Visual Data Association: Tracking, Re-identification and Retrieval

Feng Zheng

Department of Electronic and Electrical Engineering

University of Sheffield

A thesis submitted for the degree of

Doctor of Philosophy

Supervisor: Ling Shao

Declaration

Parts of this thesis have been included in the following published or submitted papers:

- **Feng Zheng**, Yi Tang, Ling Shao. Hetero-manifold Regularization for Cross-modal Hashing. Accepted by IEEE TPAMI, 2016, DOI: 10.1109/TPAMI.2016.2645565.
- **Feng Zheng** and Ling Shao. Robust and Long-term Motion Tracking for Intelligent Vehicles. Under peer-reviewed in IEEE TITS, 2017.
- **Feng Zheng** and Ling Shao. A Winner-take-all Strategy for Improved Object Tracking. Under peer-reviewed in IEEE TIP.
- **Feng Zheng**, Ling Shao, Vitomir Racic, James Brownjohn. Measuring Human-Induced Vibrations of Civil Engineering Structures via Vision-Based Motion Tracking. Measurement, Vol.83, 2016, pp. 44-56.
- **Feng Zheng**, Zhan Song, Ling Shao, Ronald Chung, Kui Jia. A Semi-supervised Approach for Dimensionality Reduction with Distributional Similarity. Neurocomputing, Vol. 103, pp. 210-221, March 2013.
- **Feng Zheng** and Ling Shao. Learning Cross-view Binary Identities for Fast Person Re-identification. IJCAI, New York, USA, July 2016.

  Feng Zheng, Ling Shao, James Brownjohn. Learn++ for Robust Object Tracking. BMVC, Nottingham, UK, Sep. 2014.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor Prof. Ling Shao, who gave me continuous encouragement and offered me such a great opportunity to study in Sheffield. His professional, flexible and patient guidance makes me feasible to freely drive interesting ideas of research and creatively conduct the studies in this thesis. More importantly, he gave me the continued support during the most difficult times.

I would like to thank the external examiner Prof. Peter Hall for his insightful suggestions and comments to improve the thesis.

I would like to thank the internal examiner Dr. Ali Gooya for his supports during the viva and the thesis editing process.

I must appreciate Prof. James Brownjohn very much for offering a chance to broaden my research vision to the area of civil structuring. Thanks for his selflessly support and generosity.

I must deeply acknowledge Dr. Xiantong Zhen for collaborating several interesting works and discussing some promising ideas in the past four years. I really admire his attitude of passion and enthusiasm on research and I learn a lot from him.

I would like to thank Prof. Shoubiao Tan for assisting me to successfully transfer my system of human computer interaction on Android platform. I feel very happy to know a person who has a frank manner to research and great enthusiasm on programming.

I sincerely express my thanks to Dr. Yi Tang for helping me to achieve the most exciting theoretical research of cross-modal hashing. I benefited a lot working with him all the time.

I am thankful to Bingzhang Hu for cooperating a novel idea of cross-age face retrieval. His fantastic idea makes it possible to realise such an amazing application.

I would also like to thank all members and visiting researchers from the University of Sheffield including Bo Dong, Di Wu, Fan Zhu, Jun Tang, Li Liu, Ruomei Yan, Simon Jones, Yang Long and Ziyun Cai, and the members from Northumbria University including Daniel Organisciak, Mengyang Yu, Lining Zhang, Yuming Shen and Yi Zhou. Thank them for their friendly help and their constructive discussions.

Thanks to the Engineering and Physical Sciences Research Council (EPSRC) and the University of Sheffield, who provided the financial support for me to complete my Ph.D. I recognize that this research would not have been possible without the financial assistance.

Finally, I take this opportunity to express the profound gratitude to my parents, my wife and my son. Thanks for all of their unconditional love, support, help, encouragement and dedication.

Abstract

As there is a rapid development of the information society, large amounts of multimedia data are generated, which are shared and transferred on various electronic devices and the Internet every minute. Hence, building intelligent systems capable of *associating these visual data at diverse locations and different times* is absolutely essential and will significantly facilitate understanding and identifying where an object came from and where it is going. Thus, the estimated traces of motions or changes increasingly make it feasible to implement advanced algorithms to real-world applications, including human-computer interaction, robotic navigation, security in surveillance, biological characteristics association and civil structure vibration detection.

However, due to the inherent challenges, such as ambiguity, heterogeneity, noisy data, large-scale property and unknown variations, visual data association is currently far from being established. Therefore, this thesis focuses on the studies of associating visual data at diverse locations and different times for the tasks of tracking, reidentification and retrieval. More specifically, three situations including single camera, across multiple cameras and across multiple modalities have been investigated and four algorithms have been developed at different levels.

Chapter 3 The first algorithm is to *explore an ensemble system for robust object tracking, primarily considering the independence of classifier members.* An empirical analysis is firstly given to show that object tracking is a non-i.i.d. sampling, under-sample and incomplete-dataset problem. Then, a set of independent classifiers trained sequentially on different small datasets is dynamically maintained to overcome the particular machine learning problem. Thus, for every challenge, an optimal classifier can be approximated in a subspace spanned by the selected competitive classifiers.

Chapter 4 The second method is to *improve the object tracking by exploiting a winner-take-all strategy* to select the most suitable trackers. This topic naturally extends the concept of ensemble in the first

topic to a more general idea: a multi-expert system, in which members come from different function spaces. Thus, the diversity of the system is more likely to be amplified. Based on a large public dataset, a prediction model of performance for different trackers on various challenges can be obtained off-line. Then, the learned structural regression model can be directly used to efficiently select the winner tracker online.

Chapter 5 The third one is to *learn cross-view identities for fast person re-identification*, in a cross-camera setting, which significantly differs from the single-view object tracking in the first two topics. Two sets of discriminative hash functions for two different views are learned by simultaneously minimising their distance in the Hamming space, and maximising the cross-covariance and margin. Thus, similar binary codes can be found for images of the same person captured at different views by embedding the images into the Hamming space.

Chapter 6 The fourth model is to *develop a novel Hetero-manifold regularisation framework for efficient cross-modal retrieval*. Compared with the first two settings, this is a more general and complex topic, in which the samples can be relaxed to the images captured in the very far distance or very long time, even to text, voice and other formats. Taking advantage of the hetero-manifold, the similarity between each pair of heterogeneous data could be naturally measured by three order random walks on this hetero-manifold.

It is concluded that, by fully exploiting the algorithms for solving the problems in the three situations, an integrated trace for an object moving anywhere can be definitely discovered.

Contents

Contents **vi**

List of Figures **x**

List of Tables **xii**

Nomenclature **xiii**

1 Introduction **1**

1.1 Visual Data Association 1

1.1.1 Potential Applications 3

1.1.2 Definition of Problems 5

1.2 Challenges, Hypotheses, and Solutions 8

1.2.1 Single Camera Object Tracking 8

1.2.2 Cross-camera Person Re-identification 12

1.2.3 Cross-modal Retrieval 14

1.3 Contributions 17

1.4 Thesis Outline 18

2 Literature Review **22**

2.1 General Learning Framework 22

2.2 Learning to Data Association 24

2.2.1 Associating in a Same Feature Space 25

2.2.2 Associating in Multiple Sources 26

2.3 Classical Learning Methods 27

2.3.1 Ensemble Learning 27

2.3.2 Online Learning 30

2.3.3 Graph-based Embedding 31

2.3.4 Hash Function Learning 34

2.4 Object Tracking 36

2.5 Person Re-identification 38

2.6 Cross-modal Retrieval 40
2.7 Discussion 41

3 Sampling Competitive Classifiers for Robust Object Tracking 43
3.1 Preliminaries 44
3.1.1 Hypotheses 44
3.1.2 Motivation 45
3.2 Structural Representation for Image Patch 47
3.3 "Concept Drift" in Object Tracking 49
3.4 Learn++ for Solving the Problem of "Concept Drift" 53
3.4.1 Motion constraints 53
3.4.2 Objective function 54
3.4.3 Basic classifier 55
3.4.4 Tracking by detection 56
3.4.5 Collecting and weighting samples 56
3.4.6 Sampling the classifiers 57
3.5 Experiments 59
3.5.1 Comparison with state-of-the-art methods 60
3.5.2 More analysis on the long-term sequences 65
3.6 Summary 66

4 A Winner-take-all Strategy for Improved Object Tracking 68
4.1 Preliminaries 69
4.1.1 Hypotheses 69
4.1.2 Motivation 70
4.2 Learning to Predict the Performance 71
4.2.1 Evaluation criterion 72
4.2.2 Motion features 72
4.2.3 Structural regression modal training 74
4.3 Online Winner-take-all Tracking 76
4.3.1 Winner selection 76
4.3.2 Online updating 77
4.4 Experiments 78
4.4.1 Two public datasets 78
4.4.2 Selected trackers 79
4.4.3 Overall performance on benchmark 80
4.4.4 Component analysis 82
4.4.5 Analysis of long-term tracking 84
4.5 Summary 85

5 Learning Cross-view Binary Identities for Fast Person Re-ID 86
5.1 Preliminaries . . . 87
5.1.1 Hypotheses . . . 87
5.1.2 Motivation . . . 88
5.2 Learning Cross-view Binary Identities . . . 90
5.2.1 Maximising the variance of bits . . . 90
5.2.2 Minimising the Hamming distance . . . 91
5.2.3 Overall objective function . . . 93
5.3 Optimisation . . . 94
5.3.1 Dual problem . . . 95
5.3.2 Greedy selection . . . 96
5.3.3 Convergence . . . 97
5.4 Experiments . . . 98
5.4.1 The efficiency of CBI . . . 99
5.4.2 Comparison with the state-of-the-art methods . . . 101
5.4.3 Comparison with other hashing methods . . . 104
5.4.4 Optimisation analysis . . . 109
5.5 Extension . . . 110
5.6 Summary . . . 112

