \mathbf{v}_i \neq \varnothing, \\ \sigma^2 \mathbf{v'}_i^T \mathbf{P}_i \mathbf{\Lambda}^{-1} \mathbf{P}_i^T \mathbf{v'}_i + \frac{\sigma^2}{\sigma_1^2} \|\mathbf{v'}_i - \mathbf{P}_i \mathbf{P}_i^T \mathbf{v'}_i\|^2, & \mathbf{v}_i = \varnothing. \end{cases}$$

When the target image $\mathbf{v}_i$ at position i of the aging pattern is not empty, we still use the square error loss function. When the target image $\mathbf{v}_i$ at position i is empty, we enforce the reconstructed $\mathbf{v'}_i$ to keep the common characteristics at age group i. $\mathbf{P}_i \in \mathbb{R}^{d \times m}$ is the projection matrix calculated from the training data of group i by PCA [70]. m is the dimension of the projected subspace after PCA. m is set to $1,000$ to keep $\sim 95\%$ energy. We assume that the training images have already been normalized to zero mean and unit variance. $\mathbf{\Lambda} = diag[\lambda_1; \lambda_2; \cdots; \lambda_m]$ where $\lambda_1, \lambda_2, \cdots, \lambda_m$ are the m largest eigen values. $\Phi(\mathbf{v}_i, \mathbf{v'}_i)$ for $\mathbf{v}_i = \varnothing$ is inferred from probabilistic principal component analysis (PPCA) [135] and minimizing the

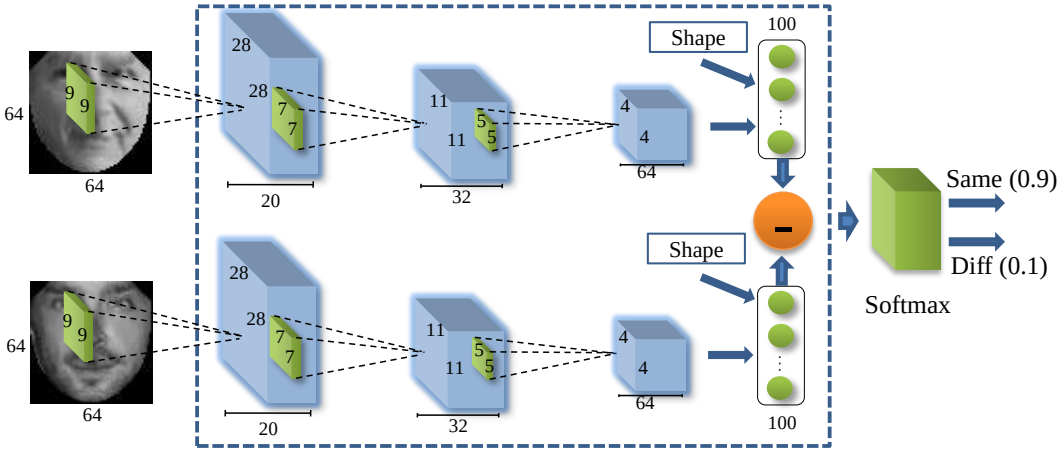


Figure 3.7: The architecture of the parallel CNN. It takes a face pair as input, and predicts whether this pair belonging to the same person.

weighted sum of these two terms is equivalent to maximizing the probability of the synthesized face in the corresponding age group.

3.4.4 Aging Face Verification Module

Face verification aims to distinguish whether a face pair has the same identity [66]. After the aging pattern synthesis module, we can obtain the synthesized faces of the input face in all the age groups (child, young, adult and old age). For each pair of reconstructed faces of a certain age group, we train a parallel CNN, which takes a face pair as the input, for the verification task. Because there are totally four age groups, we use the four synthesized face pairs and one original face pair to train five CNNs. The final verification score is obtained by fusing all the CNNs. There are two main reasons for that the original face pair is also used to train CNN. Firstly, the CNN trained from the original face pairs can still learn some strong discriminative aging-invariant features. Secondly, the synthesized faces may contain some reconstruction errors inevitably, which may be imperfect for the verification purpose. However, if the CNNs trained on reconstructed face pairs combine with that trained on the original face pairs, the complementarity among these CNNs will result in enhanced performance. Because of the synthesized faces of all the age groups, the problem of calculating the similarity of I_a and I_b has been changed to computing the similarity of $\hat{I}_a$ and $\hat{I}_b$ when $age(\hat{I}_a) = age(\hat{I}_b)$.

As shown in Figure 3.7, the parallel CNN we use takes an image pair $(\hat{I}_a, \hat{I}_b)$ as the input and outputs the similarity score $s(\hat{I}_a, \hat{I}_b)$ of this pair:

$$s(\hat{I}_a, \hat{I}_b) = softmax(\mathbf{W}_s|o(\hat{I}_a) - o(\hat{I}_b)| + \mathbf{b}_s), \tag{3.4}$$

where $o(\cdot)$ is the output of the fully-connected layer and $|\cdot|$ is element-wise absolute value. $\mathbf{W}_s$ and $\mathbf{b}_s$ are the learnable parameters in the softmax layer. $s(\hat{I}_a, \hat{I}_b)$ is the estimated probability of $(\hat{I}_a, \hat{I}_b)$ belonging the same person or different person. The goal is that $o(\cdot)$ can keep the best discriminative ability by mapping the input data into a semantic space. So, in that space, the similarity score of image pairs from the same person should be large and that from different persons should be small.

The parallel CNN structure has nine layers and is trained by stochastic gradient descent. The input layer takes a pair of images as the input. The next three convolutional layers are followed by max-pooling layers to extract discriminative image features hierarchically. A nolinear activation function is followed after the convolution operation is conducted on the input data. Here we use the rectified linear function (ReLu) [107] as the activation function, which is defined as $\max(0, \cdot)$. Compared with *tanh* and *sigmoid* units, ReLu converges faster and does not easily suffer from the saturation problem [74]. The followed max-pooling layer has pooling stride of 2×2 pixels. The following fully connected layer is learned as a semantic space, where the similarity score of image pairs from the same person is enlarged, while that from different persons is reduced. Besides the convolutional features extracted from the input face texture, the normalized coordinates of the 68 landmarks are combined as a 132 dimension vector, which is also incorporated to learn the discriminative space. The last layer is a softmax layer to produce the similarity score of the input image pair. The parameters from the input layer to the fully connection layer are shared between the input image pair.

3.4.5 Training the Whole Framework

The training process of the whole framework is summarized in Algorithm 2 and described as follows.

Algorithm 2 Training the whole framework.

Inputs: The face images and image pairs in the Training set.
Outputs: The learned parameters of the a^2-DAEs and the CNNs.
Train the a^2-DAEs:
Pretrain the encoding layers of a^2-DAE with M_0^b.
for $i = 1 \rightarrow 4$ **do**
    Pretrain the decoding layers on the i-th branch of a^2-DAE with M_i^b.
end for
Train the a^2-DAE model M_0^a with the whole training set.
for $t = 1 \rightarrow T$ **do**
    Train the t-th a^2-DAE model M_t^a.
    Obtain the synthesized aging patterns on the t-th training subset.
end for
Obtain the synthesized aging patterns of all the training subsets.
Train the parallel CNNs:
Train the CNN M_0^s on the original images.
for $i = 1 \rightarrow 4$ **do**
    Train the i-th CNN M_i^s on the constructed faces of the i-th age group.
end for
Jointly fine-tune the CNNs $\{M_i^s | i = 0 \cdots 4\}$ to obtain the CNN M^s.

We employ deep belief networks (DBN) [60] to pretrain the parameters in the a^2-DAEs. DBN is stacked by RBMs as each layer and trained layerwisely [60]. The ordinary RBM models only use binary units for the visible layers, so they are not suitable for the natural images which have real pixel values. To handle this problem, the binary visible units are replaced by linear units with Gaussian noise. We totally train five DBN models $\{M_i^b | i = 0 \cdots 4\}$ and use the trained parameters as the initialization of the a^2-DAEs. M_0^b is trained using all the training data, and the trained parameters are used as the initialization of the encoding layers of a^2-DAE. The a^2-DAE has four branches in the decoding layers, each of which reconstructs the faces in the specific age group. So we use $\{M_i^b | i = 1 \cdots 4\}$ to pretrain the i-th branch in the decoding layers of the a^2-DAE from the training data in the i-th age group. The trained parameters in the decoding layers of $\{M_i^b | i = 1 \cdots 4\}$ are used as the initialization in the decoding layers of a^2-DAE. In all the DBN models, we

use *tied weights*, which means the weights in the decoding layers are the transpose of the corresponding weights in the encoding layers.

After pretraining the a^2-DAE to obtain a good initialization, we fine-tune it with the training images and their corresponding aging pattens. The detailed network structure and training process are described in Section 3.4.3. To make the later parallel CNNs based on the reconstructed faces correctly trained, the overfitting of a^2-DAEs on the training set should be handled. Because the values of the loss function of a^2-DAEs on the training set can be low and those of the testing set are generally higher, the training images can "perfectly" reconstruct the aging patterns but the reconstructed aging patterns in the testing set will have relatively large reconstruction errors. This mis-match of training and testing images will result in the consequence that the later trained CNNs based on the reconstructed faces are poorly trained. To handle this problem, we train the a^2-DAEs in a cross validation way. we first train the a^2-DAE M_0^a with all the training images. Then we divide the training set into T non-overlap subsets. T is set to 6 in our experiments. M_0^a is fine-tuned with data from the training set but the i-th subset, to obtain the $\{M_i^a | i = 1 \cdots T\}$ a^2-DAE. Then the aging patterns of the i-th subset are constructed from model $\{M_i^a | i = 1 \cdots T\}$. In this way, we can obtain synthesized aging patterns of all the training images, during which process overfitting is well controlled. The testing synthesized faces can be constructed from the a^2-DAE M_0^a.

We use the original faces and the reconstructed aging patterns to train the parallel CNNs and fine-tune them jointly. The reason of using the original face is explained in Section 3.4.4. Based on the image pairs of the original faces in the training set, we first train the CNN M_0^s. Then we use the reconstructed faces from the training pairs at the i-th age group to train the CNN $\{M_i^s | i = 1 \cdots 4\}$. Then these five CNNs $\{M_i^s | i = 0 \cdots 4\}$ are combined before the softmax layer. The softmax layer from each CNN is removed, and then a new softmax layer is added to combine all these CNN models. After that, the combined CNNs are fine-tuned to obtain the CNN M_f^s, which will give the final verification score.

3.5 Experiments

We evaluate the performance of our proposed framework and other baselines for aging face recognition on our CAFE dataset. We first visualize some learned intermediate parameters. Then, we show the synthesized faces from the a^2-DAE. After that, we report the performance of face verification by quantitative evaluations. 5-fold cross validation is used to train the models and the performance is reported. For each fold, we randomly select 600 celebrities and generate $14,000$ pairs ($7,000$ pairs with same identities and $7,000$ pairs with different identities) as the training set. We generate $2,000$ pairs ($1,000$ same and $1,000$ different pairs) from the other 301 celebrities for testing. The performance is reported in verification accuracy and plotted in receiver operating characteristic (ROC) curves. All the deep learning based experiments are conducted on a server of 8 CPU cores and 32 GB physical memory. It is equipped with a GTX TITAN GPU of $2,688$ CUDA cores and 6 GB GPU memory. The deep learning library we use is Pylearn2 [47], which is a python-based machine learning library and built on Theano [14].

In the aging pattern synthesis module, ϵ_1 is set to 0.0001. The initial learning rate is set to 0.1 and the initial momentum is 0.5. Batch size is set to 100. Around 2.5 hours are used to train each a^2-DAE of 500 epochs.

In the aging Face verification module, the batch size is set to 200. The initial learning rate is set to 0.01 and the initial momentum is 0.5. The number of kernels for each convolution layers are 20, 32 and 64. The kernel sizes are 9×9, 7×7 and 5×5. The fully connection layer has 100 neurons. The weight decay in this layer is set to 0.001. Dropout [61] probability is set to 0.2. Around 4 hours to train each parallel CNN of 100 epochs.

3.5.1 Visualize the Learned Parameters

We visualize the learned parameter W in the first layer of a^2-DAE. Each hidden unit is fully connected with the visible units, and the parameter between them can be plotted as a face-like filter. Since there are totally $2,500$ hidden units in the first

Figure 3.8: The learned filters in the first layer of a^2-DAE. Some parts in the filters are emphasized to capture discriminative information on the input faces.

layer, it is hard to plot all these parameters. We randomly select 30 of them and plot in Figure 3.8. It can be seen that the plotted learned parameters are "ghost"-like faces. Some parts in the filters are emphasized, such as the corner of eyes, where the filtered faces will have larger responses. Unlike Eigen-faces [11] learned from PCA, the filters learned by a^2-DAE can perform more complex nonlinear transformations.

3.5.2 Synthesis Results from a^2-DAE

We visualize the reconstructed aging patterns of the testing images in Figure 3.9. We plot six examples of aging patterns in three rows and two columns. In each example, the first image is the input testing image and the remaining four images are the synthesized aging patterns containing four age groups: child, young, adult and old age. The synthesized aging patterns are predicted from the trained a^2-DAE as mentioned in the previous sections. It can be seen that the faces from all the four age groups are well synthesized. The synthesized faces look very similar to the original faces, but in different age groups. The skin of the faces in the "child" group is very smooth and has no wrinkles. From "child" to "old age", the faces become less smooth, and have more wrinkles and freckles. In the "adult" and "old age" groups, some faces even have beard and mustache.

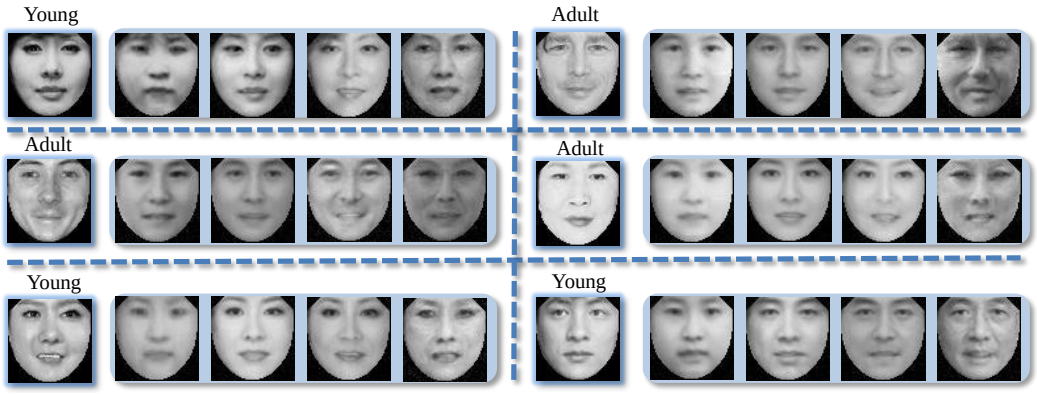


Figure 3.9: The synthesized aging patterns. The first face of each group is the input face, and other four faces are synthesized faces in four age ranges. The age labels of the input faces are labeled above them.

3.5.3 Quantitative Evaluation

In this subsection, We evaluate the performance of face verification. Given a pair of images, the goal is to verify whether the two images belong to the same person or not. The performance is reported as the verification accuracy and plotted on ROC curves. We compare our method with current state-of-the-art features and classifiers for general face recognition, such as high dimensional local binary feature (HDLBP) [21] with probabilistic linear discriminant analysis (PLDA) [87] and probabilistic elastic matching (PEM) [86]. We also compare our method with the aging pattern subspace (AGES) [44] method, which is designed specifically for cross-age face recognition. We take AGES as our cross-age baseline method, because only AGES has the ability to synthesize the faces at other age groups in the testing set, which is most similar with ours. Though the works of Suo *et al.* [131, 132] can also synthesize the aging faces, their focus is only aging synthesis, and aging face recognition is not considered. So, their target is different from ours. The performances of our method and the baselines are evaluated on our CAFE dataset. The photos from each subject in the MORPH dataset have too small age gaps, which are not suitable to train the a^2-DAE model. The FG-NET dataset has too few subjects and images, which is not applicable for our DAFR architecture. Like other deep learning based methods, the DAFR architecture requires more samples in the training process to learn discriminative features and robust classifiers.

HDLBP High dimensional local binary pattern (HDLBP) [21] is used as the feature and followed by probabilistic linear discriminant analysis (PLDA) [87]. Similar to the way of extracting HDLBP features in [21], We extract image patches at 27 main landmark points at four scales of image sizes: 200, 160, 96, 64 from the original images of size 160×160. The patch size is set to 20×20. Each patch is divided into 2×2 cells and LBP [4] histogram is calculated in each cell. The total dimension of features is $25,056$. PCA [70] is used to reduce the feature dimension to 600 to maintain $\sim 90\%$ energy, and then PLDA is used to learn a 64 dimension latent identity space, where the similarity metric of features is calculated.

PEM In the probabilistic elastic matching (PEM) [86] method, we first crop out the center region of the image at the size 96×96. SIFT features are extracted over 3-scale image pyramid with scaling factor 0.9, from sliding window of 8×8 with 4-pixel spacing. UBM-GMM of $1,024$ mixture Gaussian clusters is trained.

AGES In this method, aging pattern subspace (AGES) [44] is learned. The testing image is projected into the aging pattern subspace, and reprojected back to get the reconstructed faces at another age. Similar to our face preprocessing method, the face images are mapped into the mean face to separate shape and texture. The face size is set to 64×64.

Comparison of DAFR and other baselines

The comparison results of our method DAFR and other baselines are shown in Table 3.1. From Table 3.1, it can be seen that for each fold, our method DAFR reaches the highest performance. For example, in Fold 1, our method reaches the accuracy of 0.7895, which is 2.45% higher than HDLBP. High dimensional LBP is one of the best hand-craft features, which has a strong discriminative capacity and shows the state-of-the-art performance in Labeled Face in the Wild (LFW) benchmark [66]. The probabilistic elastic matching (PEM) gives worse performance than our method and HDLBP. Though PEM represents each face image as a bag of spatial-appearance features, which is robust to mis-alignment, the lack of strong prior information about the precise landmark position makes it hard to achieve the

good performance as our method and HDLBP. The AGES method has the worst performance among all the methods. This is because AGES only uses the pixel intensity as feature representation, which has a weak discriminative capacity. The similarity between the image pair is computed by the Mahalanobis distance, and no strong supervised classifiers are used for classification. Compared with it, CNN can not only extract strong discriminative features by the convolution operation in the convolutional layers for each input image, but also jointly optimize feature extraction and classification to achieve the optimal performance.

Table 3.1: The verification accuracies of DAFR and the baselines. "Avg" means average accuracy across all five folds.

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg
HDLBP	0.7650	0.7500	0.7540	0.7450	0.7560	0.7540
PEM	0.7105	0.7090	0.7165	0.6950	0.7215	0.7105
AGES	0.5735	0.5670	0.5400	0.5610	0.5480	0.5579
DAFR	0.7895	0.7790	0.7530	0.7680	0.7800	0.7739

The performance of CNNs based on the synthesized faces

We show the performance of each CNN based on the synthesized faces of all the five folds in Table 3.2 and Table 3.3. The a^2-DAE used in Table 3.2 is trained with our proposed cross-validation strategy, while that in Table 3.3 is trained in a traditional way (without error control). Benefited from our proposed cross-validation training strategy, the accuracy of each single CNN based on the synthesized faces in Table 3.2 keeps relatively high performance. Averagely, CNNs in age groups "young", "adult" and "old age" even have a little higher performance than PEM. It tells that the synthesized faces maintain most of the information related to identity. The performances in "young" and "adult" are a little higher than those in "child" and "old-age", which is because that more faces in the aging patterns from "child" and "old-age" are lost than those in "young" and "adult". In contrast, the CNNs trained in the traditional way in Table 3.3 shows much lower performance than that in Table 3.2. With the traditional way of training, a^2-DAE suffers from heavy overfitting in training, which produces unbalanced training and testing errors. Our proposed

training strategy takes the reconstruction errors into consideration, which can better control overfitting in training and testing and produces error-aware outputs.

Table 3.2: The accuracies of CNNs based on the synthesized faces in each age group of the five folds, which are trained by the proposed cross-validation way.

Group	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg
"child"	0.7105	0.7090	0.6895	0.7240	0.7165	0.7099
"young"	0.7295	0.7255	0.7005	0.7451	0.7075	0.7216
"adult"	0.7415	0.7260	0.7180	0.7280	0.7270	0.7281
"old-age"	0.7195	0.7105	0.6980	0.7180	0.7190	0.7130
original	0.7665	0.7595	0.7460	0.7570	0.7630	0.7584

Table 3.3: The accuracies of CNNs based on the synthesized faces in each age group of the five folds, which are trained in the traditional way.

Group	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg
"child"	0.6435	0.6250	0.6345	0.6495	0.5820	0.6269
"young"	0.5650	0.5560	0.5805	0.5945	0.5820	0.5756
"adult"	0.6225	0.6060	0.6220	0.6335	0.6260	0.6220
"old-age"	0.6475	0.6535	0.6690	0.6690	0.6400	0.6558
original	0.7665	0.7595	0.7460	0.7570	0.7630	0.7584

Comparison of CNNs with different shape and texture combinations

The comparison of CNNs with different shape and texture combinations is shown in Table 3.4. The performance is evaluated based on the different preprocessing of the original images. "shp/tex" means the shape and the texture are separated in the preprocessing step, but both are input into CNNs for joint optimization. The "texture" means only texture information is used as the input. "3 points" means the faces are aligned by the similarity transform from the two eye centers and the mouth center. It shows that "shp/tex' has a quite limited performance improvement than "texture". Restricted by the pose and expression changes in the real face images, the shape information cannot provide much discriminative information. "shp/tex' and "texture" have much higher performance than "3 points" alignment. Simply based on three landmarks of the eye centers and the mouth center, the image pixels on the faces cannot be well aligned, which will lower down the performance.

Table 3.4: The accuracies of CNNs based on different shape and texture combinations, trained from the original faces. "shp/tex" indicates shape and texture.

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg
shp/tex	0.7665	0.7595	0.7460	0.7570	0.7630	0.7584
texture	0.7715	0.7520	0.7485	0.7510	0.7605	0.7567
3 points	0.7455	0.7095	0.7065	0.7340	0.7365	0.7264

3.6 Chapter Summary

In this work, we have studied the aging problem in realistic face recognition. We have developed a novel framework DAFR for aging face recognition with large gaps. To alleviate the performance drop caused by aging, two modules, aging pattern synthesis and aging face verification, are included in this framework. In the aging pattern synthesis module, we have proposed a novel deep aging-aware denoising auto-encoder (a^2-DAE) to synthesize the faces of four age groups for the input face of an arbitrary age. In the aging face verification module, given a face pair as the input, each pair of synthesized faces of the same age group is fed into a parallel CNN, and multiple parallel CNNs are fused to give the final verification score. To avoid overfitting in the aging pattern synthesis module, the cross-validation strategy is used to produce error-aware outputs. Extensive experiments on the CAFE dataset have verified the effectiveness of our proposed framework.

Chapter 4

Clothing Attributes Assisted Person Re-identification

In this chapter, we investigate person re-identification in realistic video surveillance system [84]. In most cases, surveillance cameras are of low quality, and the videos are captured with multiple view points and illuminations. Thus, the faces are usually not identifiable and can not be used as clues for person re-identification. We propose to use clothes information and part based approach to handle this difficult situation. These informations are well integrated into a latent support vector machines framework for recognition. To evaluate our algorithm, we construct a large-scale and realistic dataset, which is collected from NUS canteen.

4.1 Introduction

Person re-identification is to match people across non-overlapping camera views. It has a wide range of applications and great commercial value. However, it still remains an unsolved problem because of the low video quality and variations of viewpoints, pose and illumination [142].

In the video surveillance system, to get a wider perspective range, cameras are usually installed at positions much higher than the height of people. High camera position leads to longer sight distance, which make it difficult to get clear faces.

Therefore, the appearance of people is most influenced by their clothes. Most person re-identification approaches represent clothing appearance by low-level texture descriptors [143, 51, 168, 118, 37, 169, 170, 63], which are similar to the feature representation for rigid object recognition, e.g. face recognition. Compared with rigid objects, the variations of clothing appearance are severer and more complex. Low-level descriptors cannot well distinguish whether the variation is caused by identity difference or other factors like body movements. There exists a semantic gap between the low-level descriptors and the high-level classification task.

Despite the great variety of clothes, people usually wear ordinary clothes in daily life, which have similar characteristics. Such characteristics bear middle-level semantic meanings and can be utilized to bridge the semantic gap. Based on the above observations, we propose a clothing attributes [67] assisted approach for person re-identification.

Middle-level attributes can be embedded into a high-level classifier as latent variables via a latent support vector machines (LSVM) framework [144, 163]. Attributes are considered as discrete-valued variables. We find that describing clothing attributes by discrete values usually cannot cover the diversity of clothes. For example, trousers can be roughly represented as short and long, and based such categorization, it is difficult to tell whether a pair of knee pants belong to shorts or longs. Therefore, the attributes are treated as continuous-valued variables.

Moreover, we also propose a part based approach for representing the appearance of pedestrians. Because of body movements, commonly used holistic feature representation methods suffer from the pose misalignments. The proposed part based approach reduces the misalignment and achieves considerable improvements.

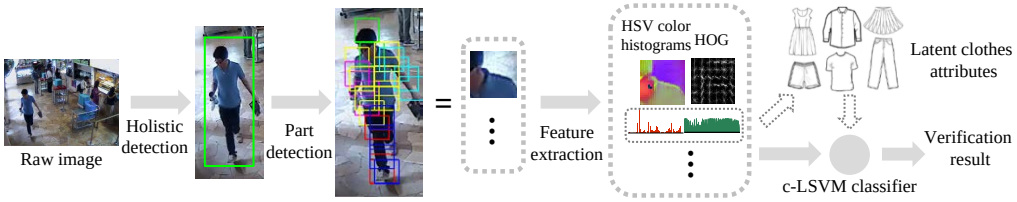


Figure 4.1: The flowchart of our proposed person re-identification method.

Besides, a large-scale dataset is constructed in this work. Existing publicly available datasets for person re-identification are limited in sample number and camera views. The frequently used VIPeR [50] dataset, for example, only consists of 632 image pairs captured from 2 cameras, while our dataset contains more than 1,000 video clips recorded from 10 cameras. We evaluate the proposed person re-identification approach on this dataset.

The contributions of this work can be summarized as:

- A latent SVM approach for clothing attributes assisted person re-identification.
- A part-based approach for representing the appearance of pedestrians.
- A large-scale dataset for person re-identification.

4.2 Related Work

In recent years, person re-identification has attracted growing attention in computer vision community. A brief literature survey can be found in [142].

Studies on person re-identification can be roughly divided into two categories, i.e. *feature* and *learning* [142]. For feature, color histograms and local texture descriptors are commonly utilized [168, 118, 169, 170, 63]. Besides, Ad Hoc features are also proposed, including local motion features [45], shape and appearance context features [143] and the symmetry-driven accumulated local features [37].

For learning models, AdaBoost [51], partial least squares [123] and ranking SVM [118] have been applied in person re-identification. Recently, metric learning models have become popular in person re-identification, which include the probabilistic relative distance comparison model [169] and the relaxed Mahalanobis metric learning model [63]. Besides metric learning, Zheng et al. [170] introduced transfer learning techniques for image set based person re-identification.

Attributes information has been utilized in computer vision in recent years. Liu *et al.* [92] use an information theoretic approach to discover the attributes of human actions automatically. The inferred attributes are embedded into a latent SVM classifier for action recognition. Yamaguchi et al. [160] used clothing attributes in

clothes parsing. In the work of Liu *et al.* [103], clothing attributes are used in cross-scenario clothing retrieval. The work of Vaquero et al. [139] is the first to introduce middle level attributes in human recognition. However, the attributes used in this work are mainly facial attributes. The recent work of Layne et al. [76, 75] on person re-identification is the most related one to our approach. In their work, 15 attributes are defined and predicted by SVM. However, these attributes are only intuitively used as a kind of new features to help improve human re-identification, and the basic idea is quite straightforward, and thus we do not further compare with these two works in our experiments.

4.3 Proposed Framework

This section describes the elements of the proposed person re-identification framework except for the latent SVM model, which will be shown in the following section.

The process of re-identifying a person in the video surveillance system usually includes three necessary steps: human detection, visual feature representation and classification. Our method, however, contains two more steps, corresponding to two key aspects of our contributions respectively, as shown in Figure 5.2. One of them is body part detection, which is described in Subsection 4.3.1 together with the resultant feature representation. The other one is embedding clothing attributes into the classifier. In Subsection 4.3.2, we describe the properties of the classifier. The definitions of clothing attributes are given in Subsection 4.3.3.

4.3.1 Part-based Feature Representation

In the proposed method, the input of body part detection is an initial bounding box of holistic human body, which is obtained by a deformable part model based cascade detector [38]. Then we perform body part detection by using the method of Yang and Ramanan [161], where human body is represented by several local parts. Candidates of these local parts are also obtained by the deformable part model based detector [40]. These local part candidates produce many candidates

Figure 4.2: Body part detection results illustrated in skeletons.

of configurations. Part locations are estimated by selecting the configuration with the best matching score. Figure 4.2 shows some results of part detection in colored skeletons, from which we can see that performing body part detection improves the alignment and provides useful and necessary information for further analysis.

The next step after part detection is visual feature representation. We sample local patches centered at each body part, and then normalize them. As shown in Figure 5.2, histograms of oriented gradients (HOG) [27] and color histograms in the hue, saturation and value (HSV) space are extracted from the normalized patches. Consequently, the appearance of a person is described by a feature vector, which is obtained by concatenating features of all aligned parts. To lower the computational cost, the dimension of the feature vector is reduced by principal component analysis (PCA).

4.3.2 Open Set Person Re-identification

Person re-identification can be considered as either a closed set identification problem or an open set verification problem. Due to the limitation of publicly available

datasets, many works on person re-identification are evaluated in closed set experimental settings [143, 51, 37, 118, 169, 63]. As shown in Figure 4.3, in many scenarios, people appearing in one camera do not necessarily appear in another camera, and a camera view may include people never appearing in other cameras. For a boarder range of applications, person re-identification is treated as a verification problem.

Verification can be formulated as a binary classification problem, i.e. whether two testing samples belong to the same person or not. In this work, we tackle this problem by a LSVM classifier. The input is a pair of testing samples, and the output is the confidence of their belonging to the same person.

Figure 4.3: The illustration of open set person re-identification task, in which people appearing in one camera do not necessarily appear in the other, and a camera view may include people never appearing in other cameras.

Since person re-identification is an application in the video surveillance system, multiple images of a person are usually available. The problem can be further formulated as a set-to-set classification problem. To simplify this problem, the LSVM classifier is trained and tested on single image pairs. Based on the similarities between image pairs, the similarity between a pair of video clips or image sets is measured by set-to-set metric, for example, the Hausdorff distance [56].

4.3.3 Clothing Attributes

In this paper, we use middle-level clothing attributes to bridge the semantic gap between low-level features and high-level classification task. In the literature of

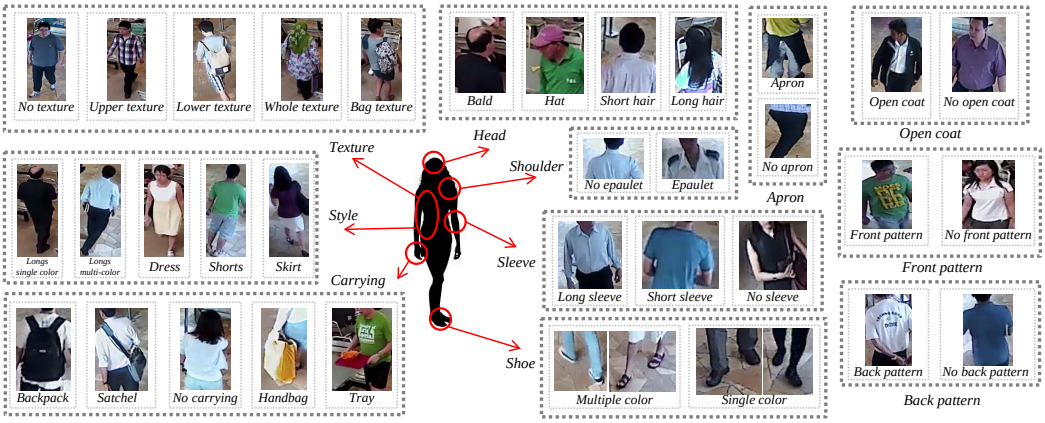


Figure 4.4: The definitions of clothing attributes.

computer vision, clothing attributes are obtained via two approaches. In the work of Yamaguchi et al. [160], the clothing attributes are crawled from the fashion websites. Although such data acquisition method can provide plenty of attributes, these attributes are not suitable for person re-identification. For example, *jacket* and *coat* are difficult to distinguish in low-quality surveillance videos. In person re-identification task, the attributes should be more visually separable.

Besides mining web data, the other approach for obtaining clothing attributes is manual design [76]. In this work, we define 11 kinds of clothing attributes, each has 2∼5 attributes. The combined number of attributes are bigger than that in [76]. Details of the attribute definitions are shown in Figure 4.4.

For operability in manual annotation, clothing attributes are usually labeled as discrete values [76, 75, 160, 103]. However, in the real-world, clothing characteristics are unnecessarily discretely distributed. For example, longs and shorts are usually used to describe the length of trousers. Under such categorization, the knee pants are ambiguous. There may exist uncertainties on the values for the clothing attributes. Therefore, though we annotate the clothing attributes by discrete values, they are relaxed to continuous values in the model learning and inference processes of the proposed person re-identification approach, namely the binary values for each value of attribute is relaxed to be real valued indicating the confidence for the attributes to be assigned the given value.

4.4 The Latent SVM Model

Intuitively, discrete attributes can be embedded into discriminative classifier as latent variables via a latent SVM (LSVM) framework [144]. To adapt the continuous-valued attributes, we follow the spirit of [163], and model the relations among the low-level features, the clothing attributes and the re-identification label all in real-value space.

Denote $\mathbf{x} \in \mathbb{R}^{N_x}$ and $\mathbf{a} \in \mathbb{R}^{N_a}$ the low-level feature vector and clothing attribute vector of image I, where N_x and N_a are the dimensions of $\mathbf{x}$ and $\mathbf{a}$. Note that N_a equals to the sum of attribute value numbers for all clothing attributes mentioned in Section 4.3.3. $\mathbf{a} = [a_1, a_2, \cdots, a_{N_a}]^T$, where a_i is the value indicating the confidence of being assigned to certain value of certain attribute, indexed by i. The training set is represented as a set of N sample tuples $\{(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}_n^1, \mathbf{a}_n^2, y_n)\}_{n=1}^N$, where $\mathbf{x}_n^1, \mathbf{x}_n^2$ and $\mathbf{a}_n^1, \mathbf{a}_n^2$ are the low-level feature vectors and the clothing attribute vectors of the image pair (I_n^1, I_n^2) respectively, and y_n is the re-identification label[1] indicating the confidence of (I_n^1, I_n^2) belonging to the same person.

4.4.1 Model Formulation

The relations among the low-level features, clothing attributes and re-identification label are represented in linear models:

$$\begin{aligned} y &= \mathbf{w}_{x,y}^T \tilde{\mathbf{x}} + b_{x,y}, \\ y &= \mathbf{w}_{a,y}^T \tilde{\mathbf{a}} + b_{a,y}, \\ a_i^1 &= \mathbf{w}_{x,a_i}^T \mathbf{x}^1 + b_{x,a_i}, \\ a_i^2 &= \mathbf{w}_{x,a_i}^T \mathbf{x}^2 + b_{x,a_i}, \forall i. \end{aligned} \tag{4.1}$$

These linear models predict the re-identification label from the raw features, the re-identification label from the clothing attributes, and the clothing attributes from

[1]The re-identification label y_n is annotated by binary value, and relaxed to real value in the proposed method.

the corresponding raw feature vectors, respectively. Here $\tilde{\mathbf{x}}$ is defined as $|\mathbf{x}^1 - \mathbf{x}^2|$, element-wise absolute values of the difference between $\mathbf{x}^1$ and $\mathbf{x}^2$. $\tilde{\mathbf{a}}$ is defined as $\tilde{\mathbf{a}} = [\mathbf{a}^1; \mathbf{a}^2]$. $\mathbf{w}_{x,y}$, $b_{x,y}$, $\mathbf{w}_{a,y}$, $b_{a,y}$, $\mathbf{w}_{x,a_i}$ and b_{x,a_i} are the *Regression Parameters*, which will be determined in model learning process.