6 Hetero-manifold Regularisation for Cross-modal Hashing 114
6.1 Preliminaries . . . 115
6.1.1 Hypotheses . . . 115
6.1.2 Motivation . . . 116
6.2 Hetero-manifold of multi-modal data . . . 117
6.3 Hash function learning on the hetero-manifold . . . 120
6.3.1 Distance inequality on a graph . . . 122
6.3.2 Objective function . . . 123
6.4 Sequential optimisation . . . 125
6.4.1 The first hash function learning . . . 126
6.4.2 The following hash function learning . . . 127
6.5 Experiments . . . 128
6.5.1 Cross-camera re-identification . . . 130
6.5.2 Cross-modal retrieval . . . 132
6.5.3 Cross-age face retrieval . . . 134
6.6 Summary . . . 138
6.7 Appendices . . . 140

7 Conclusions 146
7.1 Discussion . . . 146
7.2 Future Work . . . 148

Bibliography **151**

List of Figures

1.1 The general cases of visual data association 2
1.2 Vision based human-induced vibration detection 5
1.3 An overview of the main studies and the latent structure of the researches carried out in this thesis. 6
1.4 Two application examples of signal camera object tracking 8
1.5 A schematic diagram to show that the general purpose of Re-ID . 12
1.6 A sample of multi-modality to describe a kind of animal elephant 15
1.7 Summarisation and structure of all chapters. 21

2.1 The general learning flow for classical tasks 23
2.2 The general learning flow for data association. 26
2.3 Combining classifiers with different decision boundaries to reduce error and/or model selection 28

3.1 Visual data association in a signal camera using multiple classifiers 44
3.2 Motivation of learning competitive classifiers 46
3.3 The framework of fast structural representation 47
3.4 Two short videos to show the difference between the trivial case and drift case 50
3.5 Statistical analysis of the two situations in object tracking: “concept drift” and the trivial case. 51
3.6 Signoidal weights 57
3.7 The success plots and AUC rankings of 10 tracking methods. . . . 59
3.8 The success plots and AUC rankings on different challenging sequences. 62
3.9 More plots and rankings on different challenging sequences 63
3.10 Screenshots of top 9 tracking methods on challenging sequences . 64
3.11 The parameter setting 65
3.12 Comparsion of tracking results on three more challenging sequences 66
3.13 The weights for the optimal classifier 66

4.1 Visual data association in a signal camera using a winner classifier 69

4.2 Comparison of two trackers Struck and CT 70
4.3 Four cases of challenges captured by the motion flow 73
4.4 Screenshots of the 7 tracking methods 79
4.5 Overlap success and distance precision plots over the benchmark using OPE. 80
4.6 Overlap success and distance precision plots for different challenges. 81
4.7 More plots for different challenges 82
4.8 The parameter setting 83

5.1 Visual data association in a cross-camera setting 87
5.2 The motivation of learning cross-view identities 89
5.3 Some image samples of the two datasets 97
5.4 The Hamming distance between four images of two persons 100
5.5 The CMC rankings of the methods using the ELF feature on the VIPeR 102
5.6 The CMC rankings of the state-of-the-art methods on the VIPeR datasets 103
5.7 The CMC rankings on the CUHK01 datasets with #100 test persons104
5.8 The CMC rankings on the CUHK01 datasets with #486 test persons105
5.9 Comparison with four multi-modal hashing methods using ELF feature on the VIPeR dataset 106
5.10 Comparison with four multi-modal hashing methods using SCNCD feature on the VIPeR dataset 107
5.11 Comparison with four multi-modal hashing methods using LOMO feature on the VIPeR dataset 109
5.12 The optimisation analysis 111

6.1 Visual data association in a cross-modal setting 115
6.2 A hetero-manifold with three modalities 118
6.3 Cross-modal similarity 121
6.4 Some image examples of the two person re-identification datasets 129
6.5 The CMC rankings of the compared methods on the VIPeR dataset130
6.6 The CMC rankings of five methods on the CUHK01 dataset . . . 133
6.7 Precision recall curves on Wiki by varying the Hamming distance. 135
6.8 MAP performance for each category at 32 bits. 136
6.9 Some image examples of the FG-NET dataset 137
6.10 Overall performance comparison 138
6.11 The cross-modal first three matching results of two probe images . 140

List of Tables

1.1 The general challenges in object tracking 9

3.1 The discrepancy between the two distributions 53
3.2 The precision rankings of 14 tracking methods on challenging sequences 61

4.1 The properties of selected trackers 78
4.2 Comparison with the state-of-the-art methods 83
4.3 Component analysis 84
4.4 Precision comparison of tracking results on 7 long-term challenging sequences. S1 and S2 are collected from [1], S3 are from [2] and the last 4 sequences are from [3]. Precision is defined as the centre distance between the truth rectangle and the detected rectangle is less than 20 pixels. 84
4.5 Fail comparison of tracking results on 7 long-term challenging sequences. The number denotes which number of frames the centre distance of one tracker is larger than 20 pixels. 85

5.1 Time comparison of computing the similarities between one probe sample and all the gallery samples (316) using the compared methods. 99
5.2 Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using ELF feature on VIPeR dataset. 108
5.3 Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using SCNCD feature on VIPeR dataset. 108
5.4 Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using LOMO feature on VIPeR dataset. 108
5.5 Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using ELF feature on CUHK01 dataset. 110

5.6 Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using LOMO feature on CUHK01 dataset. 110

6.1 Ranking accuracy comparison 131
6.2 AUC Comparison on CUHK01 132
6.3 MAP Comparison on Wiki. 134
6.4 Rank 1 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities. 136
6.5 Rank 10 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities. 139
6.6 Rank 20 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities. 139

Chapter 1

Introduction

As there is a rapid development of diverse informational platforms, such as the Internet and surveillance systems and electronic devices, such as the smartphone and intelligent devices, enormous amounts of visual data are generated, shared and transferred. These data are huge assets and beneficial to improve the quality of our daily life. To that end, building intelligent systems capable of *associating these visual data at diverse locations and different times* is absolutely essential and will significantly facilitate the understanding of the behaviours and principals behind the data. Simply speaking, this thesis will fully identify where an object came from and where it is going.

However, due to the inherent challenges, such as ambiguity, heterogeneity, noisy data, large-scale property and unknown variations, the intrinsic structures within the multimedia data and relationships between them have not been fully discovered up to now. To address these difficulties, from the perspective of information sources, this thesis will comprehensively investigate three problems: single camera object tracking, cross-camera re-identification and cross-modal retrieval. By fully exploring the algorithms for the three situations, an integrated trace of a moving object anywhere can be definitely discovered, see Fig. 1.1. Thus, the estimated traces of motions or changes increasingly make it feasible to implement advanced algorithms to real-world applications, including human-computer interaction, robotic navigation, security in surveillance, biological characteristics association and civil structure vibration detection.

1.1 Visual Data Association

Data association is a fundamental tool for the research of natural science. In fact, since humans started to record data to represent natural things and phenomena in the real world, data association has become the significant component of human

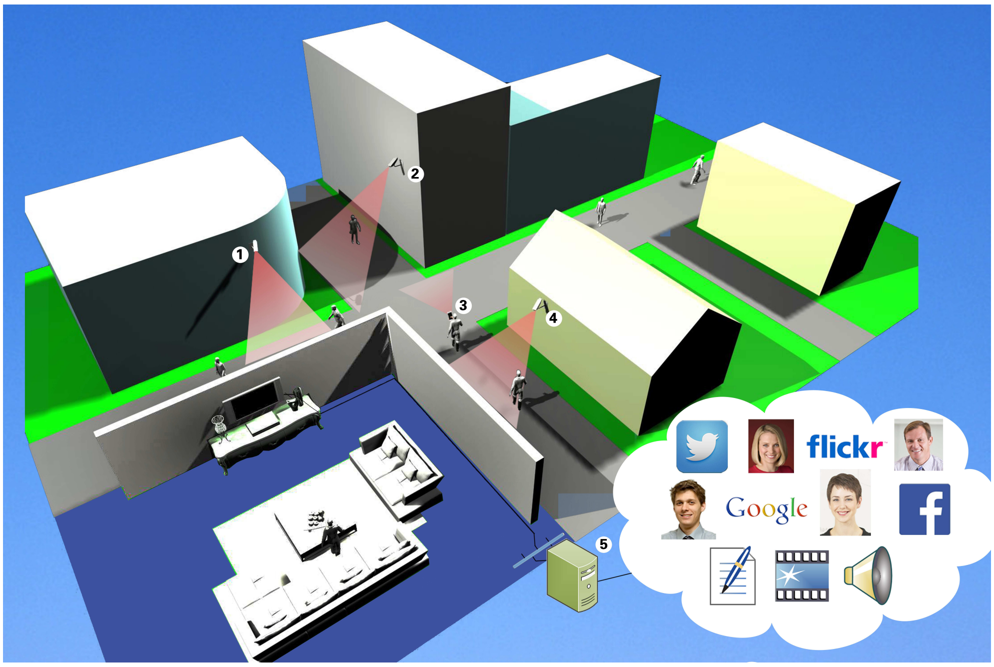


Figure 1.1: The general cases of visual data association studied in this thesis. Considering different types of information sources, these cases can be classified into three levels and several related tasks can be realised. Firstly, motions in a view of a moving (③) or a stationary camera (①, ②, ④) can be detected. Secondly, if one subject first goes though the view of camera ② and then goes thought the view of camera ③, the movement between the two cameras can also be estimated. Finally, the activities in a real world (①, ②, ③, ④) could be connected to some contents or activities on Internet or some other types of data (⑤).

living, the development of human intelligence and environmental exploitation, etc. The principal tasks of data association are to find the relationships, connect the traces, and deduce the semantic concepts between the data collected at diverse locations and different times. Generally, the most popular forms of data used by humans to describe the diversity of the world include signs, texts, signals, images, voice and video, which are generated by various sensors. On the one hand, according to the research conducted by the 3M corporation[1], 90% of information transmitted to the brain is visual. On the other hand, due to the advancement of the Internet and the availability of many imaging sensors, a huge amount of multimedia data are being generated every day in the form of images or video.