The loss function of each linear model can be defined as:

$$\begin{aligned} L_{x,y}(\tilde{\mathbf{x}}, y) &= (\mathbf{w}_{x,y}^T \tilde{\mathbf{x}} + b_{x,y} - y)^2, \\ L_{a,y}(\tilde{\mathbf{a}}, y) &= (\mathbf{w}_{a,y}^T \tilde{\mathbf{a}} + b_{a,y} - y)^2, \\ L_{x^1,a_i^1}(\mathbf{x}^1, a_i^1) &= (\mathbf{w}_{x,a_i}^T \mathbf{x}^1 + b_{x,a_i} - a_i^1)^2, \\ L_{x^2,a_i^2}(\mathbf{x}^2, a_i^2) &= (\mathbf{w}_{x,a_i}^T \mathbf{x}^2 + b_{x,a_i} - a_i^2)^2, \forall i. \end{aligned} \tag{4.2}$$

The re-identification label y is inferred by minimizing the overall loss, namely maximizing the following function:

$$\begin{aligned} &\mathbf{z}^T \phi(\mathbf{x}^1, \mathbf{x}^2, \mathbf{a}^1, \mathbf{a}^2, y) = \\ &- \beta_1 L_{x,y}(\tilde{\mathbf{x}}, y) - \beta_2 L_{a,y}(\tilde{\mathbf{a}}, y) \\ &- \sum_{i=1}^{N_a} \lambda_i^1 L_{x^1,a_i^1}(\mathbf{x}^1, a_i^1) - \sum_{i=1}^{N_a} \lambda_i^2 L_{x^2,a_i^2}(\mathbf{x}^2, a_i^2), \end{aligned} \tag{4.3}$$

where β_1, β_2, λ_i^1 and λ_i^2 are nonnegative *Model Parameters*.

$$\mathbf{z} = [\beta_1, \beta_2, \lambda_1^1, \lambda_2^1, \ldots, \lambda_{N_a}^1, \lambda_1^2, \lambda_2^2, \ldots, \lambda_{N_a}^2]^T \tag{4.4}$$

is a vector for all the model parameters, and the vector of all the negative loss is written as:

$$\begin{aligned} &\phi(\mathbf{x}^1, \mathbf{x}^2, \mathbf{a}^1, \mathbf{a}^2, y) = [-L_{x,y}(\tilde{\mathbf{x}}, y); -L_{a,y}(\tilde{\mathbf{a}}, y); -\tilde{L_1}; -\tilde{L_2}], \\ &\tilde{L_1} = [L_{x^1,a_1^1}(\mathbf{x}^1, a_1^1), L_{x^1,a_2^1}(\mathbf{x}^1, a_2^1), \cdots, L_{x^1,a_{N_a}^1}(\mathbf{x}^1, a_{N_a}^1)]^T, \\ &\tilde{L_2} = [L_{x^2,a_1^2}(\mathbf{x}^2, a_1^2), L_{x^2,a_2^2}(\mathbf{x}^2, a_2^2), \cdots, L_{x^2,a_{N_a}^2}(\mathbf{x}^2, a_{N_a}^2)]^T. \end{aligned}$$

Equation (4.1) describes the relations among features, attributes and labels, while the objective is given in Equation (4.3).

4.4.2 Model Learning

To maximize the function in Eqn. (4.3), the regression and model parameters need to be determined. The former are learned by max-margin regressions [36]. The latter ones are obtained by a latent max-margin framework with $\mathbf{a}^1, \mathbf{a}^2$ as latent variables. The objective function for learning the model parameters $\mathbf{z}$ is shown in Sec. 4.4.2, while the corresponding optimization method is described in Sec. 4.4.2.

Problem Formulation for LSVM

The optimal values of model parameters $\mathbf{z}$ are learned via a latent SVM framework [144, 163], the objective function is written as:

$$\begin{aligned} & \min \frac{\gamma}{2}\|\mathbf{z}\|^2 + \frac{1}{N}\sum_{n=1}^{N}\zeta_n \\ \text{s.t.} \quad & \mathbf{z} \geq 0, \\ & \max_{\mathbf{a}^1,\mathbf{a}^2} \mathbf{z}^T\phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}^1, \mathbf{a}^2, y_n) \geq \\ & \max_{\mathbf{a}^1,\mathbf{a}^2} \mathbf{z}^T\phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}^1, \mathbf{a}^2, \hat{y}_n) + 1 - \zeta_n, \forall n, \hat{y}_n \in \hat{\mathcal{Y}}_n. \end{aligned} \tag{4.5}$$

Here ζ_n is a slack variable, and $\gamma > 0$ is the balance weight. $\hat{\mathcal{Y}}_n$ is defined as:

$$\hat{\mathcal{Y}}_n = \begin{cases} \{y|y < 1-\rho\} & y_n = 1 \\ \{y|y > \rho\} & y_n = 0 \end{cases} \tag{4.6}$$

where ρ is a parameter to control the tolerance of the prediction error and set as 0.5 throughout the experiments in this work.

Note that the attributes $\mathbf{a}^1$ and $\mathbf{a}^2$ are latent variables, which are not predetermined but rather inferred. When $\mathbf{x}^1$, $\mathbf{x}^2$ and y are fixed, the inferred $\mathbf{a}^1$ and $\mathbf{a}^2$ should maximize the function of Eqn. (4.3), which corresponds to the maximization operations in Eqn. (4.6). In discrete latent SVM [144], the optimal $\mathbf{a}^1$ and $\mathbf{a}^2$ are obtained by searching all the possible discrete value combinations, which is time consuming. In the continuous latent SVM [163], they can be directly inferred by

continuous quadratic programming as later introduced.

To guarantee the discriminating power of the derived model, the objective function value corresponding to ground truth re-identification label y_n should be greater than those corresponding to imperfect labels $\hat{y}_n \in \hat{\mathcal{Y}}_n$, which is expressed as the constraint in Eqn. (4.6).

Figure 4.5: Example frames of the 10 camera views in NUS-Canteen database.

Optimization

The above optimization problem can be solved by minimizing its Lagrange form [163, 144],

$$\mathcal{L}(\mathbf{z}) = \frac{\gamma}{2}\|\mathbf{z}\|^2 + \theta^T\mathbf{z} + \frac{1}{N}\sum_{n=1}^{N} R^n(\mathbf{z}), \tag{4.7}$$

where θ is Lagrange multiplier and

$$R^n(\mathbf{z}) = \max_{\substack{\mathbf{a}^1,\mathbf{a}^2 \\ \hat{y}_n \in \hat{\mathcal{Y}}_n}} \mathbf{z}^T\phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}^1, \mathbf{a}^2, \hat{y}_n) \tag{4.8a}$$

$$- \max_{\mathbf{a}^1,\mathbf{a}^2} \mathbf{z}^T\phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}^1, \mathbf{a}^2, y_n) + 1. \tag{4.8b}$$

Eqn. (4.8a) and Eqn. (4.8b) can be solved by quadratic programming, and the closed form solution for variable $\tilde{\mathbf{a}}_n = [\mathbf{a}_n^1; \mathbf{a}_n^2]$ of the n-th training sample in Eqn.(4.8b) is

given by

$$\begin{aligned}
&\tilde{\mathbf{a}}_n^\star = (\beta_2 \mathbf{w}_{a,y}\mathbf{w}_{a,y}^T + \mathbf{\Lambda}^2)^{-1}(\mathbf{\Lambda}^2\tilde{\mathbf{W}} + \beta_2(y_n - b_{a,y})\mathbf{w}_{a,y}),\\
&\tilde{\mathbf{W}} = [\mathbf{W}\mathbf{x}_1 + \mathbf{b}_{x,a}; \mathbf{W}\mathbf{x}_2 + \mathbf{b}_{x,a}],\\
&\mathbf{W} = [\mathbf{w}_{x,a_1}, \mathbf{w}_{x,a_2}, \ldots, \mathbf{w}_{x,a_{N_a}}]^T,\\
&\mathbf{b}_{x,a} = [b_{x,a_1}, b_{x,a_2}, \ldots, b_{x,a_{N_a}}]^T.\\
&\mathbf{\Lambda} = diag(\lambda_1^1, \lambda_2^1, \ldots, \lambda_{N_a}^1, \lambda_1^2, \lambda_2^2, \ldots, \lambda_{N_a}^2).
\end{aligned} \tag{4.9}$$

The optimal values of $\mathbf{a}_n^1$, $\mathbf{a}_n^2$ and $\hat{y}_n$ in Eqn. (4.8a) can also be solved by quadratic programming, but in a constrained way. It can be generated as a standard quadratic programming problem:

$$\mathbf{t}^\star = \arg \max_{\substack{\mathbf{t}=[\mathbf{a}_n^1;\mathbf{a}_n^2;\hat{y}_n] \\ \hat{y}_n \in \mathcal{Y}_n}} \frac{1}{2}\mathbf{t}^T\mathbf{H}\mathbf{t} + \mathbf{t}^T\mathbf{b} \tag{4.10}$$

Define

$$\begin{aligned}
\tilde{\mathbf{w}}_{x,y} &= [0; -1; \mathbf{w}_{x,y}\tilde{\mathbf{x}}],\\
\tilde{\mathbf{w}}_{a,y} &= [\mathbf{w}_{a,y}; -1; 0],\\
\tilde{\Lambda} &= \begin{bmatrix} \Lambda^T\Lambda & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\\
A &= \beta_1 \cdot \tilde{\mathbf{w}}_{x,y}\tilde{\mathbf{w}}_{x,y}^T + \beta_2 \tilde{\mathbf{w}}_{a,y}\tilde{\mathbf{w}}_{a,y}^T + \tilde{\Lambda}\\
&= \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix}
\end{aligned}$$

where $A_{2,2}$ is a scalar, and $H = A_{1,1}$, $\mathbf{b} = [-4 \cdot \Lambda^T\Lambda\tilde{\mathbf{w}}_{x,y}; 0]$. The solution is given by

$$\begin{aligned}
\tilde{\mathbf{a}}^* = &(\frac{\beta_1\beta_2}{\beta_1+\beta_2}\mathbf{w}_{a,y} \cdot \mathbf{w}_{a,y}^T + \Lambda^T\Lambda)^{-1}\\
&\cdot (\Lambda^T\Lambda\tilde{\mathbf{w}} + \frac{\beta_1\beta_2}{\beta_1+\beta_2}(\mathbf{w}_{x,y}^T\mathbf{x} + b_{x,y})\mathbf{w}_{a,y})
\end{aligned} \tag{4.11}$$

$$y^\star = \frac{\beta_1(\mathbf{w}_{x,y}^T\mathbf{x} + b_{x,y}) + \beta_2(\mathbf{w}_{a,y}^T\tilde{\mathbf{a}}^\star + b_{a,y})}{\beta_1 + \beta_2} \tag{4.12}$$

The $\mathbf{z}$ can be solved by minimizing Eqn. (4.7) using sub-gradient descent method [17]. The sub-gradient is calculated as follows[2],

$$\begin{aligned}\partial\mathcal{L}(\mathbf{z}) = \gamma \cdot \mathbf{z} + \theta + \sum_{n=1}^{N} \phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}_n^{1,*}, \mathbf{a}_n^{2,*}, \hat{y}_n^*) \\ - \sum_{n=1}^{N} \phi(\mathbf{x}_n^1, \mathbf{x}_n^2, \mathbf{a}_n^{1,\star}, \mathbf{a}_n^{2,\star}, y_n).\end{aligned} \tag{4.13}$$

4.4.3 Inference

With the learned regression parameters and model parameters, the re-identification label y^* of a pair of testing images ($\mathbf{x}^1$,$\mathbf{x}^2$) can be inferred by

$$\{\mathbf{a}^{1,*}, \mathbf{a}^{2,*}, y^*\} = \arg\max_{\mathbf{a}^1, \mathbf{a}^2, y} \mathbf{z}^T \phi(\mathbf{x}^1, \mathbf{x}^2, \mathbf{a}^1, \mathbf{a}^2, y). \tag{4.14}$$

The solutions are in closed-form, where $\mathbf{a}^{1,*}$ and $\mathbf{a}^{2,*}$ can be obtained by a quadratic programming solver. Based on the implicitly inferred $\tilde{\mathbf{a}}^* = [\mathbf{a}^{1,*}; \mathbf{a}^{2,*}]$, y^* is given by:

$$y^* = \frac{\beta_1}{\beta_1 + \beta_2}(\mathbf{w}_{x,y}^T \tilde{\mathbf{x}} + b_{x,y}) + \frac{\beta_2}{\beta_1 + \beta_2}(\mathbf{w}_{a,y}^T \tilde{\mathbf{a}}^* + b_{a,y}).$$

4.4.4 Discussions

The goal of this work is to enhance the person re-identification by extra clothing attribute knowledge, which can be achieved by both discrete and real-value attributes. Under the framework of latent SVM, there exist two main differences between the proposed approach and conventional discrete attributes based method [144]:

- Treated as continuous-valued variables, the relations among the input images, attributes and the re-identification labels are modeled by regression models. In discrete attribute based LSVM, the corresponding parts are modeled by SVMs or just simple co-occurrence statistics.
- The maximization of the terms in Eqn. (4.8) and (4.14) is obtained by quadratic programming in the proposed approach. The corresponding operation in discrete attribute based LSVM is performed by brutal force search,

[2]Here * and * denote the optimal prediction of corresponding variable.

which may be very time consuming when attribute value space is huge.

Overall, continuous attributes based LSVM is more flexible in modeling the uncertainties in attribute value assignments and re-identification label.

4.5 Database

4.5.1 The NUS-Canteen Database

In person re-identification, VIPeR [50], i-LIDS [168] and ETHZ [123] are the most frequently used datasets. They are limited in sample numbers and camera views. To address this problem, a large-scale person re-identification dataset is collected and annotated. As shown in Figure 4.5, the raw videos are captured from 10 cameras installed at an university canteen. The canteen has roofs but no inclosure wall, and therefore can be considered as a semi-outdoor scenario. The illumination is influenced by both controlled lights and the sunlight. There are multiple entrances in the canteen, and the cameras cannot completely cover.

We have annotated 1,129 video clips, each corresponds to one person and contains 12∼61 frames. 74.31% of the clips contain 61 frames. There are 215 people annotated in total. Each person appears in 1∼6 camera(s) and has 2∼19 videos which corresponds to 51∼970 total frames. The detailed statistics of the data are shown in Figure 4.6. On average, one person appears in more than 3 cameras and has more than 5 video clips. The mean frame/image number for a person is 287, which is much bigger than the number in VIPeR (2 images) and i-LIDS (4 images). The database can be referred and downloaded from website "http://www.lv-nus.org/nus_canteen/".

4.5.2 Evaluation Settings

The database is divided into the training set and the testing set. The former is used to train the person re-identification model, while the latter is used for evaluation. There is no intersection between them. Since the person re-identification is treated as

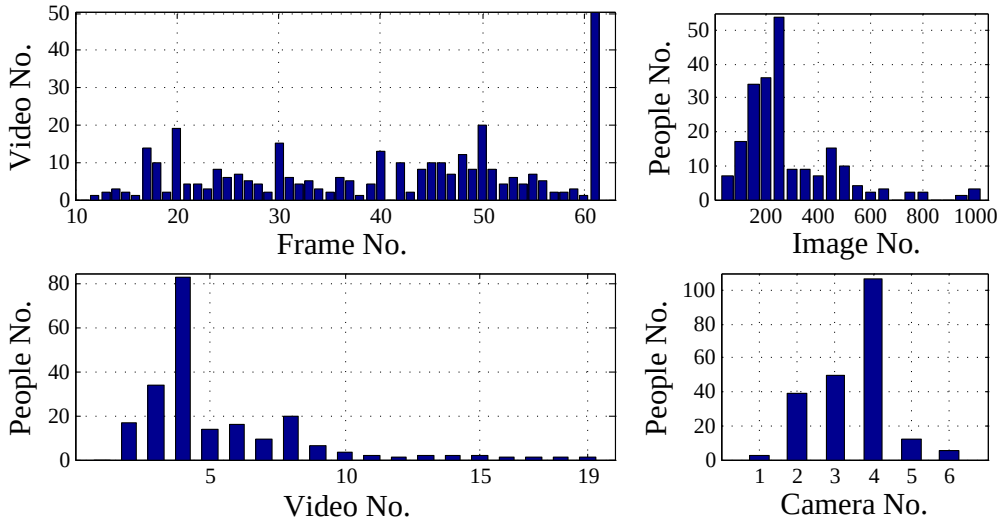


Figure 4.6: The statistics of NUS-Canteen database.

an open-set verification problem, training and testing are performed on sample pairs. The pair number of different persons is much bigger than that of the same person. To balance and reduce the computation cost, a subset of the pairs of different people for training and testing is randomly sampled. The concrete statistics of training set and testing set are shown in Table 4.1. The performance of a person re-identification approach is measured by the receiver operating characteristic (ROC) curves.

Table 4.1: The sample and pair number.

Set	Subject number	Video number	Pair number (same person)	Pair number (different persons)
Training	100	514	1512	4884
Testing	115	615	1889	4918

4.6 Experiments

The experiments are conducted include two phases. In Subsection 4.6.1, we make comparisons between the holistic feature representation and the proposed part-based approach. The proposed clothing attributes assisted person re-identification approach is validated in Subsection 4.6.2.