[1] http://www.3m.co.uk/

Thus, we can conclude that visual contents are the most important cues for the understanding and reasoning of humans in the real world. Therefore, in this thesis, we will focus on the visual data association.

We are experiencing an era of visual information explosion where the contents of visual data play a part in daily life everywhere. Firstly, according to the report from Worldwide Quarterly Mobile Phone Tracker in the International Data Corporation (IDC)[1], a total of 334.9 million smartphones were shipped worldwide in the first quarter of 2016 alone. Nowadays, a digital image sensor is a necessary configuration for almost all users and it is still a very important factor to evaluate a smartphone. Using these devices, hundreds of thousands of images and videos are generated, uploaded and shared by the users from all over the world. Secondly, from the research report released by IHS technology[2], 245 million worth of video surveillance cameras were installed globally in 2014 alone. These cameras, which are placed in public spaces ranging from transport infrastructures, shopping centres, and sport arenas to residential streets, are consistently producing huge numbers of videos. Moreover, for the Internet, the DMR[3] statistics report demonstrates that the multimedia data are the very important contents of daily life in modern society. As reported in April 2014, 300 hours of new videos were uploaded to YouTube every minute and 6 billion hours of videos were watched per month on YouTube. Up to July 2015, about 10 billion images were shared on Flickr, to name just two.

As a result, we can see that these large numbers of users mean that visual information is now a very important part of daily life and it is also a huge asset.

1.1.1 Potential Applications

By using the visual data, we have opportunities to discover the intrinsic structures, bridge the gap and associate the objects in different environments and platforms, then to create exciting and interactive applications. In fact, in general, methods of visual data association, including tracking, identifying and retrieval, have many potential applications. In this thesis, several examples of promising applications will be briefly introduced in the following, from the perspective of the three levels of research.

Vehicle and robotic navigation: The visual object tracking based intelligent system can be developed to automatically navigate and control the vehicles and robotics [4]. By utilising the extracted trace of targets around the automatic machines, the visual system can help the drivers of vehicles or robotics plan an effective path to pass through every area and avoid potential threats of collision.

[1] http://www.idc.com/

[2] https://technology.ihs.com

[3] http://expandedramblings.com/

Moreover, the intelligence visual system can also provide assistance to human drivers by alerting of the potential collisions and dangers.

Human-computer interaction: The gestures and motion of the human body can be exploited as the input signals in computers, mobiles or other intelligent devices for natural communicating between humans and equipment [5]. Due to the recent prevalence of low-cost Kinect, recently, visual object tracking algorithms have attracted many attentions from researchers from the Human Computer Interaction (HCI) community. Various methods have been proposed for tracking the poses or gestures of the human body or its parts, including the eye [6], head [7] and hand [8, 9].

Security in surveillance: Closed Circuit TeleVision (CCTV) cameras are common in commercial, industrial, and residential environments [10]. The fundamental task of visual surveillance is to track and identify the object, which has appeared in the view of a single camera or across multiple cameras [11]. Apart from the physical space, very recently, the association between images captured in surveillance and activities (texts, voices and images) in virtual space on the Internet, termed as cross-modal retrieval [12], has attracted much attention from researchers in both the computer vision and machine learning communities. By associating the images across cameras, or other types of contents on the Internet, the traces and activities of a suspect can be fully tracked and accurately identified. Moreover, some potential security threats can be found before taking action.

Biological characteristics association: Age estimation [13, 14] and kinship verification [15] using visual features in images are two general biological characteristic related topics in the computer vision community. Starting from these tasks, two more challenging but interesting applications, which will be discussed in the Chapter 6, can be implemented by associating the biological characteristics. The first one is cross-age face image retrieval, which can be used to search for an image of a person of a certain age in a large-scale dataset. The second one is to discover kinship links in which, using an image of a person, one can search the images of the kinship group of that person. The two tasks, which need to find age or heredity invariant features, could be very interesting and used in social platforms or websites.

Human-induced vibration detection: In addition to these popular applications of visual data associations including robotics, HCI, surveillance and biology, recently some researchers [16] have adopted a vision-based motion tracking method to detect the human-induced vibration of a civil structure. An example of experimental setting is given in Fig. 1.2. Compared to marker based systems and inertial acceleration sensors, vision motion tracking methods have the following advantages: 1) it is possible to measure people in outdoor environments; 2) the number of tracking individuals is not limited; 3) people are not aware of

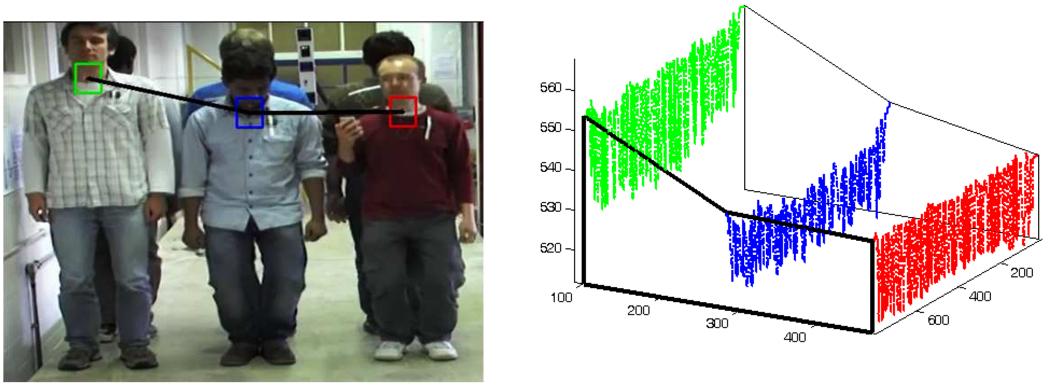


Figure 1.2: Vision based human-induced vibration detection. Left: experimental setting; Right: the detected signals of three subjects.

being recorded; 4) it is a cheap, remote and long-term monitoring system.

Visual data association has extensive applications, which are not only limited to the above examples but also in the experimental investigation of physics and chemistry, cell tracking in biomedicine and heavenly body motion tracking in astronomy. By associating the visual features, the relationship between samples, or the changes over time and locations, can be discovered and identified. Therefore, visual data association plays an important role in the social development, economy and research of science.

1.1.2 Definition of Problems

Basically, the general problems in computer vision include: 1) recognising what the object is and 2) determining where it is going. In recent years, for object detection and recognition, remarkable progress has been made by using the deep neural networks related algorithms. However, determining the traces or associating the samples at diverse locations and different times is still a very difficult task and far more than being established, because of the inherently challenging factors. Generally, these visual data are generated by a single camera, multiple cameras or shared in other platforms. From the perspective of data sources, the tasks or problems to associate samples can be divided into three parts: single-camera object tracking, cross-camera person re-identification and cross-modal searching. An overview of the main studies and the latent structure of the research carried out in this thesis are given in Fig. 1.3. To put it simply, the basic problem to be considered in this thesis is to determine where the object is going by using visual

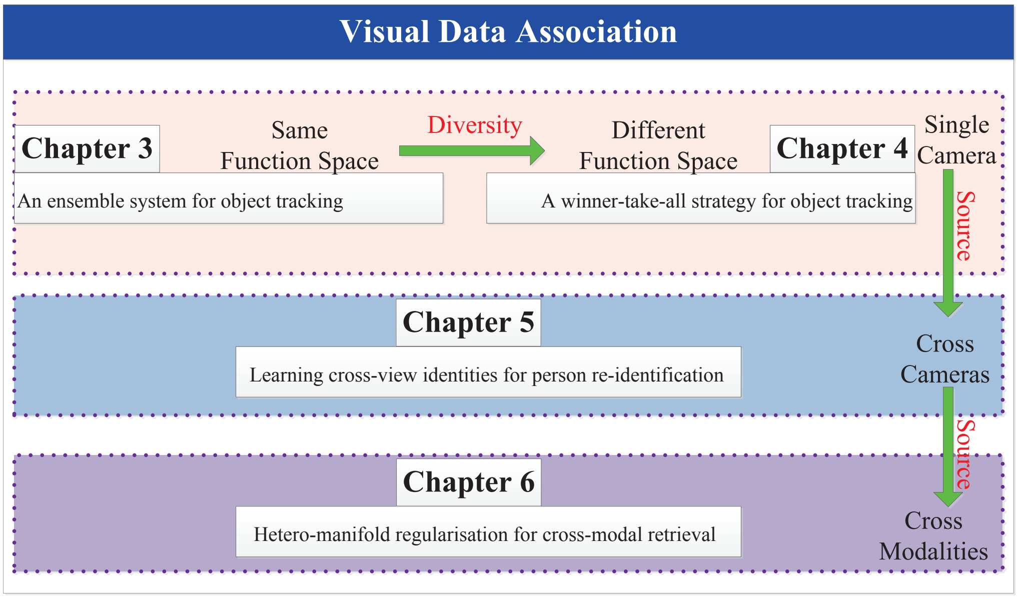

Visual Data Association
Chapter 3
Same Function Space
Diversity
Different Function Space
Chapter 4
Single Camera
An ensemble system for object tracking
A winner-take-all strategy for object tracking
Source
Chapter 5
Learning cross-view identities for person re-identification
Cross Cameras
Source
Chapter 6
Hetero-manifold regularisation for cross-modal retrieval
Cross Modalities

Figure 1.3: An overview of the main studies and the latent structure of the researches carried out in this thesis.

data association, no matter whether they are in a single view, across multiple cameras or active on other platforms.