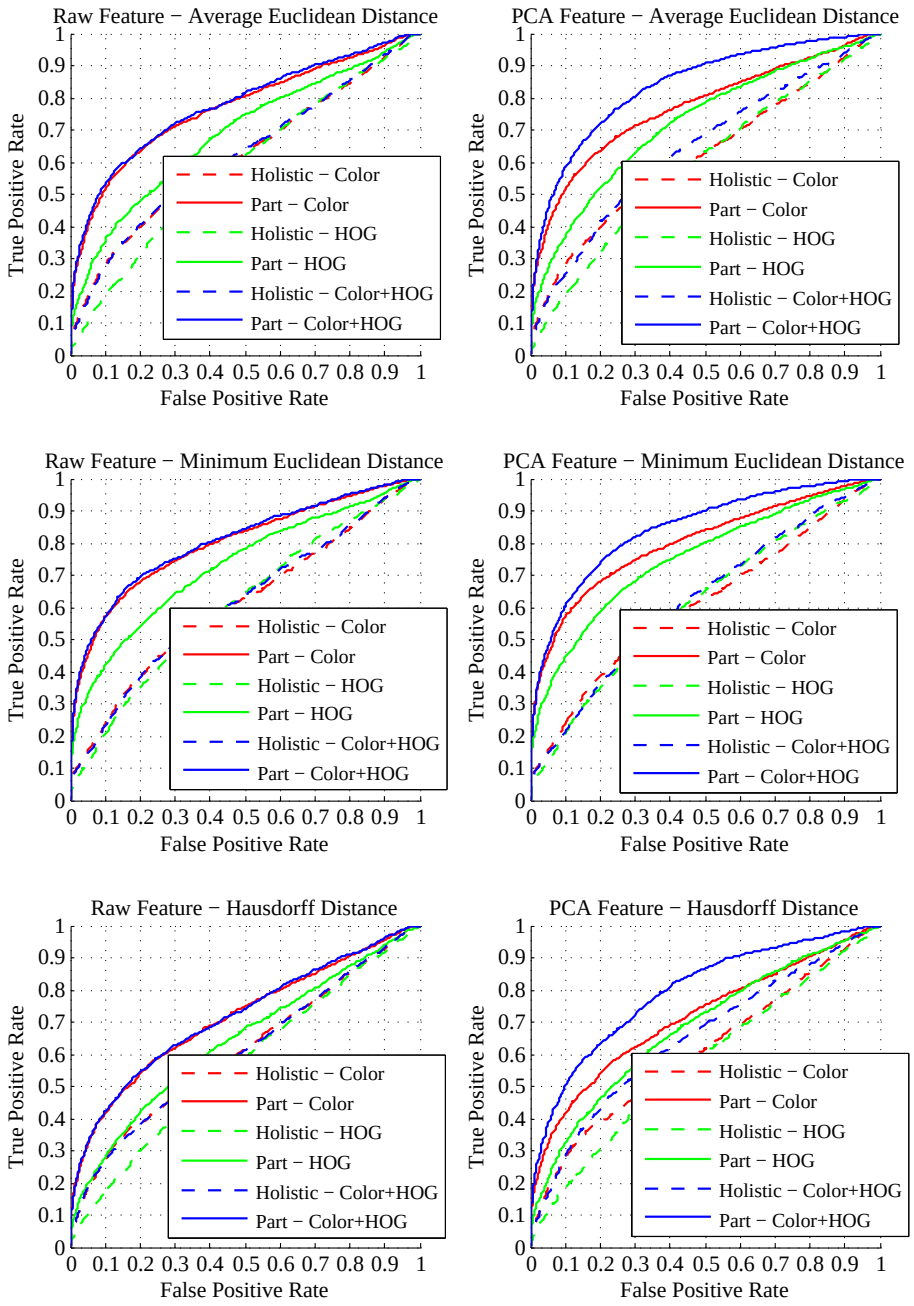


Figure 4.7: Performance of holistic (dashed lines) and part-based (solid lines) feature representations approaches compared using average Euclidean distance (left column), minimum Euclidean distance (middle column) and Hausdorff distance (right column), with (bottom row) and without PCA enhancement (top row).

Figure 4.8: Holistic human detection results obtained by [38].

4.6.1 Holistic vs. Part-based Feature Representation

As shown in Figure 5.2, the first step of person re-identification is the holistic pedestrian detection. For this step, the detector in [38] is used, and some detection results are shown in Figure 4.8. In the next step, body part detection are performed using the approach in [161].

In the holistic feature representation, the detected holistic human regions are first normalized (see Figure 4.8) to 48×128 and divide them into 3×8 grid of 16×16 non-overlapping patches. HOG and color histogram features are extracted from each patch. The size of HOG cell is set to 4 and the color histograms are quantified into 16 bins in each channel. Consequently, each patch is represented by a 48 dimensional color feature and a 124 dimensional HOG feature vector. The total length of HOG and color feature vector is 2,976 and 1,152.

Following the definitions in [161], human body is divided into 26 local parts. In the experiments, the body parts are normalized to 32×32. The size of HOG cell is set to 8. The color histograms are quantified into 16 bins in each channel. Consequently, the length of color and HOG feature vectors extracted from a body part are 48 and 496 respectively. Then a 1248 dimensional color vector and a 12896

dimensional HOG vector are obtained for each sample.

Since the dimensions are too high, the dimension of HOG and color vectors is reduced to 1000 by PCA. The combination of color and HOG feature is simply done by concatenating them into one vector.

The performance comparisons between holistic and part-based feature representation are shown in Figure 4.7. Besides the Hausdorff distance, the results using minimum and average Euclidean distance as set-to-set metrics are also shown. As can be seen, no matter using what type of visual features, part-based feature representation is obviously better than holistic feature representation. Performing PCA also enhances the representation power. Based on the PCA feature, integrating color and HOG feature also improves the performance. The experimental results show that the proposed part-based feature representation is very effective. We also find that simply using average Euclidean distance gets better performance than Hausdorff distance.

The part-based and PCA enhanced color+HOG feature representation achieves the best results. We use this feature in the next experiments.

4.6.2 With vs. Without Clothing Attributes Assistance

To validate the effectiveness of embedding clothing attributes into person re-identification, comparisons between SVM classifier and latent SVM classifier are made. The former does not use any clothing attributes information and corresponds to the first term of Eqn. (4.1). The latter integrates attributes information and corresponds to all the four terms of Eqn. (4.1). To make fair comparison, the implementations of SVM and latent SVM model are both linear [36].

The performance comparisons are shown in Figure 4.9. Besides the ROC curves of SVM, discrete attributes based latent SVM (d-LSVM) and continuous-value attributes based latent SVM (c-LSVM), the results of only using attributes (*Attributes*) are plotted, which is obtained by disabling $\mathbf{w}_{x,y}$ in Eqn. (4.1). The ROC of PCA features is shown as a baseline. Three kinds of set-to-set metrics are used in the evaluation, i.e., average Euclidean distance, minimum Euclidean distance and

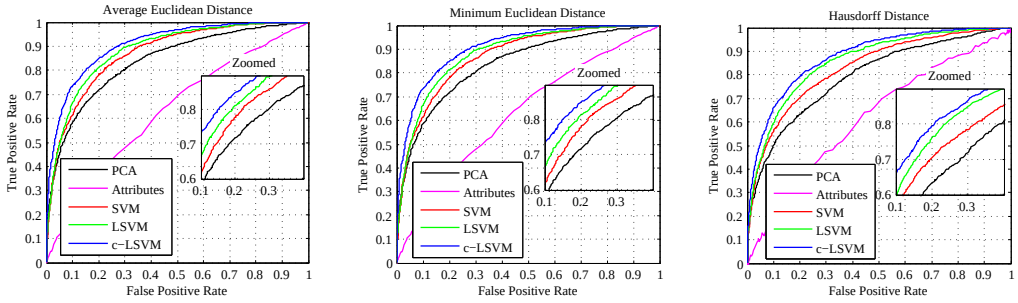


Figure 4.9: Re-identification performance comparisons between original PCA features, SVM, LSVM and the proposed c-LSVM. The PCA and SVM do not use clothing attributes information, while the LSVM and c-LSVM are clothing attributes assisted.

Hausdorff distance respectively.

As can be seen, the d-LSVM outperforms both SVM in all the three metrics. The proposed c-LSVM gets better performance than d-LSVM. In Table 4.2, the comparisons of equal error rate (EER) are shown. Compared with original PCA feature, SVM classifier with no clothing attributes enhancement reduces the EER by 2.96% using average distance. Enhanced by clothing attributes via c-LSVM, the EER is further reduced by 3.53%. The effectiveness of SVM in discriminative learning has been widely proved. Although solely using clothing attributes cannot achieve good results, the performance enhancement of clothing attributes is as strong as SVM. The results show that utilizing middle-level clothing attributes information is an effective way to improve the performance of person re-identification. Experimental results also show that the c-LSVM model outperforms d-LSVM. Treating clothing attributes as continuous valued varibales is more effective in person re-identification.

Besides SVM, a metric learning approach applied in person re-identification, which is named as *keep it simple and straightforward metric* (KISSME) [73], is also compared[3]. In the experiments it is found that KISSME is sensitive to the dimension of input feature. When using the 2000 dimentional features, there exists

[3]We have explored more metric learning approaches (Logistic Discriminant Metric Learning (IDML) [54], Large Margin Nearest Neighbor (LMNN) [146], Information Theoretic Metric Learning (ITM) [28]) based on the source codes available from http://lrs.icg.tugraz.at/research/kissme/. For the reasons of memory limit and the failure of convergence, IDML, LMNN and ITM do not work properly on NUS-Canteen.

the problem of singular matrix which is similar to the small sample size problem in Fisher linear discriminant analysis. To avoid this problem the dimension of input feature is reduced to 1000. As can be seen from Figure 4.9, the performance of KISSME is better than conventional SVM, but not as good as the proposed c-LSVM. It implies that the proposed method can extract additional discriminant information that metric learning cannot discover.

Note that most previous methods for human re-identification [37, 51, 63, 118, 143, 168, 169, 170] do not use clothing attributes, and directly model the relations between low-level features and re-identification label. These methods can be used to replace the first model in Eqn. (4.1) for further performance improvement of c-LSVM.

Table 4.2: Comparisons of equal error rates.

Metric	PCA	SVM	d-LSVM	c-LSVM	KISSME
Average	24.02%	21.06%	19.31%	17.53%	19.65%
Hausdorff	29.03%	25.00%	21.90%	20.37%	21.88%

4.6.3 Comparisons on VIPeR Dataset

Although the proposed approach is designed for open-set scenario, it can be applied in closed-set scenatios. To make comparions with the Attribute Interpreted Re-identification (AIR) method [75] which also ultilizes attributes, the proposed approach is evaluated on VIPeR dataset [50] which contains 632 pairs of pedestrians. For the pedestrian image is cropped and the image number is limit, it is difficult to train an effective part detection model on VIPeR. In the experiments, the 2784 dimensional holistic features provided by Zheng et al. [169][4] and 15 attributes in [75][5] are used. Following the experiment settings in [75], 382 and 250 pairs are used for training and testing respectively.

Experimental comparisons by Cumulative Match Curve (CMC) and recognition rates are shown in Figure 4.10 and Table 4.3[6] respectively. In attribute prediction,

[4] `http://www.eecs.qmul.ac.uk/~jason/metadata/viper.mat`

[5] `http://www.eecs.qmul.ac.uk/~rlayne/hosted/layne_qmul_bmvc2012_attribute_annots.zip`

[6] In Figure 4.10 and Table 4.3, 4.4, the performance of AIR and SDALF are cited from [75] and

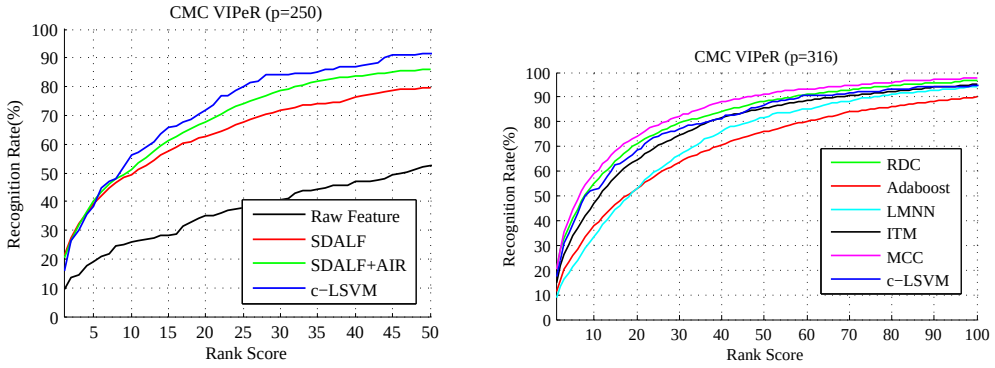


Figure 4.10: Re-identification performance comparisons on VIPeR dataset.

both the proposed approach and AIR use the features prposed by Zheng et al. [169]. In re-identification, the AIR is combined with Symmetry-Driven Accumulation of Local Features (SDALF) [37], while our approach still use the features of Zheng et al.. As can be seen, SDALF is more discriminative than the features proposed by Zheng et al.. Using less discriminative raw features, the performance of c-LSVM is close to AIR+SDALF in rank 1 to 5 and better than AIR+SDALF in rank 10, 25 and 50. The proposed method gets obviously bigger performance enhencements than AIR.

Besides attributes based method, comparisons with metric learning based person re-identification approaches on VIPeR are made. We follow the experiment setups in [169], the people number used in training and testing are both 316. Relative Distance Comparison (RDC) [169], Adaboost [51], Large Margin Nearest Neighbor (LMNN) [146], Information Theoretic Metric Learning (ITM) [28] and Metric Learning by Collapsing Classes (MCC) [46] are compared. As can be seen from Figure 4.10 and Table 4.4, MCC achieves the best overall results. The performance of the proposed method is approximately between those of ITM and RDC, and better than LMNN and Adaboost. It gets the best rank 1 recognition rate. These results well demonstrate the effectiveness of the proposed method. It should be pointed out that experimental results reported in Figure 4.10 and Table 4.3 and 4.4 are all obtained using features proposed by Zheng et al. [169] for fair comparison. Performance of

the results of RDC, Adaboost, LMNN, ITM and MCC are cited from [169].

the proposed approach and these compared metric learning methods can be further enhanced by improving the feature representation.

Table 4.3: Comparisons of recognition rates on VIPeR dataset using 250 people for test.

Method	Rank 1	Rank 5	Rank 10	Rank 25	Rank 50
AIR	5.56%	15.76%	24.72%	45.16%	65.68%
Weighted AIR	4.48%	17.44%	29.24%	50.60%	68.64%
SDALF	**18.28%**	37.88%	49.16%	67.96%	79.80%
SDALF+AIR	17.00%	36.48%	50.76%	72.52%	84.88%
SDALF+Weighted AIR	17.40%	**39.04%**	50.84%	74.44%	86.44%
Raw Feature	9.60%	19.20%	26.00%	38.00%	52.40%
c-LSVM	16.80%	38.40%	**56.00%**	**81.20%**	**91.60%**

Table 4.4: Comparisons of recognition rates on VIPeR dataset using 316 people for test.

Method	Rank 1	Rank 5	Rank 10	Rank 20
RDC	15.66%	38.42%	53.86%	70.09%
Adaboost	8.16%	24.15%	36.58%	52.12%
LMNN	6.23	19.65%	32.63%	52.25%
ITM	11.61%	31.39%	45.76%	63.86%
MCC	15.19%	**41.77%**	**57.59%**	**73.39%**
c-LSVM	**17.09%**	38.61%	52.53%	68.35%

4.7 Chapter Summary

In this chapter, we study the realistic issues in low resolution surveillance videos. Different from previous small-scale dataset, we collect a large-scale dataset, which contains more samples and camera views than currently available datasets. We propose to use middle-level clothing attributes information to assist person re-identification. The assistance is performed by embedding clothing attributes as latent variables via a latent SVM framework. We further improve the proposed approach by treating clothing attributes as real-value variables. As a necessary preprocessing step, a body part-based feature representation approach is also proposed.

Chapter 5

Facial Makeup and Hairstyle Recommendation and Synthesis

In this chapter, Beauty e-Experts [101, 102], a fully automatic system for hairstyle and facial makeup recommendation and synthesis, is introduced. Given a user-provided frontal face image with short/bound hair and no/light makeup, the Beauty e-Experts system can not only recommend the most suitable hairdo and makeup, but also show the synthetic effects.

5.1 Introduction

Beauty is a language, which enables people to express their personalities, gain self-confidence and open up to others. Everybody wants to be beautiful. Hairstyle and makeup are two main factors that influence one's judgment about whether someone looks beautiful or not, especially for female. By choosing proper hair and makeup style, one can enhance the whole temperament and thus look more attractive. However, people often encounter problems when they make choices. First of all, the effects of different makeup products and hairstyles vary for different individuals, and are highly correlated with one's facial traits, *e.g.* face shape, skin color, etc. It is quite difficult, or even unlikely, for people to analyze their own facial features and make proper choices of hair and makeup care & dressing products.

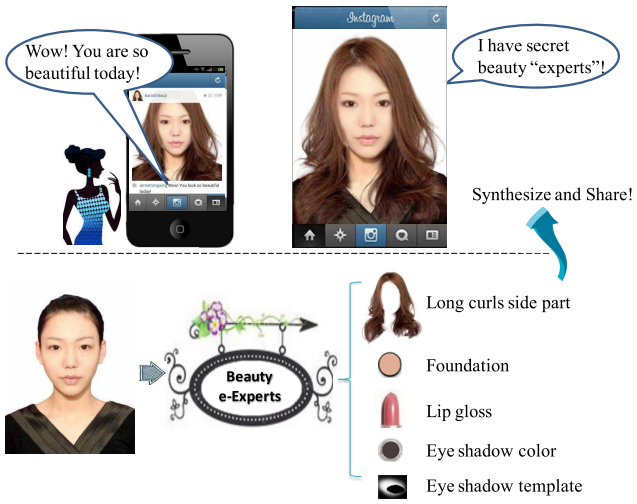


Figure 5.1: Overall illustration of the proposed Beauty e-Experts system. Based on the user's facial and clothing characteristics, our Beauty e-Experts system automatically recommends the suitable hairstyle and makeup products for the user, and then produces the synthesized visual effects. All the figures in this paper are best viewed in original color PDF file. Please resize $\times 2$ for better visual effects.

Second, nowadays cosmetics have developed into a large industry and there exist an unimaginable variety of products. Making choices in such a great variety can cost people a lot of time and money. Therefore, how to choose the proper hairstyle & makeup rapidly becomes a challenge.

In order to solve this problem, people have made some attempts. Some virtual hairstyle & makeup techniques have been developed. Softwares like Virtual Haircut & Makeover[1], which allow people to change hairstyle and perform virtual makeup on their photos, have been put into use. With software of this kind, users can input a facial image, and then choose any hairstyle or makeup they prefer from the options provided by the system. Users can see the effects of applying the chosen hairstyles and also makeup products on their faces, and make decisions on whether to choose these hairstyles or makeup products in reality. But there exists a problem of too much manual work for these softwares. For example, users have to mark out special regions, such as corners of eyes, nose, mouth or even pupil, etc., on their photos manually. Besides, these softwares do not have recommendation function.

[1]http://www.goodhousekeeping.com/beauty

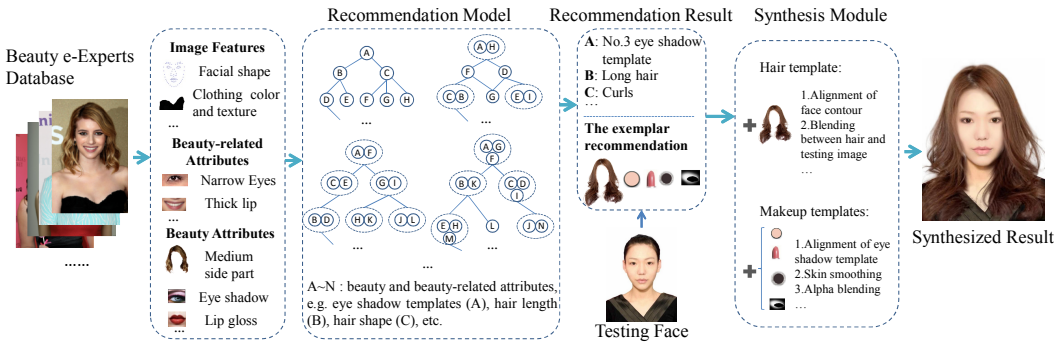


Figure 5.2: System processing flowchart. We firstly collect the Beauty e-Experts Database of 1,505 facial images with different hairstyles and makeup effects. With the extracted facial and clothing features, we propose a multiple tree-structured super-graphs model to express the complex relationships among beauty and beauty-related attributes. Here, the results from multiple individual super-graphs are fused based on voting strategy. In the testing stage, the recommended hair and makeup templates for the testing face are then applied to synthesize the final visual effects.