Single-camera object tracking. The aim of object tracking is to detect a moving predefined target and associate its locations in the image space across frames.

- From the perspective of machine learning, what is the essential problem in object tracking? In general, in most cases, the target would be merely labelled by experts or detected by other classification methods in the first frame. Due to limited knowledge of a predefined target, the numbers of negative and positive samples are not balanced. In addition, the more important factor is that following changes of the environment around the target and the target itself are unpredictable when the classifiers are trained at the first frame.

- How can we balance the efficiency and robustness to overcome the "drift" problem for object tracking? Normally, to address the complicated challenges, complex algorithms will achieve better results but be more computationally expensive. In the real-world applications, most systems require less computation and real-time properties to reduce cost.

- How can we integrate the existing methods processing diverse properties to improve both the performance and efficiency? A large number of object tracking methods are proposed to consider diverse aspects of problems. It is very difficult to say which one is the best or which one can track targets in any conditions. However, they are all valuable and it is necessary to integrate them with a suitable strategy.

Cross-camera person re-identification. Person re-identification (Re-ID) has been defined as the recognition of an individual across non-overlapping camera views at diverse locations and different times. Solving such inter-camera people association problems involves tracking individuals across disjointed multiple camera views and enables consistent labelling of a person from diverse disconnected scenes.

- Similarity computing is quite important for data association and sample pair verification. Then, how is it possible to compute the similarity between the images captured by disjointed multiple cameras to identify people? Due to the difference of views and locations, the appearance of people changes a lot, leading to difficulties to capture the view-invariant features.

- How can we efficiently find the exact matches across views? A large number of surveillance cameras have been installed in public spaces where there may be tens of thousands of people assembled, even in a day. Thus, hundreds of thousands of images are generated and, therefore, it is very computationally expensive to search for an image in such a large-scale dataset. However, in real-world applications, real-time performance is generally required.

Cross-modal searching. Given a sample, the algorithm of cross-modal searching is used to search the most matched samples captured in other modalities or in other information sources. The task is normally achieved by learning cross-modal similarity. With the help of the learned similarity, it will enable us to realise image-document retrieval, heterogeneous face recognition (i.e., sketches and photos), kinship verification and cross-age face retrieval, etc.

- Compared with classical retrieval methods, in most cases of cross-modal searching, the samples are not in the same feature space. Then, how is it possible to compute the similarity between samples with different dimensions and diverse physical meanings? The key component is to overcome the heterogeneity between samples, then to learn the cross-modal similarity between them.

- How is it possible to connect and integrate all the information from data in different modalities to describe the diversity of the world? Although the

Figure 1.4: Two application examples of signal camera object tracking. Left: Automatically tracking and imaging by unmanned air vehicle; Right: non-touched human-computer interaction by hand gesture tracking.

data from different modalities have diverse forms and meanings, it is all connected in some relationships and reflects certain aspects of things and laws of nature.

- How is it possible to efficiently search the most matched samples in a very large dataset? In the cross-modal setting, due to the rapid development of the Internet and hardware, enormous volumes of multimedia data are generated. Thus, algorithmic efficiency is very prominent for real-world applications.

1.2 Challenges, Hypotheses, and Solutions

However, to date, visual data association is still very challenging. This section will further detail the discussions at three levels for each task: the inherent challenges, the hypothesis behind the proposed methods and the general framework to solve the problems. Two application examples are shown in Fig. 1.4[1].

1.2.1 Single Camera Object Tracking

From the perspective of development history, the research of object tracking undergoes four stages when we consider the different challenges and hypotheses. In the first stage, most methods generally supposed that the target moves smoothly and some of them also assumed that the cameras keep stationary. Thus, the image patch matching for two consecutive frames, such as Lucas-Kanade tracker

[1] The left image is from Dji: www.dji.com

[17] can be directly used. After that, some researchers thought object tracking as a dynamic system [18] to incorporate historical information by constructing the posterior probability density function. Then, about ten years ago, classification methods started to be used in object tracking [19, 20], in which the problem was considered as a detection and the classifiers would be updated to be adaptive to the changes. Recently, due to the limitations of the single model, multi-expert systems have been used to improve the diversity of trackers. The methods developed from a simple model to complex algorithm and the hypotheses are more relaxed and more challenges have been addressed.

Challenges: As in other areas in computer vision, a variety of challenges affect the performance of a tracking algorithm [21]. The general factors include: illumination, deformation, scale variation, rotation, occlusion, motion blur, clutters, disappearance, low resolution, fast motion and moving camera. Moreover, some algorithms have considered to solve more complex difficulties including specularity, transparency, long duration, low contrast and confusion etc. [3]. More details about the challenges are given in Table 1.1.

Table 1.1: The general challenges in object tracking

Illumination	the illumination in the target region is significantly changed
Deformation	non-rigid object deformation
Scale variation	the bounding box of the target is significantly changed
Rotation	the target rotate in or out of the image plane
Occlusion	the target is partially occluded
Motion blur	the target region is blurred due to the motion of target or camera
Clutters	the background has a similar colour or texture as the target
Disappearance	the target are fully occluded or move out of view
Low resolution	the number of pixels of the target is less than a certain small value
Fast motion	the motion of the target is larger than a certain large value
Moving camera	hand-held cameras
Specularity	specular highlights on the surface of the target
Transparency	being clear and transparent of the surface of the target
Low contrast	the variance of all pixels is less than a small value
Confusion	some similar objects are in the same view

Beyond these challenges, the following more complex difficulties are also considered in this thesis.

- In most cases, the varied challenges detailed in Table 1.1 may occur simultaneously. For example, when a running car is going across a curve, it is possible that the car will make a rotation in a certain angle in the image space and also be occluded by other cars or trees. Thus, it results in more difficulties in designing features and models to overcome the two challenges simultaneously.

- The total number of frames, which need to be tracked, is larger than a very large value. In this case, various challenges occur in the same video, thus, the tracker maybe updated by wrongly labelled samples. At present, normally, most methods are explored to solve the problems in short videos (only hundreds of frames) and seldom can be robust in a long duration tracking (more than 10 minutes).
- The proposed algorithms could be run in real time on a low-cost machine. In addition to performance, efficiency is also a significant factor in real-world applications, but most existing methods focus on performance only. In general, improving the performance will make more computation whist implementing a real-time system tries to reduce the computational expense as much as possible. Thus, it is difficult to balance the efficiency and performance on a low-cost platform.

Hypotheses: In this study, the hypotheses, which were used to base the proposed solutions for object tracking, include:

- Given a dataset with ground truth locations, the empirical analysis about the changes of the sample distribution can be discovered. Moreover, the empirical investigations can be used to reveal the nature of object tracking then to guide the design of more powerful trackers.
- By training on the different groups of samples, the classifiers could be relatively independent with each other and then be utilised to deal with different difficulties.
- It is assumed that, for a particular application, a set of trackers owning diverse properties can be selected from existing methods. Furthermore, a relationship between the performance of these selected trackers and motions (challenges) of a target can be modelled off-line on a large labelled dataset.

Solutions: To address the above complex problems, two models at different levels were proposed in this thesis, with considering the diversity of the models in different function spaces.

On the one hand, by introducing a method Learn++ into the object tracking, a set of Bayesian classifiers in a same function space was considered. In this model, the “concept drift” problems in object tracking was firstly empirically investigated. Then, according to the discoveries, a Learn++ (LPP) tracker was proposed to dynamically sample **competitive** classifiers for robust and long-term object tracking. To increase the efficiency and stability for the model, a competitive strategy was adopted to separately solve various “concept drift” challenges, which appeared in the same video sequence. Learn++ is a new group of machine learning methods used to learn additional information from new data, without accessing the original samples and it can be used for recognition tasks in very complex situations where new classes would join in. Moreover, Learn++ can address a set of databases in which the samples are generated by different distributions. To increase the diversity of the model, Learn++ keeps all the classifiers as long as they can achieve good performance on a subset. In Learn++, the weights of samples are updated using the ensemble performance. The subsets of basic classifiers are independent of each other and can be specified to solve certain different sub-problems, which occur in a non-stationary environment. Thus, for every challenge, an optimal classifier can be approximated in a subspace spanned by the selected competitive classifiers which can address the current problem according to the distribution of the samples and recent performance. As a result, the LPP tracker can efficiently address the various "concept drift" problems, which occur together in a long video sequence. Due to the use of sparse weights for the competitive classifiers, the LPP tracker can keep the balance between the efficiency and the performance.

On the other hand, to further improve the diversity of a system, a winner-take-all strategy was exploited to select a **winner** tracker, which is the most suitable and efficient to tackle the current challenge, according to the motion features extracted from the current environment and an efficiency factor. To fast extract features in a tracking environment, a dense trajectories-based motion feature was designed to describe the characteristics and challenges of the movement of an object and its surroundings. Based on a large public dataset, a prediction model of performance for different trackers on various challenges can be obtained off-line. Then, the learned structural regression model can be directly used to efficiently select the winner tracker online. To increase the flexibility of all members, the tracked results of the winner will be used to update other trackers. WTA is useful for two reasons: 1) by eliminating the non-maximal models, it reduces computation; 2) it provides additional robustness. The recently proposed tracking methods with multiple components generally achieve better results than other single models but most of them are far from real-time. In fact, the most computationally expensive part of every tracker is the procedure for image patch representation and state searching. However, to fuse the results from different components, all the components of most methods have to be run. In this thesis, the WTA strategy was exploited to select a winner tracker without running all members. Thus, not only will the performance be improved by incorporating multiple trackers, but also the average efficiency will be boosted.

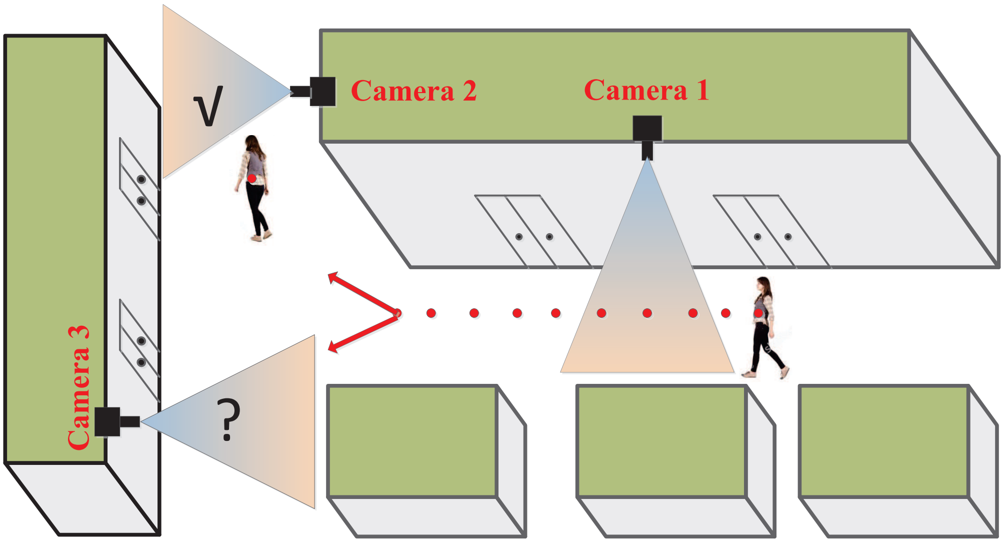


Figure 1.5: A schematic diagram to show that the general purpose of Re-ID is to detect the trace of a person for security or missing people. In this environment, a person walked though the view of camera 1 and then went into the view of camera 2. Using the person re-identification, the system can automatically match the images of the same person in the views of camera 1 and 2 efficiently and it will be at the same time, because we need not investigate the view of camera 3.

1.2.2 Cross-camera Person Re-identification

Cross-camera person re-identification (Re-ID) is a fundamental solution for automated video surveillance [22]. It has been defined as the recognition of an individual across non-overlapping camera views at diverse locations and different times. Solving such inter-camera people association problems involves tracking individuals across disjointed multiple camera views and it enables consistent labelling of a person from diverse disconnected scenes. Therefore, an effective Re-ID system is able to accelerate understanding of crimes, advance conventional fingerprints/DNA and contribute to all levels of policing and forensic science applications. A schematic diagram to show the general purpose is given in Fig. 1.5.

Challenges: Re-ID is a difficult vision problem because of the diverse visual appearance variations, visual ambiguity and spatial-temporal uncertainty and it possesses very challenging machine learning issues because of the high intra-class and low inter-class variations and limited/imbalanced image samples.

- Cross-camera variation: As there is a large difference in the camera views, which are installed in different locations, the appearances of same person can change significantly while different persons can look alike across camera views. Thus, this creates a typical problem of inter- and intra-class variation in the machine learning area.
- Long-term re-identification: At present, most person Re-ID methods are only limited to deal with the problems in a setting where the two cameras are not far from each other. However, re-identification in open environments can potentially scale to arbitrary levels, covering huge spatial areas spanning not just different buildings, but also different cities, or countries. In these cases, the greater change will be made due to some people wearing diverse clothes or a variety of carried objects will be taken by some persons in different camera views.
- Limited resources: To discover the relationship between any two views with a huge time and space separation, a large-scale dataset, which offers sufficient structural information and captures variability, is definitely required. However, in general, only one image or several images for each person can be offered in most cases. This then formulates a one-shot or multi-shot learning which is more complex than classical recognition tasks. Moreover, collecting sufficient labelled data from every camera to build a supervised model would be prohibitively expensive as well.
- Scalability: In general, the efficiency of matching mainly depends on two aspects: (1) the number of samples stored in the gallery set; (2) the definition of similarity. Considering the rapidly increasing usage of cameras in surveillance, it is impossible to reduce the number of samples. Recently, several metric learning methods have been exploited for associating people across views. However, these existing approaches generally cause a heavy computational burden when searching in a large dataset.

Hypothesis: To associate the persons at diverse locations and different times, it is assumed that some invariant features about the appearance and structure of a human body can be learned to represent individuals. Even if these meaningful features can be discovered by some heuristic methods or learning algorithms, it is still difficult to directly compute the similarity between the images captured in different views, because of the problem of ambiguity and uncertainty. Therefore, the general ways to compare the features suppose that a common feature space, in which a certain effective metric will be used, could be explored by some linear or kernel-based methods.

Solutions: In this thesis, to address the above problems, we accomplish person re-identification by learning a set of hash functions for each view. In fact, hashing has been widely used for nearest neighbour search in computer vision areas, such as image retrieval, object recognition and image matching, but it has seldom been used in re-identification. Using the hash functions, various special properties can be preserved in the learned codes, such as locality, variance and affinity. From the feature learning perspective, CBI learns a discriminative binary representation for each person. Furthermore, from the metric learning perspective, a more efficient distance metric in the Hamming space is learned for matching. Thus, similar binary codes can be found for images of a same person captured at different views by embedding the images into the Hamming space. Therefore, person re-identification can be solved by efficiently computing and ranking the Hamming distances between the images. Moreover, compact binary codes are extremely economical for large-scale data storage.

Specifically, in real-world applications, once an image of one person in one view is obtained, the projections are used to learn the on-line identity (ID) of that person. In fact, it is likely that the images of that person in other views somewhere else have also been captured and the ID has been obtained off-line. Then, the on-line ID can be used to search the corresponding person in another view by computing the Hamming distance between two sets of bits (IDs).

1.2.3 Cross-modal Retrieval

The nearest neighbour search across heterogeneous data has attracted considerable attention in recent years, due to the explosion of large-scale multimedia data in different modalities on the Internet and various devices, but it remains a very challenging problem. In a cross-media retrieval system, the query examples and retrieval results need not be of the same media type. For example, by submitting either a text or an image as a query, related audios can be searched. A sample to describe the kind of animal elephants using text, image and voice is given in Fig. 1.6. To speed up the nearest neighbour search, cross-modal hashing, which incorporates hashing techniques into cross-modal retrieval, has recently attracted much attention.

Challenges Cross-modal hashing has attracted considerable attention in recent years. However, despite the progress made by existing methods, it remains a very challenging task because of the integration complexity and heterogeneity of the multi-modal data.

- Semantic gap: In computer vision, low-level features, no matter whether they were designed by handcraft or learned automatically, are generally insufficient to directly represent the semantic meaning of data, such as labels, identities or classes. There is a big difference between these high-level tasks and computational representations.

Elephants are large mammals of the family Elephantidae and the order Proboscidea. All elephants have several distinctive features, the most notable of which is a long trunk or proboscis, used for many purposes, particularly breathing, lifting water and grasping objects. Their incisors grow into tusks, which can serve as weapons and as tools for moving objects and digging. Elephants' large ear flaps help to control their body temperature. Their pillar-like legs can carry their great weight. African elephants have larger ears and concave backs while Asian elephants have smaller ears and convex or level backs.

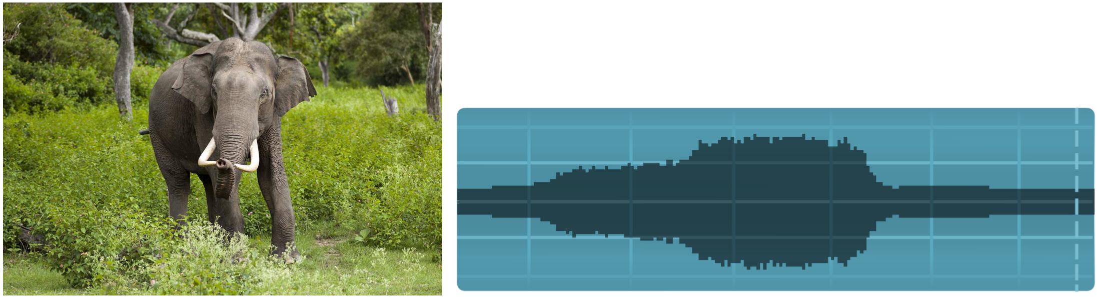


Figure 1.6: A sample of multi-modality to describe a kind of animal elephant using text (first row), image (left in second row) and voice (right in second row). It should be noticed that the sequential oscillogram is just used to denote the real voice of the elephant. The contents of text and image come from the free encyclopaedia, Wikipedia, and the real voice comes from a website: www.soundbible.com/.