Users have to make choices on their own, and adjust the synthetic effects of these choices manually. It is too complicated for unprofessional people to accomplish such a process.

The research is still quite limited in this field, although some researchers have tried some approaches. Tong *et al.* [136] extracted makeup from before-and-after training image pairs, and transferred the makeup effect defined as ratios to a new testing image. Guo and Sim [55] considered makeup effect existing in two layers of the three-layer facial decomposition result, and the makeup effect of a reference image was transferred to the target image. Some patents also try to address the hair suggesting problem, *e.g.* [114], which searches for hairstyles in a database from the plurality of the hairstyle parameters based on manually selected hair attributes by the user. These works all fail to provide a recommendation function, and the synthetic effects may not be suitable for every part of the face. Besides, to our best knowledge, most of these works need a lot of user interactions and the final results are highly dependent on the efforts of users.

The aim of this work is to develop a novel Beauty e-Experts system, which helps users to select hairstyle and makeups automatically and produces the synthesized visual effects as shown in Figure 5.1. The main challenge in this problem is how to model the complex relationships among different beauty and beauty-related

attributes for reliable recommendation and natural synthesis. To address this challenge, we build a large dataset, Beauty e-Experts Dataset, which contains $1,505$ images of beautiful female figures selected from professional fashion websites. Based on this Beauty e-Experts Dataset, we first annotate all the beauty attributes for each image in the whole dataset. These beauty attributes, including different hairstyles and makeup types, are all adjustable in the daily life. Their specific combination is considered as the recommendation objective of the Beauty e-Experts System. To narrow the gap between the high-level beauty attributes and the low-level image features, a set of mid-level beauty-related attributes, such as facial traits and clothing properties, are also annotated for the dataset.

Based on all these attributes, we propose to learn a multiple tree-structured super-graphs model to explore the complex relationships among these attributes. As a generalization of a graph, a super-graph can theoretically characterize any type of relationships among different attributes and thus provide powerful recommendation. We propose to use its multiple tree-structured approximations to reserve the most important relationships and make the inference procedure tractable. Based on the recommended results, an effective and efficient facial image synthesis module is designed to seamlessly synthesize the recommended results into the user facial image and show it back to the user. Experimental results on 100 testing images show that our system can obtain very reasonable recommendation results and appealing synthesis results. The whole system processing flowchart is illustrated in Figure 5.2.

The contributions of this work are summarized as follows:

- A comprehensive system considering both hairstyle and makeup recommendation and synthesis is explored for the first time.
- A large database called Beauty e-Experts Database is constructed, including $1,505$ facial images with various hairstyles and makeup effects, and fully annotated with different types of attributes.
- A multiple tree-structured super-graphs model is proposed for hairstyle and makeup recommendation.

Figure 5.3: Some exemplar images from the Beauty e-Experts Dataset and the additional testing set. The left three images are from the Beauty e-Experts Dataset used for training, while the right two images are from the testing set.

5.2 Dataset, Attributes and Features

Hairstyle & makeup products make a lucrative market among female customers, but no public datasets for academic research exist. Most previous research [122, 55, 136] only works for a few samples. Chen and Zhang [22] released a benchmark for facial beauty study, but their focus is geometric facial beauty, not facial makeup and hairstyle. Wang and Ai [140] sampled $1,021$ images with regular hairstyles from Labeled Faces in the Wild (LFW) Database [65], which is not designed for hairstyle analysis. In addition, the sampled LFW database contains only a few hairstyles, since it is designed only for hair segmentation. In order to obtain enough knowledge to perform beauty modeling, a large dataset specifically designed is needed for this task. In the following, we will describe the construction of the Beauty e-Experts Dataset, as well as its attribute annotation and feature extraction process.

5.2.1 The Beauty e-Experts Dataset

To build our Beauty e-Experts Dataset, we have downloaded $\sim 800K$ images of female figures from professional hairstyle and makeup websites (e.g. www.stylebistro.com). We search these photos with the key words like *makeup, cosmetics, hairstyle, haircut and celebrity*. The initial downloaded images are screened by a commercial face analyzer[2] to remove images with no face detected, and then 87 keypoints are located

[2]OMRON OKAO Vision: `http://www.omron.com/r_d/coretech/vision/okao.html`

Table 5.1: A list of the high-level beauty attributes.

Name	**Values**
hair length	long, medium, short
hair shape	straight, curled, wavy
hair bangs	full, slanting, center part, side part
hair volume	dense, normal
hair color	20 classes
foundation	15 classes
lip gloss	15 classes
eye shadow color	15 classes
eye shadow template	20 classes

for each image. The images with high resolution and confident landmark detection results are retained. The retained images are further manually selected, and only those containing female figures who are considered as attractive and with complete hairstyle and obvious makeup effects are retained. The final $1,505$ retained images contain female figures in distinct fashions and with clear frontal faces, and are thus very good representatives for beauty modeling. Besides, we also collect 100 face images with short/bound hair and no/light makeup, which better demonstrate the synthesized results, for experimental evaluation purpose. Figure 5.3 shows some exemplar images in the Dataset.

5.2.2 Attributes and Features

To obtain beauty knowledge from the built dataset, we comprehensively explore different beauty attributes on these images, including various kinds of hairstyles and facial makeups. All these beauty attributes can be easily adjusted and modified in the daily life and thus have practical meaning for our beauty recommendation and synthesis system. We carefully organize these beauty attributes and set their attribute values based on some basic observations or preprocessing on the whole dataset. Table 5.1 lists the names and values of all the beauty attributes considered in the work. For the first four beauty attributes in Table 5.1, their values are set intuitively, and for the last five ones, their values are obtained by running the k-means clustering algorithm on the training dataset for the corresponding features

(the cluster number is determined empirically according to each specific attribute). The pixel values within the specific facial regions on each training image are clustered by Gaussian mixture models (GMM) in RGB color space. The centers of the largest GMM components are used as the representative colors. Then the colors are clustered by k-means to obtain the color attributes of hair and makeup templates. Hair templates and eye shadows are extracted by image matting [82]. For eye shadows, only the alpha channel is retained and further clustered to learn the eye shadow template attribute. The left eye shadows are sufficient for clustering. We show the visual examples of specific attribute values for some beauty attributes in Figure 5.4.

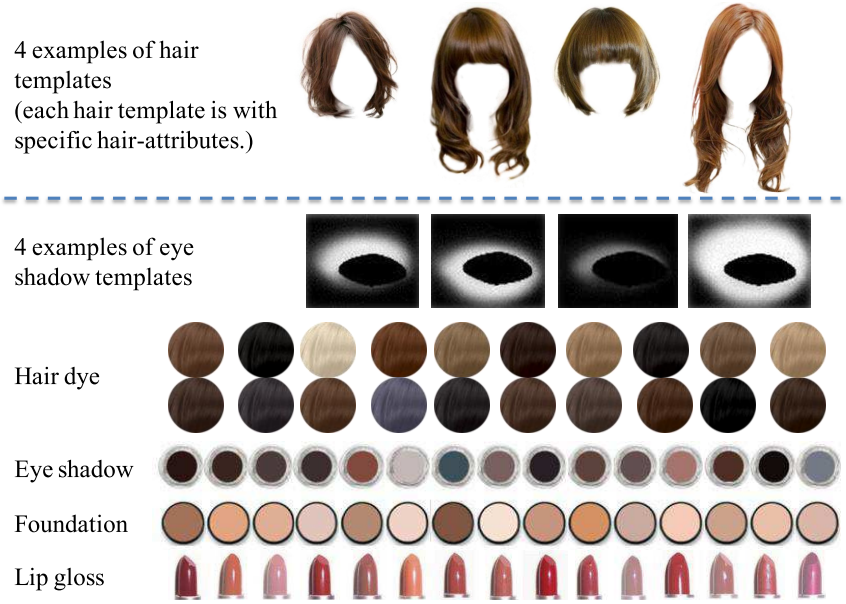


Figure 5.4: Visual examples of the specific values for some beauty attributes.

A straightforward way to predict the values of these high-level beauty attributes is using some low-level features to train some classifiers. However, since there is a relatively big gap between the high-level beauty attributes and the low-level image features, and the beauty attributes are intuitively related to some mid-level attributes like eye shape and mouth width, we further explore a set of mid-level beauty-related attributes to narrow the gap between the high-level beauty attributes and the low-level image features. Table 5.2 lists all the mid-level beauty-related attributes annotated for the dataset. These mid-level attributes mainly focus on the

facial shapes and clothing properties, which are kept fixed during the recommendation and the synthesis process.[3]

After the annotation of the high-level beauty attributes and mid-level beauty-related attributes, we further extract various types of the low-level image features on the clothing and facial regions for each image in the Beauty e-Experts Dataset to facilitate further beauty modeling. The clothing region of an image is automatically determined based on its geometrical relationship with the face region. Specifically, the following features are extracted for image representation:

- RGB color histogram and color moments on clothing region.
- Histograms of oriented gradients (HOG) [27] and local binary patterns (LBP) [4] features on clothing region.
- Active shape model [26] based shape parameters.
- Shape context [12] features extracted at facial points.

All the above features are concatenated to form a feature vector of $7,109$ dimensions, and then Principal Component Analysis (PCA) [70] is performed to reserve 90% of the energy. The compacted feature vector with 173 dimensions and the annotated attribute values are then fed into an SVM classifier to train for each attribute a classifier.

5.3 The Recommendation Model

A training beauty image is denoted as a tuple $(\langle \mathbf{x}, \mathbf{a}^r \rangle, \mathbf{a}^b)$. Here $\mathbf{x}$ is the image features extracted from the raw image data; $\mathbf{a}^r$ and $\mathbf{a}^b$ denote the set of the beauty-related attributes and beauty attributes respectively. Each attribute may have multiple different values, *i.e.* $a_i \in \{1, \ldots, n_i\}$, where n_i is the number of attribute values for the i-th attribute. The beauty-related attributes $\mathbf{a}^r$ act as the mid-level cues to narrow the gap between the low-level image features $\mathbf{x}$ and the high-level beauty attributes $\mathbf{a}^b$. The recommendation model needs to uncover the complex relation-

[3] Although the clothes of a user can be changed to make one look more beautiful, they are kept fixed in our current Beauty e-Experts system.

Table 5.2: A list of mid-level beauty-related attributes considered in this work.

Names	**Values**
forehead	high, normal, low
eyebrow	thick, thin
eyebrow length	long, short
eye corner	upcurved, downcurved, normal
eye shape	narrow, normal
ocular distance	hypertelorism, normal, hypotelorism
cheek bone	high, normal
nose bridge	prominent, flat
nose tip	wide, narrow
mouth opened	yes, no
mouth width	wide, normal
smiling	smiling, neutral
lip thickness	thick, normal
fatness	fat, normal
jaw shape	round, flat, pointed
face shape	long, oval, round
collar shape	strapless, v-shape, one-shoulder, high-necked, round, shirt collar
clothing pattern	vertical, plaid, horizontal, drawing, plain, floral print
clothing material	cotton, chiffon, silk, woolen, denim, leather, lace
clothing color	red, orange, brown, purple, yellow, green, gray, black, blue, white, pink, multi-color
race	Asian, Western

ships among the low-level image features, mid-level beauty-related attributes and high-level beauty attributes, and make the final recommendation for a given image.

5.3.1 Model Formulation

We model the relationships among the low-level image features, the mid-level beauty-related attributes, and the high-level beauty attributes from a probabilistic perspective. The aim of the recommendation system is to estimate the probability of beauty attributes, together with beauty-related attributes, given the image features, *i.e.* $p\left(\mathbf{a}^b, \mathbf{a}^r | \mathbf{x}\right)$, which can be modeled using the Gibbs distribution,

$$p\left(\mathbf{a}^b, \mathbf{a}^r | \mathbf{x}\right) = \frac{1}{Z\left(\mathbf{x}\right)} \exp\left(-E\left(\mathbf{a}^b, \mathbf{a}^r, \mathbf{x}\right)\right), \tag{5.1}$$

where $Z(\mathbf{x}) = \sum_{\mathbf{a}^b, \mathbf{a}^r} \exp\left(-E\left(\mathbf{a}^b, \mathbf{a}^r, \mathbf{x}\right)\right)$, also known as the partition function, is a normalizing term dependent on the image features, and $E(\mathbf{a}^b, \mathbf{a}^r, \mathbf{x})$ is an energy function measuring the compatibility among the beauty attributes, beauty-related attributes and image features. The beauty recommendation results can be obtained by finding the most likely joint beauty attribute state $\hat{\mathbf{a}}^b = \arg\max_{\mathbf{a}^b} \max_{\mathbf{a}^r} p\left(\mathbf{a}^b, \mathbf{a}^r | \mathbf{x}\right)$.

The capacity of this probabilistic model fully depends on the structure of the energy function $E(\mathbf{a}^b, \mathbf{a}^r, \mathbf{x})$. Here we propose to learn a general super-graph structure to build the energy function which can theoretically be used to model any relationships among the low-level image features, mid-level beauty-related attributes, and high-level beauty attributes. To begin with, we give the definition of super-graph.

Definition 1. *Super-graph: a super-graph $\mathcal{G}$ is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is called super-vertices, consisting a set of non-empty subsets of a basic node set, and $\mathcal{E}$ is called super-edges, consisting of a set of two-tuples, each of which contains two different elements in $\mathcal{V}$.*

It can be seen that a super-graph is actually a generalization of a graph in which a vertex can have multiple basic nodes and an edge can connect any number of basic nodes. When all the super-vertices only contain one basic node, and each super-edge is forced to connect to only two basic nodes, the super-graph then becomes a traditional graph. A super-graph can be naturally used to model the complex relationships among multiple factors, where the factors are denoted by the vertices and the relationships are represented by the super-edges.

Definition 2. *k-order super-graph: for a super-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, if the maximal number of vertices involved by one super-edge in $\mathcal{E}$ is k, $\mathcal{G}$ is said to be a k-order super-graph.*

Based on the above definitions, we propose to use the super-graph to characterize the complex relationships among the low-level image features, mid-level beauty-related attributes, and high-level beauty attributes in our problem. For example, the aforementioned pairwise correlations can be sufficiently represented by a 2-order super-graph (traditional graph), while other more complex relationships, such as

one-to-two and two-to-two relationships, can be represented by other higher order super-graphs. Denote the basic node set A as the union of the beauty attributes and beauty-related attributes, *i.e.* $A = \{a_i | a_i \in \mathbf{a}^r \cup \mathbf{a}^b\}$. Supposing the underlying relationships among all the attributes are represented by a super-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{\mathbf{a}_i | \mathbf{a}_i \subset A\}$[4] is a set of non-empty subsets of A, and $\mathcal{E}$ is the super-edge set that models their relationships, the energy function can then be defined as,

$$E\left(\mathbf{a}^b, \mathbf{a}^r, \mathbf{x}\right) = \sum_{\mathbf{a}_i \in \mathcal{V}} \phi_i\left(\mathbf{a}_i, \mathbf{x}\right) + \sum_{(\mathbf{a}_i, \mathbf{a}_j) \in \mathcal{E}} \phi_{ij}\left(\mathbf{a}_i, \mathbf{a}_j\right). \tag{5.2}$$

The first summation term is called FA (feature to attribute) potential which is used to model the relationships between the attributes and low-level image features, and the second one is called AA (attribute to attribute) potential and is used to model the complex relationships among different attributes represented by the super-edges. $\phi_i\left(\mathbf{a}_i, \mathbf{x}\right)$ and $\phi_{ij}\left(\mathbf{a}_i, \mathbf{a}_j\right)$ are the potential functions of the corresponding inputs, which can be learned in different ways. Generally, the FA potential $\phi_i\left(\mathbf{a}_i, \mathbf{x}\right)$ is usually modeled as a generalized linear function in the form like

$$\phi_i\left(\mathbf{a}_i = \mathbf{s}_i, \mathbf{x}\right) = \psi_{\mathbf{a}_i}\left(\mathbf{x}\right)^\top \mathbf{w}_i^{\mathbf{s}_i}, \tag{5.3}$$

where $\mathbf{s}_i$ is the values for attribute subset $\mathbf{a}_i$, $\psi_{\mathbf{a}_i}\left(\mathbf{x}\right)$ is a set of feature mapping functions for the attributes in $\mathbf{a}_i$ by using SVM on the extracted features (see Section 5.2.2), and $\mathbf{w}_i$ is the FA weight parameters to be learned for the model. And the AA potential function $\phi_i\left(\mathbf{a}_i, \mathbf{a}_j\right)$ is defined by a scalar parameter for each joint state of the corresponding super-edge,

$$\phi_{ij}\left(\mathbf{a}_i = \mathbf{s}_i, \mathbf{a}_j = \mathbf{s}_j\right) = w_{i,j}^{\mathbf{s}_i \mathbf{s}_j}, \tag{5.4}$$

where $w_{i,j}^{\mathbf{s}_i \mathbf{s}_j}$ is a scalar parameter for the corresponding joint state of $\mathbf{a}_i$ and $\mathbf{a}_j$ with the specific value $\mathbf{s}_i$ and $\mathbf{s}_j$.

[4]Note that in this paper we use $\mathbf{a}_i$ to denote a non-empty attribute set and a_i to denote a single attribute.

5.3.2 Model Learning

The learning of the super-graph based energy function includes learning the structure of the underlying super-graph and the parameters in the potential functions.

Structure Learning

Learning a fully connected super-graph structure is generally an NP-complete problem, which makes finding the optimal solution technically intractable [72]. However, we can still find many good approximations which can model a very large proportion of all the possible relationships. Among all the possible approximations, tree structure provides a very good choice which can be learned efficiently using many algorithms [72]. Another merit of tree structure is that the inference on a tree can be efficiently performed using methods like dynamic programming. Based on these considerations, we therefore use the tree-structured super-graph to model the underlying relationships. To remedy the information loss during the approximation procedure, we further propose to simultaneously learn multiple different tree-structured super-graphs to collaboratively model the objective relationships. Learning multiple tree-structured super-graphs is also supposed to produce more useful recommendation results, since it is intuitively similar to daily recommendation scenario. These tree-structured super-graphs can be supposed to be different recommendation experts, each of which is good at modeling some kinds of relationships. The recommendation results generated by these experts are voted to form the final recommendation result.