- Heterogeneity: Compared with other tasks, the heterogeneity is the most distinctive characteristic in cross-modal searching, because the query sample and the retried dataset are collected in different conditions and even using different sensors. No matter what level of representation there is, differences between modalities are very obvious so that it is difficult to build an effective model to both capture the differences and reveal the common factors between different modalities. For example, computing the similarity between two representations with different dimensions is very challenging.

- Efficiency: Like other tasks of information retrieval, the efficiency of searching is also a very significant factor in real-world applications. In fact, the number of samples in cross-modal retrieval is much more than the number of samples in a uni-modal setting; because more sensors are used, then more samples are generated. In addition, what is more important is that the computation of similarities between the samples from different modalities is more complex.

Hypothesis: In this study, the principal aim is to connect and integrate all the information contained in different modalities into a uniform framework. Then, based on the integrated structure, a set of projections can be learned and the samples from different modalities can be embedded into a common space. The hypotheses behind this, which are used to support the solution in this thesis include:

- High dimensional data tend to lie in the local structure of a low dimensional manifold. This is a basic assumption for the classical manifold learning in a uni-modal setting, but it is still useful in cross-modal tasks. Based on this point, a sub-manifold can be modelled for each modality to capture the intrinsic structure of intra-manifold.
- The sub-manifolds in different modalities could be connected by some supervised information, or latent variables, which can be defined in diverse forms, considering the specificity of the problems. This is a basic assumption to support the connectivity and integrity of hetero-manifold in this work.
- The information could be propagated on an integrated framework in the cross-modal setting. Only when the information can be diffused in a certain pattern, can a global view be built based on these sub-manifolds so that it enables to cross-modal retrieval.

Solutions: In this thesis, by integrating the supervision information and the local structure of heterogeneous data, a novel method, termed hetero-manifold regularisation (HMR), is proposed to learn the hash functions for efficient cross-modal searching. Three significant advantages are made in the proposed framework. Firstly, a hetero-manifold well describes the local information by representing homogeneous data on the sub-manifolds. The data in different modalities are represented by different sub-manifolds, which model the relationship well between the homogeneous data. Secondly, the hetero-manifold emphasises the global information of multi-modal data as well, by modelling the *information propagation* across modalities with three-order random walks. It is clear that any pair of points could be connected via two steps on homogeneous sub-manifolds and one step crossing two different sub-manifolds. Thus, the samples across modalities can be compared by integrating the information from all related homogeneous sub-manifolds. Lastly, the hetero-manifold is flexible and can be extended to model any number of modalities. As far as we know, existing cross-modal searching algorithms are limited to only two modalities.

Taking advantage of the hetero-manifold, the similarity between each pair of heterogeneous data could be naturally measured by three order random walks on this hetero-manifold. Furthermore, a novel cumulative distance inequality, defined on the hetero-manifold, is introduced to avoid the computational difficulty induced by the discreteness of hash codes. By using the inequality, cross-modal hashing is transformed into a problem of hetero-manifold regularized support vector learning. Therefore, the performance of cross-modal searching can be significantly improved by seamlessly combining the integrated information of the hetero-manifold and the generalization of the support vector machine.

1.3 Contributions

The contributions made in this thesis are summarised below:

- Chapter 3: In sampling competitive classifiers for robust object tracking, we make the following three contributions: (1) Empirical analysis, which concludes that object tracking is a non-i.i.d. sampling and small dataset problem, is given to guide the design of the proposed tracker. (2) A new framework of Learn++ (LPP), particularly for object tracking, is proposed. Unlike classical Learn++ methods, a competitive subset of classifiers, which consists of the ones that are more adaptive to the current environment, is maintained, a constraint (motion) is added to guide the learning and new samples are only used to update these classifiers trained in similar situations. (3) As far as we know, the LPP tracker is the first tracking method that designs an explicit model for each sub-problem (i.e., challenge) and the models can be automatically altered according to the environment. The most distinctive merit of the LPP tracker is that, even when a target moves out of view and then comes back with totally different locations and appearances, the tracker can still lock the target, as long as the appearances ever appeared in the past. Because of training a relatively large set of classifiers, the LPP tracker keeps its diversity so as to improve the generalization and performance, no matter what concept drifts occur. Despite the complexity of the model, due to the sparsity of the competitive set, the LPP tracker is still a real-time method.

- Chapter 4: In the winner-take-all (WTA) strategy for improved object tracking, the main contributions of this work are as follows: 1) As far as we know, we are the first to build a structural regression model trained on a large dataset, to fast predict the probabilities of the performance for existing methods. 2) Both the effectiveness and efficiency of trackers are integrated into auniform winner-take-all framework. Only one representative tracker needs to be executed at any time, but the results can reflect the strengths of all trackers. 3) The proposed WTA framework is tested on a large dataset and our evaluation results demonstrate that WTA can hugely improve the performance, without sacrificing the efficiency.

- Chapter 5: In learning cross-view identities (CBI) for fast person re-identification, our contributions are three-fold. (1) By learning the compact binary codes, each person has a similar identity across different views. Due to the efficiency of binary codes, person re-identification in a huge dataset can be realised. For the VIPeR [23] dataset, which contains only 316 samples in the gallery, CBI is at least 2200 times faster than the non-hashing state-of-the-art methods. However, if there are millions of samples, CBI will be also millions of times faster. (2) In CBI, variances of learned bits, cross-covariance and margin of learned hash codes are simultaneously maximised and an efficient iterative optimisation solution is introduced. (3) Moreover, in CBI, a theoretical proof is given to guarantee the transfer from Hamming space to Euclidean space. Unlike most methods, which directly relax the sign function, such as [24], we consider the theoretical reason behind when it is safe to relax the sign function.

- Chapter 6: In the Hereto-Manifold Regularisation (HMR) for cross-modal hashing, the novelties conclude that, firstly, the hetero-manifold is a well-defined platform to capture both local information of sub-manifolds corresponding to homogeneous data and global information of the hetero-manifold corresponding to multi-modal data. Secondly, the proposed hetero-manifold support vector hashing, taking advantage of the hetero-manifold in representing the information of multi-modal data and the support vector machine in generalisation, can generate more effective hash functions for the cross-modal search. Finally, comprehensive experimental results show the effectiveness and efficiency of the proposed hetero-manifold regularisation-based hashing algorithm to tackle the problems of cross-camera person re-identification, cross-modal image retrieval and cross-age face retrieval.

1.4 Thesis Outline

The rest of this thesis is organised as follows:

Chapter 2 contains a full literature review of basic learning methods and the previous works relevant to visual data association. First of all, the basic knowledge of machine learning, including classification, regression and ranking are introduced. These methods construct the basis of our solutions for visual data association, considering different aspects of the essential problems. Then, the related work about single-camera object tracking, cross-camera person re-identification and cross-modal retrieval will be described and analysed sequentially.

Chapter 3 presents a Learn++ (LPP) tracker which used to dynamically sample competitive classifiers for robust and long-term object tracking. In particular, an efficient descriptor which can be selected by classifiers is firstly given. Then, an empirical analysis of "concept drift" problems is used to guide the design of tracker. Next, the Learn++ based tracker is proposed to overcome the challenges in the non-stationary environment for object tracking. Finally, extensive experiments show that LPP tracker yields state-of-the-art performance under various challenging environmental conditions and, especially, can overcome several challenges simultaneously.

Chapter 4 describes a winner-take-all (WTA) strategy to select a **winner** tracker (considering both accuracy and efficiency) from a set of prevailing methods to tackle the current challenge, according to features extracted from the present environment and an efficiency factor. To this end, firstly, a structural regression model to characterise the trackers is discussed. Then, this chapter introduces how to select the most suitable tracker, the ways to locate the target and how to update the trackers. The proposed WTA framework is tested on a large benchmark dataset and extensive experimental results illustrate that WTA can significantly improve both the performance and the efficiency.

Chapter 5 proposes a method to learn cross-view binary identities (CBI) for fast person re-identification. To achieve this, three aspects, including minimising the distance in the Hamming space, maximising the cross-covariance and maximising the margin are considered simultaneously. This chapter gives a theoretical proof for when it is safe to transfer the problem in Hamming space to a problem in Euclidean space and what constraints need to be considered, as well. Extensive experiments are conducted on two public datasets to show CBI produces comparable results as state-of-the-art re-identification approaches, but is at least 2200 times faster than these non-hashing methods.

Chapter 6 provides a novel method termed hetero-manifold regularization (HMR) to supervise the learning of hash functions for efficient cross-modal searching. Hetero-manifold integrates multiple sub-manifolds, defined by homogeneous data, with the help of cross-modal supervised information. In this chapter, at first, various definitions of hetero-graphs for different conditions are fully discussed. Next, a novel cumulative distance inequality, defined on the hetero-manifold, is introduced. Then, cross-modal hashing is transformed into a problem of hetero-manifold regularized support vector learning and solved by a sequential optimisation method. Lastly, comprehensive experiments on four datasets show the proposed HMR achieves advantageous results over the state-of-the-art methods in several challenging cross-modal tasks.