For a 2-order super-graph, learning a tree-structured approximation can be efficiently solved using the maximum spanning tree algorithm [25]. The edge weights in the graph are given by the mutual information between the attributes, which can be estimated from the empirical distribution from the annotations in the training data. For higher order super-graph, however, learning its tree-structured approximation will not be a trivial task, since the choices of vertex subsets for each super-edge are combinatorial.

Algorithm 3 Candidate set of subsets generation for super-graph structure learning.

Input: basic node set $A = \{a_1, \ldots, a_M\}$, adjacency matrix $W = \{w_{ij}\}_{1 \le i,j \le M}$, desired order of the super-graph k.

Output: candidate set of subsets $\mathcal{V} = \{\mathbf{a}_i | \mathbf{a}_i \in A\}$.

Initialization: set $\mathcal{V}$ with m unique subsets randomly collected from A, each of which has no more than $\lfloor (k+1)/2 \rfloor$ elements. Set $w_{\max} = f(\mathcal{V}, W)$.
while not converged **do**
  for $i = 1 \to M$ **do**
    for $j = 1 \to m$ **do**
      $w_j = f(\mathcal{V}, W)$ if move a_i to $\mathbf{a}_j$.
    end for
    $w_l = \text{argmax}_j(\{w_j\})$.
    if $l > w_{\max}$ **then**
      Move a_i to $\mathbf{a}_l$, $w_{\max} = w_l$.
      if $|\mathbf{a}_l| > \lfloor (k+1)/2 \rfloor$ **then**
        Split $|\mathbf{a}_l|$ into two subsets.
        $m \leftarrow m + 1$.
      end if
      if $m > \lceil 2 \times M/(k-1) \rceil$ **then**
        Merge two smallest subsets.
        $m \leftarrow m - 1$.
      end if
    end if
  end for
end while
Generate candidate vertex subsets.

Suppose for a super-graph built on basic node set $A = \{a_1, \ldots, a_M\}$ with M elements, we need to find a k-order tree-structured super-graph for these vertices. We first calculate the mutual information between each pair of vertices, and denote the results in the adjacency matrix form, i.e. $W = \{w_{ij}\}_{1 \le i,j \le M}$. Then we propose a two-stage algorithm to find the k-order tree-structured super-graph. Here mutual information is defined as $I(a_i, a_j) = \sum_{x \in a_i} \sum_{y \in a_j} p(x,y) log(\frac{p(x,y)}{p(x)p(y)})$.

In the first stage, we aim to find the candidate set of basic node subsets $\mathcal{V} = \{\mathbf{a}_i | \mathbf{a}_i \in \mathcal{V}\}$, which will be used to form the super-edges. The objective here is to find the set of subsets that has the largest amount of total mutual information in the resultant k-order super-graph. Here we first define a function that calculates

the mutual information of a subset set with a specified mutual information matrix,

$$f(\mathcal{V}, W) = \sum_{|\mathbf{a}_i| \geq 2} \sum_{a_j, a_k \in \mathbf{a}_i} w_{jk}. \tag{5.5}$$

Based on this definition, we formulate the candidate set generation problem as the following optimization problem

$$\begin{aligned} \underset{\mathcal{V}}{\text{argmax}} \quad & f(\mathcal{V}, W), \\ \text{s.t. } & |\mathbf{a}_i| \leq \lfloor \frac{k+1}{2} \rfloor, \forall i, \\ & |\mathcal{V}| \leq m, \end{aligned} \tag{5.6}$$

where the first inequality is from the k-order constraint from the resultant super-graph, $\lfloor \cdot \rfloor$ is the floor operator, and the parameter m in the second inequality is used to ensure that the generated subsets have a reasonable size to cover all the vertices and make the inference on the resultant super-graph more efficient. Specifically, its value can be set as

$$m = \begin{cases} M, & k = 2, \\ 2\lceil M/(k-1) \rceil, & \text{otherwise}, \end{cases} \tag{5.7}$$

where $\lceil \cdot \rceil$ is the ceiling operator. To solve this optimization problem, a k-means like iterative optimization algorithm is designed to find the solution. The algorithm first initializes some random vertex subsets and then re-assigns each vertex to the subsets that result in maximal mutual information increment; if one vertex subset has more than $\lfloor (k+1)/2 \rfloor$ elements, it will be split into two subsets; if the total cardinality of the vertex subset set is larger than $2\lceil M/(k-1) \rceil$, two subsets with the smallest cardinalities will be merged into one subset. This procedure is repeated until convergence. Alg. 3 gives the pseudo-code description of this procedure.

Based on the candidate vertex subsets, the second stage of the two-stage algorithm first calculates the mutual information between the element pair that satisfies the order restrictions in the each vertex subset. Then it builds a graph by using the calculated mutual information as adjacency matrix, and the maximum spanning

Algorithm 4 Learning multiple tree-structured super-graphs.

Input: basic node set $A = \{a_1, \ldots, a_M\}$, adjacency matrix $W = \{w_{ij}\}_{1 \leq i,j \leq M}$, number of desired super-graphs T.

Output: T tree-structured super-graphs $\mathbf{G} = \{\mathcal{G}_t\}_{t=1}^{T}$.

1: **Initialization**: set $\mathbf{G} = \emptyset$, $K = 5$.
2: **for** $t = 1 \rightarrow T$ **do**
3: Generate a random variable $k \in \{2, \ldots K\}$.
4: Obtain a candidate vertex subsets $\mathcal{V}$ using Alg. 3.
5: Calculate the mutual information between the elements pair with no more than k vertices in $\mathcal{V}$.
6: Make a graph using the calculated mutual information as adjacency matrix.
7: Find its maximal spanning tree using the algorithm in [25].
8: Form the k-order tree-structured super-graph $\mathcal{G}_t$.
9: $\mathbf{G} \leftarrow \mathbf{G} \cup \{\mathcal{G}_t\}$.
10: **end for**
11: Generate tree-structured super-graph set $\mathbf{G}$.

tree algorithm [25] is adopted to find its tree-structured approximation.

The above two-stage algorithm is run many times by setting different k values and initializations of subsets, which then generates multiple tree-structured super-graphs with different orders and structures. In order to make the parameters learning in the following tractable, the maximal k-value K is set to be equal to 5. The detailed description of this process is summarized in Alg. 4.

Parameter Learning And Inference

For each particular tree-structured super-graph, its parameter set, including the parameters in the FA potentials and the AA potentials, can be denoted in a whole as $\mathbf{\Theta} = \{\mathbf{w}_i^{\mathbf{s}_i}, w_{ij}^{\mathbf{s}_i\mathbf{s}_j}\}$. We adopt the maximal likelihood criterion to learn these parameters. Given N i.i.d. training samples $\mathbf{X} = \{\langle \mathbf{x}_n, \mathbf{a}_n^r \rangle, \mathbf{a}_n^b\}$, we need to minimize the loss function

$$\begin{aligned} \mathcal{L} &= \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}_n + \frac{1}{2}\lambda\sum_{i,\mathbf{s}_i}\|\mathbf{w}_i^{\mathbf{s}_i}\|_2^2 \\ &= \frac{1}{N}\sum_{n=1}^{N}\left\{-\ln p\left(\mathbf{a}_n^b, \mathbf{a}_n^r|\mathbf{x}_n\right)\right\} + \frac{1}{2}\lambda\sum_{i,\mathbf{s}_i}\|\mathbf{w}_i^{\mathbf{s}_i}\|_2^2, \end{aligned} \tag{5.8}$$

where λ is the tradeoff parameter between the regularization term and log-likelihood and its value is chosen by k-fold validation on the training set. Since the energy function is linear with respect to the parameters, the log-likelihood function is concave and the parameters can be optimized using gradient based methods. The gradient of the parameters can be computed by calculating their marginal distributions [109]. Denoting the value of attribute $\mathbf{a}_i$ for training image n as $\hat{\mathbf{a}}_i$, we have

$$\frac{\partial \mathcal{L}_n}{\partial \mathbf{w}_i^{\mathbf{s}_i}} = \left([\hat{\mathbf{a}}_i = \mathbf{s}_i] - p\left(\mathbf{a}_i = \mathbf{s}_i | \mathbf{x}_n\right) \right) \psi_{\mathbf{a}_i}\left(\mathbf{x}_n\right), \tag{5.9}$$

$$\frac{\partial \mathcal{L}_n}{\partial w_{ij}^{\mathbf{s}_i \mathbf{s}_j}} = [\hat{\mathbf{a}}_i = \mathbf{s}_i, \hat{\mathbf{a}}_j = \mathbf{s}_j] - p\left(\mathbf{a}_i = \mathbf{s}_i, \mathbf{a}_j = \mathbf{s}_j | \mathbf{x}_n\right), \tag{5.10}$$

where $[\cdot]$ is the Iverson bracket notation, i.e., $[\cdot]$ equals 1 if the experssion is true, and 0 otherwise.

Based on the calculation of the gradients, the parameters can be learned from different gradient based optimization algorithms [72]. In the experiments, we use the implementation by Schmidt[5] to learn these parameters. The learned parameters, together with the corresponding super-graph structures, form the final recommendation model.

Here each learned tree-structured super-graph model can be seen as a beauty expert. Given an input testing image, the system first extracts the feature vector $\mathbf{x}$, and then each beauty expert makes its recommendation by performing inference on the tree structure to find the maximum posteriori probability of $p\left(\mathbf{a}^b, \mathbf{a}^r | \mathbf{x}\right)$. The recommendation results output by all the Beauty e-Experts are then fused by majority voting to make the final recommendation to the user.

5.3.3 Relations with Other Models

The proposed multiple tree-structured super-graphs model characterizes the complex relationships among different attributes from a probabilistic prospective. When the maximal order value K is set to 2, our model degenerates into the classical

[5] `http://www.di.ens.fr/~mschmidt/Software/UGM.html`

graphical model used by most previous works [18, 140, 109, 157, 33, 156], where only the one-to-one pairwise correlations between two attributes are considered to model the complex relationships. Our model can generally model any order of the relationships. When the maximal order value K of the super-graph is set to 5, many other types of relationships, *e.g.*, one-to-two, two-to-two, and two-to-three, can be simultaneously modeled.

The pairwise correlations are also extensively modeled using the latent SVM model [144] from a deterministic perspective, which has been successfully applied into the problem like object detection [39], pose estimation [162], image classification [144], as well as clothing recommendation [98]. Compared with the latent SVM model, our tree-structured super-graph model not only can consider much more complex relationships among the attributes, but also is more efficient since tree structure makes both learning and inference process much faster. For a tree structured model with n nodes and k different values for each node, the time complexity of the inference process is only $O(k^2n)$, while a fully connected model (*e.g.* latent SVM) has the complexity of $O(k^n)$. Actually, during the training of the latent SVM model, some intuitively small correlations have to be removed manually to accelerate the training process. Our tree-structured super-graphs model, on the contrary, can automatically remove the small relationships during the structure learning process in a principled way. By extending to multiple tree-structured super-graphs, our model can produce much more reliable and useful recommendations as verified in experimental part, since it can well simulate the common recommendation scenario in our daily life, where one usually asks many people for recommendations and takes the majority or the most suitable one as the final choice.

5.4 The Synthesis Module

With the beauty attributes recommended by the multiple tree-structured super-graphs model, we further synthesize the final visual effect of hairstyle and makeup for the testing image. To this end, the system first uses beauty attributes to search for

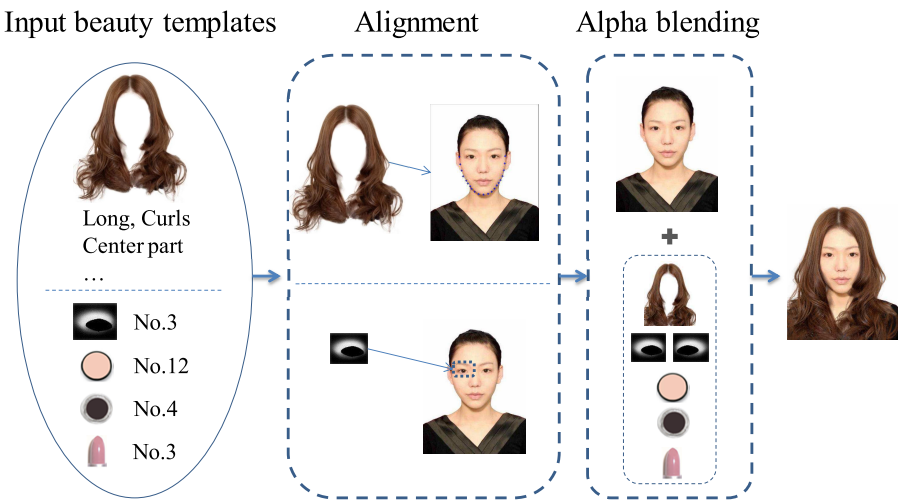


Figure 5.5: The flowchat of the synthesis module.

hair and makeup templates. A hair template is a combination of hairstyle attributes, such as long curls with bangs. We use the recommended hairstyle attributes to search the Beauty Expert Database for suitable hair templates. As mentioned in Section 5.2.2, each hair template is extracted from a training image. If more than one template is obtained, one from them is randomly selected. If we cannot find the hair template with exactly the same hairstyle attribute values, we use the one which has the values most approximating to the recommended hairstyle attribute values. Each makeup attribute forms a template which can be directly obtained from the dataset. These obtained hair and makeup templates are then fed into the synthesis process, which mainly has two steps: alignment and alpha blending, as shown in Figure 5.5.

In the alignment step, both of the hairstyle and the makeup templates need to be aligned with the testing image. For hair template alignment, a dual linear transformation procedure is proposed to put the hair template on the target face in the testing image. The dual linear transformation process first uses a linear affine transformation to perform rough alignment and then adopts a piecewise-linear affine transformation [49] to perform precise alignment. Figure 5.6 gives an illustration of this process. In the linear affine transformation, the 21 face contour points generated by the face analyzer are adopted to calculate affine transformation matrix between

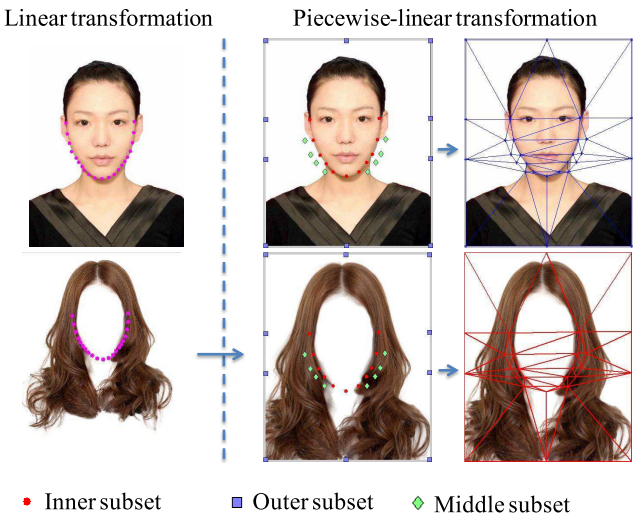


Figure 5.6: Hair template alignment process.

the hair template and the testing face. The hair template then can be roughly aligned to the testing face using transformation matrix. In piecewise linear affine transformation, three subsets of keypoints, namely, the inner subset, the middle subset, and the outer subset, are sampled based on the result of rough alignment. In Figure 5.6, these keypoints in the three subsets are drawn in red, green and blue respectively. 11 points are sampled interlacedly from 21 face contour points to consist the inner subset. The points in the inner subset are extended on the horizontal direction with half of the distance between two eye centers. They form the middle subset of 8 points. 10 points in the outer subset are fixed on the edge of the image. Their coordinates are determined by the image corners or the horizontal lines of eye centers and mouth corners. Note that points of the middle and outer subsets are at the same position in both the hair template and the testing image, which aims to keep the overall shape of the hairstyle. The total 29 points in the three subsets are then used to construct a Delaunay triangulation [29] to obtain 46 triangles. Then affine transformations are applied within the corresponding triangles between the testing face and the hair template. After that, these points on the hair template are precisely aligned with the testing face.

For the makeup templates alignment, only the eye shadow template need to be aligned to the eye region in the testing image. Other makeup templates can be

directly applied to the corresponding regions based on the face keypoint detection results. To align the eye shadow template to eye contour on the face, We use the thin plate spline method [16] to warp the eye shadow template by using the eye contour points. Because the eye shadow template attributes are learned by clustering from the left eye, the left template is mirrored to the right to obtain the right eye shadow template.

In the alpha blending step, the final result R is synthesized with hair template, makeup and the testing face I. The synthesis process is performed in CIELAB color space. L^* channel is considered as lightness because of its similarity to human visual perception. a^* and b^* are the color channels. We firstly use the edge-preserving operator on image lightness channel L^* to imitate the smoothing effect of foundation. We choose the guided filter [58], which is more efficient and has better performance near the edges among all the edge-preserving filters. It is applied to the L^* channel of facial region determined by facial contour points. Note that since we do not have contour points on the forehead, the forehead region is segmented out by GrabCut [120]. The final synthesis result is generated by alpha blending of the testing image I and hair and makeup template T in the L^*, a^* and b^* channels, respectively,

$$R = \alpha I + (1 - \alpha)T, \tag{5.11}$$

where α is a weight to balance I and T. For hair and eye shadow templates, the value of α is obtained from the templates themselves. For foundation and lip gloss, the α value is set to 0.5 for L^* channel, and 0.6 for a^* and b^* channels.

5.5 Experiments

In this section, we design experiments to evaluate the performance of the proposed Beauty e-Experts System from different aspects. We first visualize and analyze the intermediate result of model learning processing. Then the recommendation result is evaluated by comparison with several baselines, such as latent SVM [144],

multi-class SVM [20] and neural network [57]. The synthesis effects are finally presented and compared with some commercial systems related to hairstyle and makeup recommendation and synthesis.

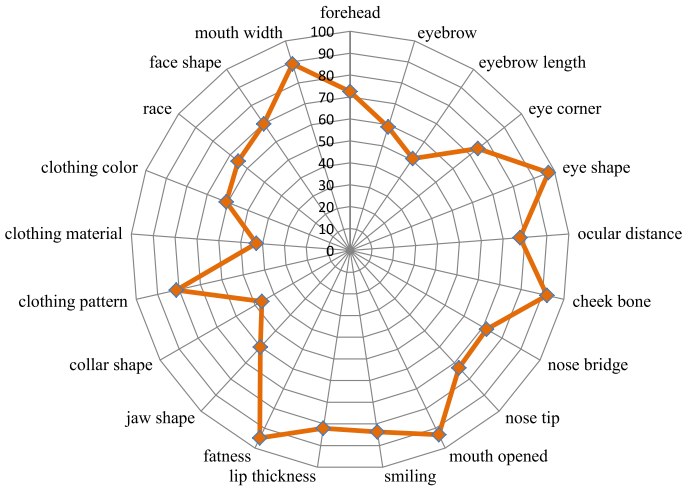


Figure 5.7: Accuracies of the predicted beauty attributes from the SVM classifier.

5.5.1 Model Learning and Analysis

We look into some intermediate aspects for a deep insight of the recommendation model structure and learning process. In Figure 5.7, we present the accuracy of predicted beauty-related attributes from the SVM classifier, which is used to build the FA potential function in the energy function to model the recommendation distribution (see Eqn. (5.2)). Note that we do not present the accuracy of beauty attributes, for it is not possible to obtain the ground truth label in this stage. It can be seen that most classifiers have the accuracies of more than 70%. It is sufficient to provide enough information to predict beauty attributes. Clothing-related attributes have the accuracy of $40\% \sim 50\%$, which is a little bit low. This is mainly caused by the large number of categories of clothing-related attributes.

We also visualize one example of the learned tree structure in the recommendation model in Figure 5.8. This tree structure is of order 4 and each super-vertex can only include two attributes at most (see Alg. 3). The weight of super-edge represents the mutual information between two related super-vertices. From the results, we can make some observations. Firstly, meaningful relationships are learned as shown in

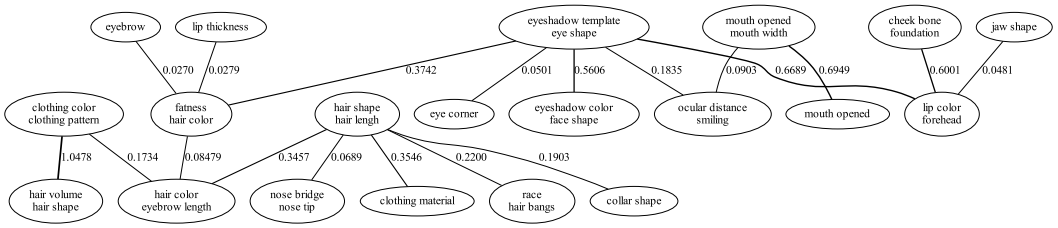


Figure 5.8: Visualization of one learned tree-structured super-graphs model.

the tree structure. The super-edges between super-vertex "hair shape, hair length" and other 5 super-vertex are retained, while the super-vertex "eye corner" only remains one super-edge with other super-vertex. It means "hair shape, hair length" is more important and has broader relationship with other nodes than "eye corner" in this structure. Secondly, some highly correlated attributes are clustered into one super-vertex, such as "hair shape" with "hair length" and "eye shadow template" with "face shape". It well fit to the intuitive perception of humans. Long hair may match well with curled hair, and certain shape of eye shadow template may also fit to some face shapes. Thirdly, the correlation between super-vertexs is represented on the super-edges. The super-vertex "eye shadow template, eye shape" has the weight 0.5606 with 'eye shadow color, face shape" , which is higher than the weight 0.0501 with "eye corner". It means "eye shadow template, eye shape" has stronger correlation with "eye shadow color, face shape".

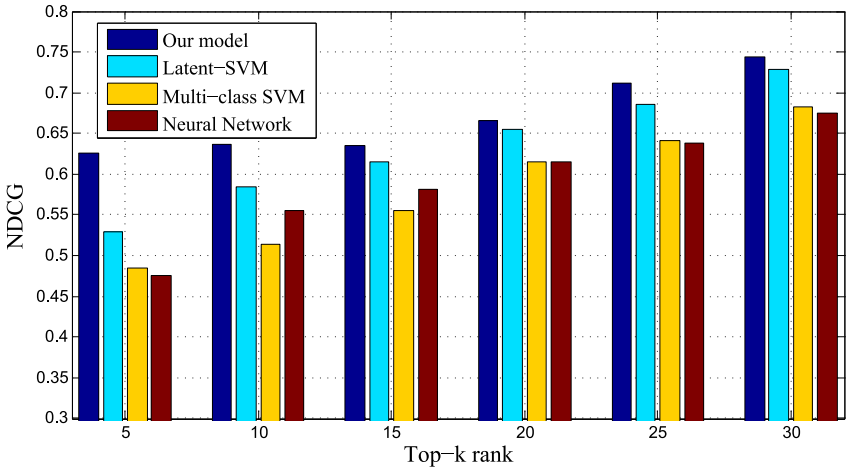


Figure 5.9: NDCG values of multiple tree-structured super-graphs model and three baselines. The horizontal axis is the rank of top-k results, while the vertical axis is the corresponding NDCG value. Our proposed method achieve better performance than the latent SVM model and other baselines.

5.5.2 Recommendation Results Evaluation

For the recommendation model in the Beauty e-Experts system, we also implement some alternatives using multi-class SVM, neutral network, and latent SVM. The first two baselines only use the low-level image features to train classifiers for high-level beauty attributes. The latent SVM baseline considers not only the low-level image features but also the pair-wise correlations between the beauty and the beauty-related attributes. We use the 100 testing images to evaluate the performance of the three baseline methods and our algorithm. To evaluate the recommendation result of the Beauty e-Experts system quantitatively, the human perception of suitable beauty makeups is considered as the ground truth measured on 50 random combinations of the attributes for all the 100 testing images. We asked 20 participants (staffs and students in our group) to label the ground truth of ranking results of the 50 types beauty makeup effects for each testing image. Instead of labeling absolute ranks from 1 to 50, we use a k-wise strategy similar to [98]: labelers are shown k images as a group each time, where k is set to 10. They only need to rank satisfying levels within each group. $C(k, 2)$ pairwise preferences can be obtained from the k ranks, and then the final rank is calculated across groups by ranking SVM [69].

In Figure 5.9, the comparison results of our model and other baselines is plotted. The performance is measured by Normalized Discounted Cumulative Gain (NDCG) [41], which is widely used to evaluate ranking systems. From the results, it can be observed that our model and latent SVM significantly outperform multi-class SVM and neural network. This is mainly because that our model and the latent SVM method are equipped with mid-level beauty-related attributes to narrow the semantic gap between low-level image features and the high-level beauty attributes. These two models are both able to characterize the co-occurrence information to mine the pairwise correlations between every two factors. From Figure 5.9 it can be further found that our model has overall better performance than the latent SVM method, especially in the top 15 recommendations. With higher order relationships embedded, our model can express more complex relationship among different attributes.

What is more, by employing multiple tree-structured super-graphs, our model tend to obtain more robust recommendation results.

5.5.3 Synthesis Results Evaluation

We compare our Beauty e-Experts system with some commercial virtual hairstyle and makeup systems, including Virtual Hairstyle (VH)[6], Instant Hair Makeover (IHM)[7], Daily Makeover (DM)[8], Virtual Makeup Tool (VMT)[9], and the virtual try-on website TAAZ[10]. They are all very popular in female customers on the Internet.

Table 5.3: Comparisons of several popular hairstyle and makeup synthesis systems.

	VH	IHM	DM	VMT	TAAZ	Ours
hairstyle	✓	✓	✓	×	✓	✓
makeup	×	✓	✓	✓	✓	✓
face detection	×	✓	✓	×	✓	✓
easy of use	×	×	✓	×	×	✓
500+ templates	×	×	×	×	✓	✓
composition freedom	×	✓	×	✓	✓	✓
recommendation	×	×	×	×	×	✓

These systems are firstly compared in an overview manner, which means that we focus on the comparison of the main functionalities among these systems. The comparison results are summarized in Table 5.3. It can be seen that IHM, DM and TAAZ systems can provide both hairstyle and makeup synthesis functions. They also provide face detection, which can largely reduce the manual workload. IHM, VMT and TAAZ ask users to choose makeup and hair products to perform composition, while VH and DM cannot support this, since their methods are mainly based on holistic transformation between the testing face and the example template. However, all these systems cannot support large data set with more than 500 templates and do not provide hairstyle and makeup recommendation functions. In the contrast, our Beauty e-Experts system can support all the functions mentioned above. What is more, it is fully automatic and can work in more general cases. The recommendation function of our system is really useful to help female users choose

[6] http://www.hairstyles.knowage.info
[7] http://www.realbeauty.com/hair/virtual/hairstyles
[8] http://www.dailymakeover.com/games-apps/games
[9] http://www.hairstyles.knowage.info
[10] http://www.taaz.com

suitable hairstyle and makeup products.

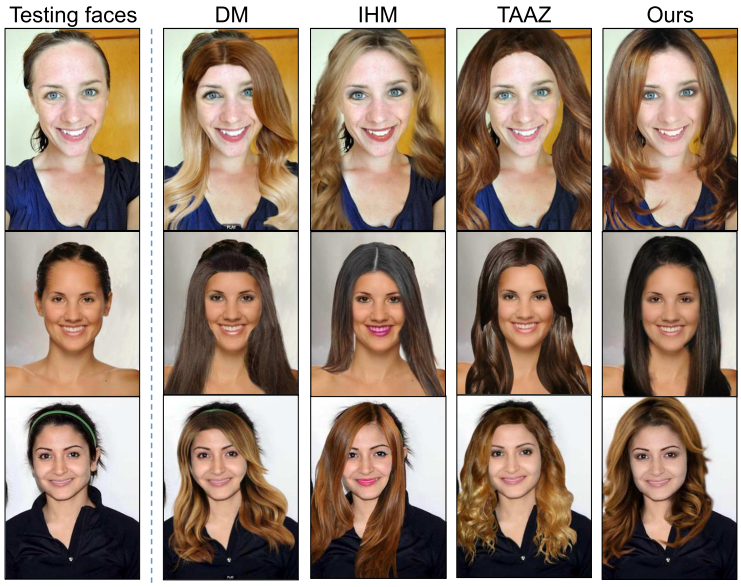


Figure 5.10: Contrast results of synthesized effect among websites and our paper.

We then compare the hairstyle and makeup synthesis results with these commercial systems. As shown in Figure 5.10, the first column is the testing images, and other four columns are the results generated by DM, IHM, TAAZ, and our system, respectively. The reason why these three systems are selected is that only these three can synthesize both the hairstyle and makeup effects. The makeup and hairstyle templates used in the synthesis process are selected with some user interactions to ensure that all the four methods share similar makeups and hairstyles. It can be seen that, even after some extra user interactions, the results generated from these three websites have obvious artifacts. The selected hair templates cannot cover the original hair area. IHM even cannot handle the mouth opened cases. We further present several testing faces with different hairstyle and makeup effects in Figure 5.11. These results still look natural with a variety of style changing, which demonstrates the robustness of our system.

5.6 Chapter Summary

In this work, the Beauty e-Experts system for automatic facial hairstyle and makeup recommendation and synthesis has been developed . To the best of our knowledge,

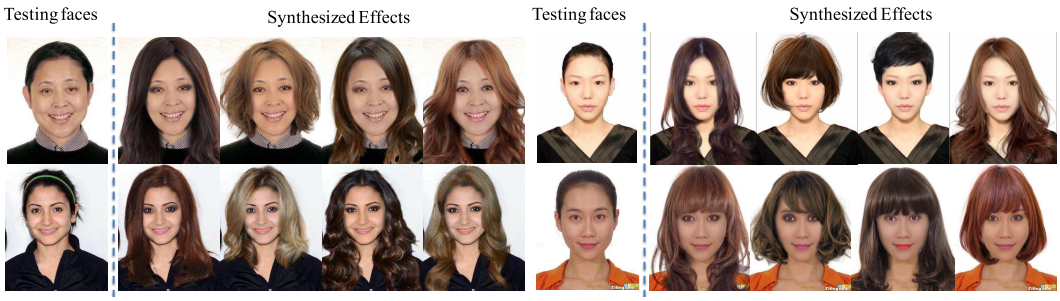


Figure 5.11: More synthesized results of the proposed Beauty e-Experts system.

it is the first study to investigate into a fully automatic hairstyle and makeup system that simultaneously deals with hairstyle and makeup recommendation and synthesis. Based on the proposed multiple tree-structured super-graphs model, our system can capture the complex relationships among the different attributes, and produce reliable and explainable recommendations results. The synthesis model in our system also produces natural and appealing results. Extensive experiments on a newly built dataset have verified the effectiveness of our recommendation and synthesis models.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

In this thesis, several datasets and algorithms to handle realistic human analytics are proposed. In the previous chapters, we have studied into several specific problems. The main content and contributions of the thesis are summarized as below.

In Chapter 2, Large Population Face identification in Unconstrained Videos. We investigate large-scale face identification in unconstrained videos with one thousand subjects. This problem is very challenging, and until now most studies have only considered the scenarios with a small number of subjects and videos captured in controlled laboratory environments. The contributions are in two folds. Firstly, we set up a large-scale video database in unconstrained environment, Celebrity-1000, with data collected from two popular video sharing websites, YouTube and Youku, for face identification research. It contains $1,000$ celebrities from different countries, $\sim 7,000$ videos, $\sim 160K$ tracking sequences and $\sim 2.4M$ sampled frames. Secondly, we boost the efficiency of Multi-Task Joint Sparse Representation (MTJSR) algorithm for video based face identification on Celebrity-1000. MTJSR is training-free and can naturally integrate multiple frames of the same tracking sequence for collaborative inference, and thus is suitable for video based face identification. We present a sparsity-induced scalable optimization method (SISO), which solves the large-scale MTJSR problem by sequentially solving a series of smaller-scale subproblems with

theoretically guaranteed convergency. Extensive experiments show several orders-of-magnitude speedup with this new optimization method, and also demonstrate the superiorities of the accelerated MTJSR algorithm over several popular baseline algorithms.

In Chapter 3, Deep Aging Face Recognition with Large Gaps. We first collect a so-called Cross-Age FacE (CAFE) dataset, with more than 900 celebrities and their faces with large age gaps, ranging from child, young, adult, to old groups. Then, we propose a novel framework, called Deep Aging Face Recognition (DAFR), for this challenging task. DAFR includes two modules, aging pattern synthesis and aging face verification. The aging pattern synthesis module synthesizes the faces of all age groups, namely an aging pattern, for the input face of an arbitrary age, and the core structure is a deep aging-aware denoising auto-encoder (a^2-DAE) with multiple outputs of different age groups. The aging face verification module then takes the synthesized aging patterns of a face pair as the input, and each pair of synthesized images of the same age group is fed into a parallel CNN, and finally all parallel CNN outputs are fused to provide similar/dissimilar prediction. For DAFR, the training of the aging face verification module easily suffers from the overfitting results from the aging pattern synthesis module, and we propose to use the cross-validation strategy to produce error-aware outputs for the synthesis module, which significantly enhances the learnability of the whole framework.

In Chapter 4, Clothing Attributes Assisted Person Re-identification. We present a comprehensive study on clothing attributes assisted person re-identification. First, the body parts and their local features are extracted for alleviating the pose-misalignment issue. A latent SVM based person re-identification approach is proposed to describe the relations among the low-level part features, middle-level clothing attributes, and high-level re-identification labels of person pairs. Motivated by the uncertainties of clothing attributes, we treat them as real-value variables instead of using them as discrete variables. Moreover, a large-scale real-world dataset with ten camera views and about 200 subjects is collected and thoroughly annotated for this study. The extensive experiments on this dataset show: 1) part features are more effective

than features extracted from the holistic human bounding boxes; 2) the clothing attributes embedded in the latent SVM model may further boost re-identification performance compared with SVM without clothing attributes; and 3) treating clothing attributes as real-value variables is more effective than using them as discrete variables in person re-identification.

In Chapter 5, Makeup and Hairstyle Recommendation and Synthesis. Beauty e-Experts, a fully automatic system for hairstyle and facial makeup recommendation and synthesis, is developed. Given a user-provided frontal face image with short/bound hair and no/light makeup, the Beauty e-Experts system can not only recommend the most suitable hairdo and makeup, but also show the synthetic effects. To obtain enough knowledge for beauty modeling, we build the Beauty e-Experts Database, which contains $1,505$ attractive female photos with a variety of beauty attributes and beauty-related attributes annotated. Based on this Beauty e-Experts Dataset, two problems are considered for the Beauty e-Experts system: what to recommend and how to wear, which describe a similar process of selecting hairstyle and cosmetics in our daily life. For the what-to-recommend problem, we propose a multiple tree-structured super-graphs model to explore the complex relationships among the high-level beauty attributes, mid-level beauty-related attributes and low-level image features, and then based on this model, the most compatible beauty attributes for a given facial image can be efficiently inferred. For the how-to-wear problem, an effective and efficient facial image synthesis module is designed to seamlessly synthesize the recommended hairstyle and makeup into the user facial image. Extensive experimental evaluations and analysis on testing images of various conditions well demonstrate the effectiveness of the proposed system.

6.2 Future Works

Although most challenges of realistic human analytics have been discussed and addressed in our systems, several limitations still exist.

- Firstly, the feature representations designed are not well generalized for some

problems. Traditional features, such as HOG, LBP and Gabor wavelets features, are used as feature representation in Chapter 2, 4 and 5. The low representation ability limits the final performance of our system.

- Secondly, the constructed datasets in Chapter 2, 3 and 4 are still far from enough to cover all possible variations in real world. Although datasets with millions of variables have been proposed, the number of identities in the datasets is still quite limited. How to effectively collect data is a quite tough problem, and better strategies need to be considered.

- Finally, the connection with social networks is still limited. Social networks, such as Facebook and Google plus, are full of valuable image/video resources and user data. If these resources can be proper crawled and utilized, we may get much reasonable recommendation, recognition and synthesis results.

Observing these limitations, we consider further research topics in the realistic human analytics:

- Firstly, we consider to utilize deep learning framework for end-to-end feature learning and recognition. In deep learning, feature representation is jointly optimized with classifiers and regressors. Thus, it is possible to learn a more effective feature than the handcraft features.

- Secondly, we consider to design a endless learning framework [90] for automatic mining useful image/video resources from the Internet. With a pre-trained weak model, the endless learning framework can automatically label the explored image/video resources and fine-tune to obtain much a better model. With this strategy, more meaningful data can be obtained to better tackle the problems in the real world.

- Finally, considering the vast information available in social networks we plan to add user profile into our recommendation and synthesis models. The user data, such as browsing history and personal preference, can be used as context information to achieve personalized recommendation and synthesis.

Bibliography

[1] `www.omron.com/r_d/coretech/vision/okao.html`.

[2] The fg-net aging database. *Face and gesture recognition working group*, 2000.

[3] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):2037–2041, 2006.

[4] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: application to face recognition. *TPAMI*, 2006.