Chapter 7 details our conclusion and future work.

To facilitate the understanding of the contents and structures in a holistic view for this thesis, an overview of main developments is given in Fig. 1.7.

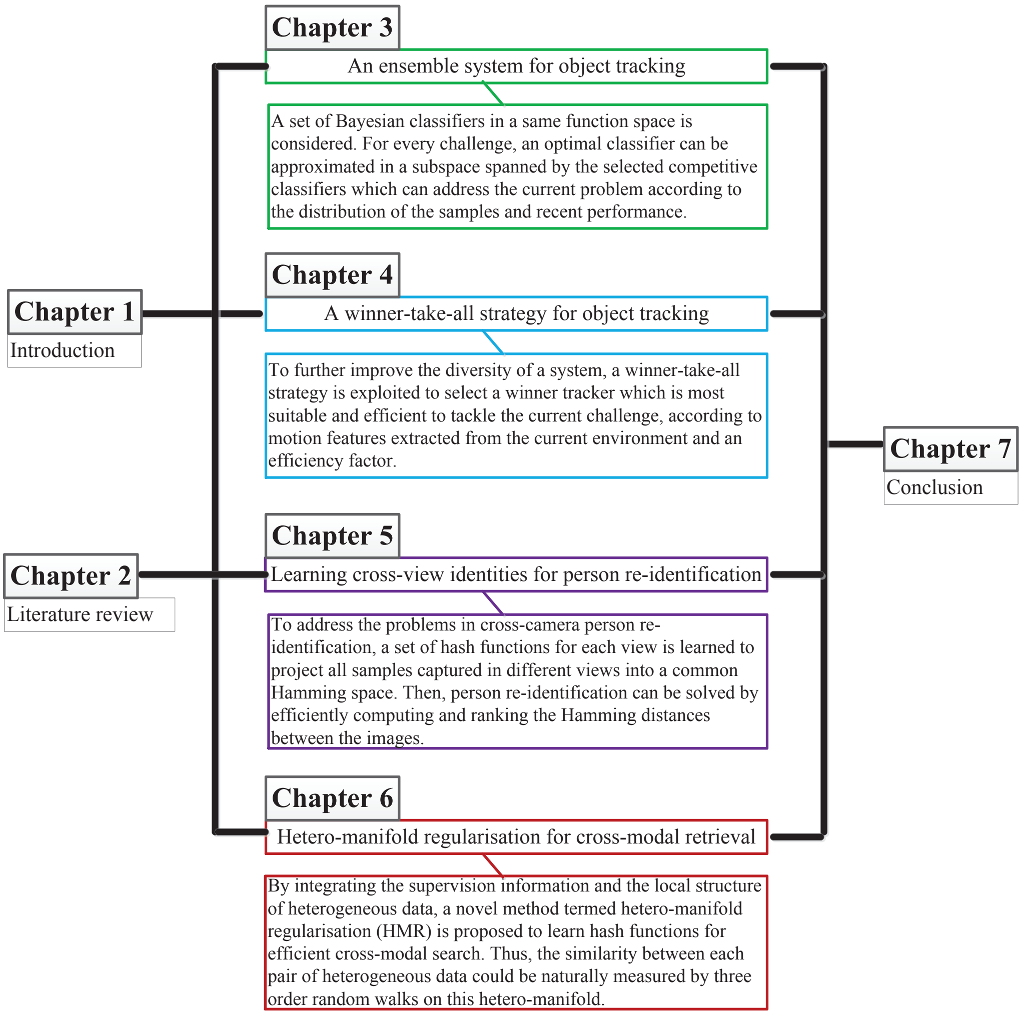

Chapter 3
An ensemble system for object tracking
A set of Bayesian classifiers in a same function space is considered. For every challenge, an optimal classifier can be approximated in a subspace spanned by the selected competitive classifiers which can address the current problem according to the distribution of the samples and recent performance.
Chapter 4
A winner-take-all strategy for object tracking
To further improve the diversity of a system, a winner-take-all strategy is exploited to select a winner tracker which is most suitable and efficient to tackle the current challenge, according to motion features extracted from the current environment and an efficiency factor.
Chapter 1
Introduction
Chapter 7
Conclusion
Chapter 5
Learning cross-view identities for person re-identification
Chapter 2
Literature review
To address the problems in cross-camera person re-identification, a set of hash functions for each view is learned to project all samples captured in different views into a common Hamming space. Then, person re-identification can be solved by efficiently computing and ranking the Hamming distances between the images.
Chapter 6
Hetero-manifold regularisation for cross-modal retrieval
By integrating the supervision information and the local structure of heterogeneous data, a novel method termed hetero-manifold regularisation (HMR) is proposed to learn hash functions for efficient cross-modal search. Thus, the similarity between each pair of heterogeneous data could be naturally measured by three order random walks on this hetero-manifold.

Figure 1.7: Summarisation and structure of all chapters.

Chapter 2

Literature Review

This chapter will firstly provide a broad review of basic knowledge and concepts including classification, regression and ranking, which are used to support our solutions and discoveries in the following chapters and then introduce extensive backgrounds of present research in visual data association from the three levels: signal-camera setting, cross-camera setting and cross-modality setting.

2.1 General Learning Framework

Machine learning primarily focuses on the development of computer programs to deal with extensive problems with respect to some kinds of tasks and theoretical research of computational learning in artificial intelligence. The essential element of learning is to provide computers with the ability to iteratively learn from data (experience) and make decisions according to their learning and understanding, without explicitly being programmed. A dataset of observations is given:

$$X = \{x_1, x_2, \cdots, x_i, \cdots, x_N, x_i \in \boldsymbol{X}\}, \tag{2.1}$$

and the corresponding latent variable:

$$Y = \{y_1, y_2, \cdots, y_i, \cdots, y_N, y_i \in \boldsymbol{Y}\}. \tag{2.2}$$

The purpose of learning is to build a model from a hypothesis space $f \in \boldsymbol{F}$ to bridge the input x and output y, where the hypothesis space has $\boldsymbol{F} = \{f|y = f(x)\}$ and N is the number of samples. In general, the samples x could be the data captured by any sensor and have diverse structural forms, including vector, matrix and tensor. For example, an image is denoted by a matrix and captured by a camera. The output y generally refers to the supervised information, such as labels, clusters or other high-level semantic variables. The output has a variety of forms ranging from two values $\{+1, -1\}$, numbers and real values to more

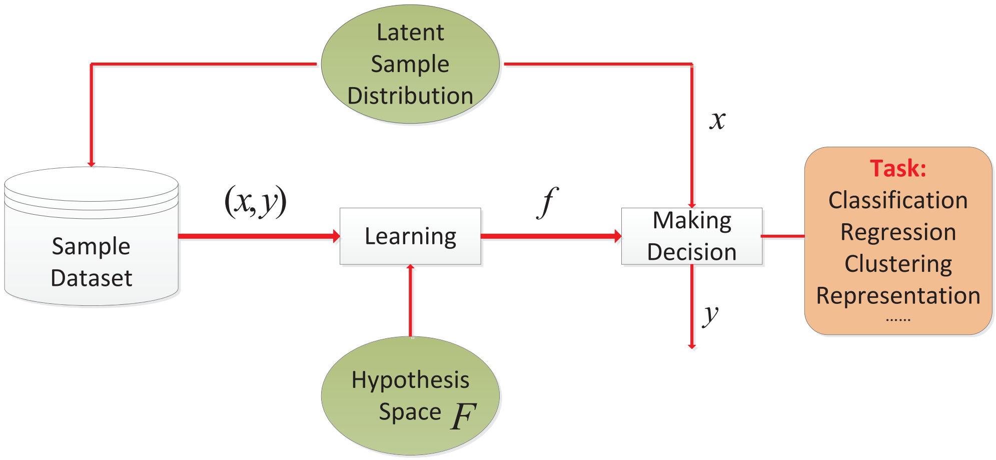


Figure 2.1: The general learning flow for classical tasks including classification, regression and clustering etc..

complex structures including vectors, matrices and tensors which derive from the complicated structural output learning. The general flow of learning to detail the procedure of training and making decisions is shown in Fig. 2.1.

In taxonomy of machine learning, most criteria depend on the output variable y. Firstly, if considering the presence status of y, most of machine learning methods can be classified into supervised learning in which all samples are given labels, semi-supervised learning in which a part of the samples (generally a little portion) are given labels and rest of the samples are unlabelled, and un-supervised learning in which all samples are unlabelled. Secondly, if considering the forms of output y, most of methods can be categorised into classifications in which latent variable is only from $\{+1, -1\}$, clustering in which latent variable denotes the index of clusters and regression in which latent variable is a real value. Finally, if considering the functional forms in hypothesis space, most of machine learning methods can be grouped into linear, non-linear and kernel-based methods. For example, neural networks in which a non-linear activation function has been adopted are non-linear methods. In fact, there are some other types of taxonomies, e.g., depending on the size of the hypothesis or the searching strategy.

In simple terms, classical machine learning methods seem to be uncorrelated to the task conducted in this thesis: data association. However, in fact, data association could be achieved if we consider these classical learning tasks as a transition procedure. Then, the straightforward strategy is that samples can be associated by grouping the latent variable y in a certain way or by defining a similarity measurement for the latent variable. The next subsection will bridge the gap between the classical tasks and the task of data association.