[5] A. Albert, K. Ricanek, and E. Patterson. The aging adult skull and face: A review of the literature and report on factors and processes of change. *Published by UNCWTR01*, 2004.

[6] M. Aly, P. Welinder, M. Munich, and P. Perona. Scaling object recognition: Benchmark of current state of the art techniques. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 2117–2124. IEEE, 2009.

[7] A. Andoni and P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pages 459–468. IEEE, 2006.

[8] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines

for multiple-instance learning. In *Advances in neural information processing systems*, pages 561–568, 2002.

[9] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density divergence. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 581–588. IEEE, 2005.

[10] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski. Face recognition by independent component analysis. *Neural Networks, IEEE Transactions on*, 13(6):1450–1464, 2002.

[11] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *TPAMI*, 1997.

[12] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *TPAMI*, 2002.

[13] Y. Bengio. Learning deep architectures for ai. *Foundations and trends in Machine Learning*, 2009.

[14] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*, 2010.

[15] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999.

[16] F. Bookstein. Principal warps: thin-plate splines and the decomposition of deformations. *TPAMI*, 1989.

[17] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[18] Y. Boykov and O. Veksler. Fast approximate energy minimization via graph cuts. *TPAMI*, 2001.

[19] H. Cevikalp and B. Triggs. Face recognition based on image sets. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2567–2573. IEEE, 2010.

[20] C. Chang and C. Lin. Libsvm: A library for support vector machines. In *TIST*, 2011.

[21] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *CVPR*, 2013.

[22] F. Chen and D. Zhang. A benchmark for geometric facial beauty study. In *Int. Conf. Medical Biometrics*, 2010.

[23] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM journal on scientific computing*, 20(1):33–61, 1998.

[24] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE, 2005.

[25] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *TIT*, 1968.

[26] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models-their training and application. *CVIU*, 1995.

[27] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[28] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic

metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

[29] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, third edition, 2008.

[30] J. Dong, Q. Chen, J. Feng, K. Jia, Z. Huang, and S. Yan. Looking inside category: subcategory-aware object recognition. *Transactions on Circuits and Systems for Video Technology*, 2014.

[31] J. Dong, Q. Chen, Z. Huang, J. Yang, and S. Yan. Parsing based on parselets: A unified deformable mixture model for human parsing. *Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[32] J. Dong, Q. Chen, W. Xia, Z. Huang, and S. Yan. A deformable mixture parsing model with parselets. In *International Conference on Computer Vision*, 2013.

[33] J. Dong, Q. Chen, S. Yan, and A. Yuille. Towards unified object detection and segmentation. In *European Conference on Computer Vision*. Springer-Verlag, 2014.

[34] J. Dong, B. Cheng, X. Chen, T. Chua, S. Yan, and X. Zhou. Robust image annotation via simultaneous feature and sample outlier pursuit. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2013.

[35] J. Dong, W. Xia, Q. Chen, J. Feng, Z. Huang, and S. Yan. Subcategory-aware object classification. In *Conference on Computer Vision and Pattern Recognition*, 2013.

[36] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[37] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person Re-Identification by Symmetry-Driven Accumulation of Local Features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2360–2367, 2010.

[38] P. Felzenszwalb, R. Girshick, and D. McAllester. Cascade Object Detection with Deformable Part Models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2241–2248, 2010.

[39] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010.

[40] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object Detection with Discriminatively Trained Part Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[41] R. Feris and L. Davis. Image ranking and retrieval based on multi-attribute queries. In *CVPR*, 2011.

[42] V. Franc and S. Sonnenburg. Optimized cutting plane algorithm for support vector machines. In *Proceedings of the 25th international conference on Machine learning*, pages 320–327. ACM, 2008.

[43] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic press, 2013.

[44] X. Geng, Z.-H. Zhou, and K. Smith-Miles. Automatic age estimation based on facial aging patterns. *TPAMI*, 2007.

[45] N. Gheissari, T. Sebastian, and R. Hartley. Person Reidentification Using Spatiotemporal Appearance. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1528–1535, 2006.

[46] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *Advances in neural information processing systems*, pages 451–458, 2005.

[47] I. J. Goodfellow, D. Warde-Farley, P. Lamblin, V. Dumoulin, M. Mirza, R. Pascanu, J. Bergstra, F. Bastien, and Y. Bengio. Pylearn2: a machine learning research library. *arXiv preprint arXiv:1308.4214*, 2013.

[48] D. Gorodnichy. Video-based framework for face recognition in video. 2005.

[49] A. Goshtasby. Piecewise linear mapping functions for image registration. *PR*, 1986.

[50] D. Gray, S. Brennan, and H. Tao. Evaluating Appearance Models for Recognition, Reacquisition, and Tracking. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, 2007.

[51] D. Gray and H. Tao. Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. In *European Conference on Computer Vision*, pages 262–275, 2008.

[52] R. Gross. Face databases. In *Handbook of Face Recognition*, pages 301–327. Springer, 2005.

[53] R. Gross and J. Shi. The cmu motion of body (mobo) database. 2001.

[54] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 498–505. IEEE, 2009.

[55] D. Guo and T. Sim. Digital face makeup by example. In *CVPR*, 2009.

[56] F. Hausdorff. Dimension und äußeres Maß. *Mathematische Annalen*, 79(1-2):157–179, 1918.

[57] S. Haykin. *Neural Networks*. Prentice Hall, 1999.

[58] K. He, J. Sun, and X. Tang. Guided image filtering. In *ECCV*, 2010.

[59] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang. Face recognition using laplacianfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):328–340, 2005.

[60] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[61] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[62] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2001.

[63] M. Hirzer, P. Roth, M. Köstinger, and H. Bischof. Relaxed Pairwise Learned Metric for Person Re-identification. In *European Conference on Computer Vision*, pages 780–793, 2012.

[64] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*, 2012.

[65] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[66] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, 2007.

[67] J. Huang, W. Xia, and S. Yan. Deep search with attribute-aware deep network. In *ACM Multimedia*, Orlando, FL, USA, 2014.

[68] K. Jia, T.-H. Chan, and Y. Ma. Robust and practical face recognition via structured sparsity. In *Computer Vision–ECCV 2012*, pages 331–344. Springer, 2012.

[69] T. Joachims. Optimizing search engines using clickthrough data. In *ACM KDD*, 2002.

[70] I. Jolliffe. Principal component analysis. *Encyclopedia of Statistics in Behavioral Science*, 2002.

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

[72] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[73] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2288–2295. IEEE, 2012.

[74] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[75] R. Layne, T. Hospedales, and S. Gong. Person Re-identification by Attributes. In *British Machine Vision Conference*, volume 2, page 3, 2012.

[76] R. Layne, T. M. Hospedales, and S. Gong. Towards Person Identification and Re-identification with Attributes. In *Workshop of European Conference on Computer Vision*, pages 402–412, 2012.

[77] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2006.

[78] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–313. IEEE, 2003.

[79] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Visual tracking and recognition using probabilistic appearance manifolds. *Computer Vision and Image Understanding*, 99(3):303–331, 2005.

[80] K.-C. Lee and D. Kriegman. Online learning of probabilistic appearance manifolds for video-based recognition and tracking. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 852–859. IEEE, 2005.

[81] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 775–781. IEEE, 2005.

[82] A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. *TPAMI*, 2008.

[83] T. Leyvand, D. Cohen-Or, G. Dror, and D. Lischinski. Data-driven enhancement of facial attractiveness. *ACM Transactions on Graphics (TOG)*, 27(3):38, 2008.

[84] A. Li, L. Liu, K. Wang, S. Liu, and S. Yan. Clothing attributes assisted person re-identification. *Circuits and Systems for Video Technology, IEEE Transactions on*, 2014.

[85] A. Li, L. Liu, and S. Yan. Person re-identfication by attribute-assisted clothes appearance. In *Person Re-identification*, page 456. Springer, 2013.

[86] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang. Probabilistic elastic matching for pose variant face verification. In *CVPR*, 2013.

[87] P. Li, Y. Fu, U. Mohammed, J. H. Elder, and S. J. Prince. Probabilistic models for inference about identity. *TPAMI*, 2012.

[88] Z. Li, U. Park, and A. K. Jain. A discriminative model for age invariant face recognition. *TIFS*, 2011.

[89] X. Liang, S. Liu, X. Shen, J. Yang, L. Liu, L. Lin, and S. Yan. Deep human parsing with active template regression. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2015.

[90] X. Liang, S. Liu, Y. Wei, L. Liu, L. Lin, and S. Yan. Computational baby learning. *arXiv preprint arXiv:1411.2861*, 2014.

[91] H. Liu and S. Yan. Robust graph mode seeking by graph shift. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 671–678, 2010.

[92] J. Liu, B. Kuipers, and S. Savarese. Recognizing Human Actions by Attributes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3337–3344, 2011.

[93] L. Liu, Y. Wang, and T. Tan. Online appearance model learning for video-based face recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–7. IEEE, 2007.

[94] L. Liu, J. Xing, S. Liu, H. Xu, X. Zhou, and S. Yan. Wow! you are so beautiful today! *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 11(1s):20, 2014.

[95] L. Liu, C. Xiong, H. Zhang, Z. Niu, M. Wang, and S. Yan. Deep aging face verification with large gaps. 2015.

[96] L. Liu, H. Xu, J. Xing, S. Liu, X. Zhou, and S. Yan. "wow! you are so beautiful today!". In *Proceedings of the 21st ACM international conference on Multimedia*, pages 3–12. ACM, 2013.

[97] L. Liu, L. Zhang, H. Liu, and S. Yan. Toward large-population face identification in unconstrained videos. *Circuits and Systems for Video Technology, IEEE Transactions on*, 24(11):1874–1884, 2014.

[98] S. Liu, J. Feng, Z. Song, T. Zhang, H. Lu, C. Xu, and S. Yan. "Hi, magic closet, tell me what to wear". In *ACM MM*, 2012.

[99] S. Liu, X. Liang, L. Liu, K. Lu, L. Lin, and S. Yan. Fashion parsing with video context. In *Proceedings of the ACM International Conference on Multimedia*, pages 467–476. ACM, 2014.

[100] S. Liu, X. Liang, L. Liu, X. Shen, J. Yang, C. Xu, L. Lin, X. Cao, and S. Yan. Matching-cnn meets knn: Quasi-parametric human parsing. *arXiv preprint arXiv:1504.01220*, 2015.

[101] S. Liu, L. Liu, and S. Yan. Magic mirror: An intelligent fashion recommendation system. In *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*, pages 11–15. IEEE, 2013.

[102] S. Liu, L. Liu, and S. Yan. Fashion analysis: Current techniques and future directions. *MultiMedia, IEEE*, 21(2):72–79, 2014.

[103] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan. Street-to-Shop: Cross-Scenario Clothing Retrieval via Parts Alignment and Auxiliary Set. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3330–3337, 2012.

[104] X. Liu and T. Chen. Video-based face recognition using adaptive hidden markov models. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–340. IEEE, 2003.

[105] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[106] P. Luo, X. Wang, and X. Tang. Hierarchical face parsing via deep learning. In *CVPR*, 2012.

[107] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the ICML*, 2013.

[108] A. M. Martinez. The ar face database. *CVC Technical Report*, 24, 1998.

[109] T. Mensink, J. Verbeek, and G. Csurka. Tree-structured crf models for interactive image labeling. *TPAMI*, 2013.

[110] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *Second international conference on audio and video-based biometric person authentication*, volume 964, pages 965–966. Citeseer, 1999.

[111] H. Moon and P. J. Phillips. Computational and performance aspects of pca-based face-recognition algorithms. *Perception-London*, 30(3):303–322, 2001.

[112] Y. Mu, J. Dong, X. Yuan, and S. Yan. Accelerated low-rank visual recovery by random projection. In *Conference on Computer Vision and Pattern Recognition*, 2011.

[113] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2, 2009.

[114] Y. Nagai, K. Ushiro, Y. Matsunami, T. Hashimoto, and Y. Kojima. Hairstyle suggesting system, hairstyle suggesting method, and computer program product. 2005.

[115] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

[116] U. Park, Y. Tong, and A. K. Jain. Age-invariant face recognition. *TPAMI*, 2010.

[117] S. Perkins, K. Lacker, and J. Theiler. Grafting: Fast, incremental feature selection by gradient descent in function space. *The Journal of Machine Learning Research*, 3:1333–1356, 2003.

[118] B. Prosser, W. Zheng, S. Gong, T. Xiang, and Q. Mary. Person Re-identification by Support Vector Ranking. In *British Machine Vision Conference*, pages 21.1–21.11, 2010.

[119] K. Ricanek and T. Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *FGR*, 2006.

[120] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *TOG*, 2004.

[121] C. Sanderson. *Biometric person recognition: Face, speech and fusion.* VDM Publishing, 2008.

[122] K. Scherbaum, T. Ritschel, M. Hullin, T. Thormählen, V. Blanz, and H. Seidel. Computer-suggested facial makeup. *CGF*, 2011.

[123] W. R. Schwartz and L. S. Davis. Learning Discriminative Appearance-Based Models Using Partial Least Squares. In *XXII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 322–329, 2009.

[124] W. R. Schwartz, H. Guo, J. Choi, and L. S. Davis. Face identification using large feature sets. *Image Processing, IEEE Transactions on*, 21(4):2245–2255, 2012.

[125] G. Shakhnarovich, J. W. Fisher, and T. Darrell. Face recognition from long-term observations. In *Computer VisionECCV 2002*, pages 851–865. Springer, 2002.

[126] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(12):1615–1618, 2003.

[127] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman. Fisher vector faces in the wild. In *BMVC*, 2013.

[128] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014.

[129] Y. Sun, D. Liang, X. Wang, and X. Tang. Deepid3: Face recognition with very deep neural networks. *arXiv preprint arXiv:1502.00873*, 2015.

[130] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, 2014.

[131] J. Suo, X. Chen, S. Shan, W. Gao, and Q. Dai. A concatenational graph evolution aging model. *TPAMI*, 2012.

[132] J. Suo, S.-C. Zhu, S. Shan, and X. Chen. A compositional and dynamic model for face aging. *TPAMI*, 2010.

[133] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[134] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[135] M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B*, 1999.

[136] W. Tong, C. Tang, M. Brown, and Y. Xu. Example-based cosmetic transfer. In *FG*, 2007.

[137] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. *submitted to SIAM Journal on Optimization*, 2008.

[138] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[139] D. Vaquero, R. Feris, D. Tran, L. Brown, A. Hampapur, and M. Turk. Attribute-Based People Search in Surveillance Environments. In *Workshop on the Applications of Computer Vision (WACV)*, 2009.

[140] N. Wang, H. Ai, and F. Tang. What are good parts for hair shape modeling? In *CVPR*, 2012.

[141] R. Wang and X. Chen. Manifold discriminant analysis. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 429–436. IEEE, 2009.

[142] X. Wang. Intelligent Multi-Camera Video Surveillance: A Review. *Pattern Recognition Letters*, 34(1):3–19, 2013.

[143] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu. Shape and Appearance Context Modeling. In *International Conference on Computer Vision*, pages 1–8, 2007.

[144] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *ECCV*, 2010.

[145] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. *arXiv preprint arXiv:1406.5726*, 2014.

[146] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, pages 1473–1480, 2005.

[147] L. Wof and A. Shashua. Kernel principal angles for classification machines with applications to image sequence interpretation. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–635. IEEE, 2003.

[148] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.

[149] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.

[150] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition*, 2008.

[151] Y. Wong, S. Chen, S. Mau, C. Sanderson, and B. C. Lovell. Patch-based probabilistic image quality assessment for face selection and improved video-based

face recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 74–81. IEEE, 2011.

[152] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2009.

[153] T. Wu and R. Chellappa. Age invariant face verification with relative craniofacial growth model. In *ECCV*. 2012.

[154] Z. Wu, Q. Ke, J. Sun, and H.-Y. Shum. Scalable face image retrieval with identity-based quantization and multireference reranking. *TPAMI*, 2011.

[155] W. Xia, C. Domokos, , J. Xiong, L.-F. Cheong, and S. Yan. Segmentation over detection via optimal sparse reconstructions. *IEEE Trasactions on Circuits and Systems for Video Technology*, 2014.

[156] W. Xia, C. Domokos, L. F. Cheong, and S. Yan. Background context augmented hypothesis graph for object segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(4):582 – 594, Sept. 2014.

[157] W. Xia, C. Domokos, J. Dong, L.-F. Cheong, and S. Yan. Semantic segmentation without annotating segments. In *International Conference on Computer Vision*, 2013.

[158] W. Xia, Z. Song, J. Feng, L. F. Cheong, and S. Yan. Segmentation over detection by coupled global and local sparse representations. In *Proceedings of European Conference of Computer Vision*, 2012.

[159] C. Xiong, L. Liu, X. Zhao, S. Yan, and T.-K. Kim. Convolutional fusion network for face verification in the wild. *Circuits and Systems for Video Technology, IEEE Transactions on*, 2015.

[160] K. Yamaguchi, M. Kiapour, L. Ortiz, and T. Berg. Parsing Clothing in Fashion Photographs. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3570–3577, 2012.

[161] Y. Yang and D. Ramanan. Articulated Pose Estimation with Flexible Mixtures of Parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1385–1392, 2011.

[162] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.

[163] C.-N. J. Yu and T. Joachims. Learning Structural SVMs with Latent Variables. In *International Conference on Machine Learning*, pages 1169–1176, 2009.

[164] Q. Yuan, A. Thangali, and S. Sclaroff. Face identification by a cascade of rejection classifiers. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pages 152–152. IEEE, 2005.

[165] X.-T. Yuan, X. Liu, and S. Yan. Visual classification with multitask joint sparse representation. *Image Processing, IEEE Transactions on*, 21(10):4349–4360, 2012.

[166] X. Zhang, L. Zhang, X.-J. Wang, and H.-Y. Shum. Finding celebrities in billions of web images. *Multimedia, IEEE Transactions on*, 14(4):995–1007, 2012.

[167] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *Acm Computing Surveys (CSUR)*, 35(4):399–458, 2003.

[168] W. Zheng, S. Gong, and T. Xiang. Associating Groups of People. In *British Machine Vision Conference*, 2009.

[169] W. Zheng, S. Gong, and T. Xiang. Person Re-identification by Probabilistic Relative Distance Comparison. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 649–656, 2011.

[170] W. Zheng, S. Gong, and T. Xiang. Transfer Re-identification: From Person to Set-Based Verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2650–2657, 2012.

[171] S. Zhou, V. Krueger, and R. Chellappa. Face recognition from video: A condensation approach. In *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pages 221–226. IEEE, 2002.

[172] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning multi-view representation for face recognition. *arXiv preprint arXiv:1406.6947*, 2014.