2.2 Learning to Data Association

This subsection will introduce what the data association is and how to associate the samples using leaning strategies. Learning recognition or prediction has a relatively long history of research and has an explicit definition of what is machine learning. However, leaning data association has rarely been discussed because it is more complicated and, in many cases, it is only considered as a straightforward extension of learning. In fact, data association which, to some extent, includes the classical learning method is a more general concept in the areas of data mining and artificial intelligence. Given a dataset of observations:

$$X = \{x_1, x_2, \cdots, x_i, \cdots, x_N, x_i \in \boldsymbol{X}\}, \tag{2.3}$$

and the corresponding associations:

$$A = \{a_1, a_2, \cdots, a_j, \cdots, a_n, a_j \in \boldsymbol{A}\}, \tag{2.4}$$

where $a_j = \{x_{j_1}, \cdots, x_{j_n}\} \subseteq X$ and j_n is the number of samples in the jth association, the purpose of data association is to build a model from a hypothesis space $f \in \boldsymbol{F}$ to verify whether all the samples in a new set $a_j = \{x_{j_1}, \cdots, x_{j_n}\} \subseteq \boldsymbol{X}$ belong to a same association (group) with a certain semantic meaning or not? Thus, simply, the hypothesis space can be defined as:

$$\boldsymbol{F} = \{f : \{x_{j_1}, \cdots, x_{j_n}\} \Rightarrow \{+1 - 1\}\} \tag{2.5}$$

where $+1$ denotes all the samples in the set, which belong to same group, whilst -1 denotes that they do not. The final best hypothesis will be the one, which achieves the best results on the training sets and obtains the highest generalisation ability for the future set.

To seamlessly connect the learning procedure and data association, the first important thing is to investigate the difference between the general learning framework and the above scheme of data association. Generally, there are three major differences:

- Feature space $\boldsymbol{X}$: The definition of feature space is very broad and generalised. In the classical learning methods, the representation of samples is required to be in the same space, have the same dimensions and describe same physical knowledge. In this thesis, this limitation is extremely relaxed and the samples could be captured by different sensors, be endowed diverse physical meanings and collected in different locations or times.

- Input of hypothesis: The number of input samples is also relaxed as well. In the classical learning methods, generally, one sample is considered as the input because these tasks are required to make a decision for the sample itself. However, the task of associating data is to connect a set of samples and it allows that any number of samples, which are normally larger than 2, are considered as the input of a system.

- Hypothesis space $\boldsymbol{F}$: Because the samples are possibly collected by different sensors, various hypothesis spaces should be adopted for the samples in different modalities according to the representation form and physical meaning. For example, a best projection for each modality, which is searched in different hypothesis spaces, can be determined and then a common new embedding space for all the samples in different modalities can be obtained. Thus, data association can be achieved in this learned common space.

TTo drive the learning scheme to the data association, a latent variable y which is the same as in classical methods, can be defined by supervised information, and is introduced into the framework of association in Eq. 2.5. The general learning flow for data association is shown in Fig. 2.2. Then, we can see that the latent variable plays a role of intermediary to bridge the input of samples and output of decisions if these samples belong to same group or not. In addition, from the above discussions, we can see that the most distinctive characteristics of data association are the information sources. Thus, we divided our topic into two levels: the same feature space and multiple information sources.

2.2.1 Associating in a Same Feature Space

The first case of data association is to connect samples in a same feature space. For example, object tracking is to associate the image patches captured by a same camera in different times. These patches can be considered as the samples in a same feature space, no matter what representation methods are finally adopted. Thus, if a latent variable y can also be introduced, then we have:

$$\{y_{j_1}, \cdots, y_{j_n}\} = f(\{x_{j_1}, \cdots, x_{j_n}\}). \tag{2.6}$$

Therefore, the final association can be determined by considering the results of the latent variable y, where we have $\{y_{j_1}, \cdots, y_{j_n}\} \Rightarrow \{+1 -1\}$. That is to say, if the corresponding values in $\{y_{j_1}, \cdots, y_{j_n}\}$ are the same as each other, then these samples can be thought of as in a same association. Due to the fact that all samples are in the same space, a single hypothesis f can be learned to project all samples into the domain of the latent variable:

$$\{y_{j_1}, \cdots, y_{j_n}\} = \{f(x_{j_1}), \cdots, f(x_{j_n})\}. \tag{2.7}$$

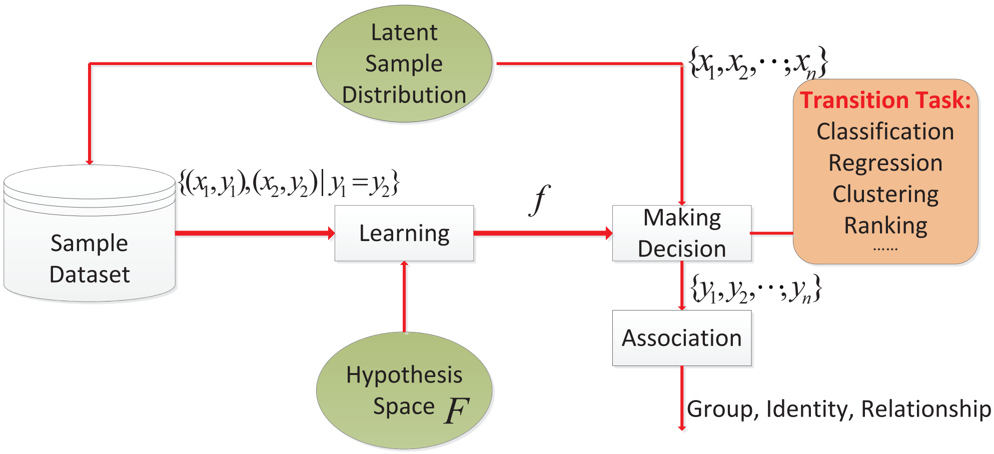

Latent Sample Distribution
$\{x_1, x_2, \cdots; x_n\}$
Transition Task:
Classification
Regression
Clustering
Ranking
……
$\{(x_1, y_1), (x_2, y_2) \mid y_1 = y_2\}$
f
Sample Dataset
Learning
Making Decision
$\{y_1, y_2, \cdots; y_n\}$
Association
Hypothesis Space F
Group, Identity, Relationship

Figure 2.2: The general learning flow for data association.

However, what is the latent variable in this case? The simplest answer is the label of samples and then the methods of classification $y = f(x)$ can be formulated to deal with the problem of data association. Particularly, this means that all the samples with same labels would be in a same association. In fact, most methods of tracking-by-detection adopt the strategy of classification to associate targets moving in the image space in different time. Furthermore, the latent variable y could de defined using other semantic information including index of clusters and identity of objects etc.

2.2.2 Associating in Multiple Sources

The second case of data association is to connect the samples captured from different sensors (modalities) where we have $\boldsymbol{X} = \{\boldsymbol{M}^1, \cdots, \boldsymbol{M}^M\}$ and M is the number of sensors. For example, in cross-modal searching, some voices can be searched from a large-scale dataset by submitting either a text or an image as a query. This application enables us to track a criminal from his/her activities on the Internet or some social platforms by only using the text descriptions of a witness at the scene of committing a crime. In the case of multiple sources, data association can be achieved by calculating the similarities between any pair of samples in a learned common space $\boldsymbol{Y}$. In general, the first step is to design a joint optimisation framework, which is used to learn a projection for each modality to embed all the samples into such a common space. The Eq. 2.5 will become:

$$\{y_{j_1}, \cdots, y_{j_n}\} = \{f^{k_1}(x_{j_1}^{k_1}), \cdots, f^{k_2}(x_{j_n}^{k_2})\}. \tag{2.8}$$

where we suppose that $x_{j_1}^{k_1} \in \boldsymbol{M}^{k_1}$ and $x_{j_n}^{k_2} \in \boldsymbol{M}^{k_1}$. Various strategies, which are used to integrate and connect all the information from different modalities and the critical components of models, are to overcome the heterogeneity in the multiple sources. The second step is to define a similarity or a distance $D_{\boldsymbol{Y}}(y_1, y_2)$ between samples in the learned common space to associate the samples. Therefore, if Euclidean distance is used, then we have:

$$D_{\boldsymbol{Y}}(y_1, y_2)^2 = ||y_1 - y_2||_2^2 = ||f^{k_1}(x_1^{k_1}) - f^{k_2}(x_2^{k_2})||_2^2. \tag{2.9}$$

In fact, the measurement in the new learned common space is not only limited to Euclidean distance but also some more complicated measurements with special advantageous properties can be used in this framework. For example, a Hamming space can be considered as the learned common space and then the Hamming distance of binary codes can be used to measure the similarity between any pair of samples. It will enable us to fast cross-modal data association in a very large-scale dataset.

2.3 Classical Learning Methods

From the above discussions, we can see that learning data association can be achieved by using classical machine learning methods, such as classification and embedding learning, etc. In this section, some basic works, which are related to the new proposed methods in this thesis, will be deliberately discussed including ensemble learning, online learning, graph-based embedding and hash function learning.

2.3.1 Ensemble Learning

Ensemble learning is generally used to fusion the results of individual models to improve the overall performance, which is superior to those of its constituent individuals. In the area of machine learning, the first key component of ensemble learning is to train the individuals by dividing the samples into separated groups [25], partitioning the feature space [26] or even considering different hypothesis spaces [27]. The second key component is to integrate the individual models to reach the final decisions, according to a certain criterion including voting, weighting or a strategy of winner-take-all. From the two components, we can see that, in fact, ensemble learning is derived from a classical strategy of divide-and-conquer. That is to say, the first component is used to divide the problem

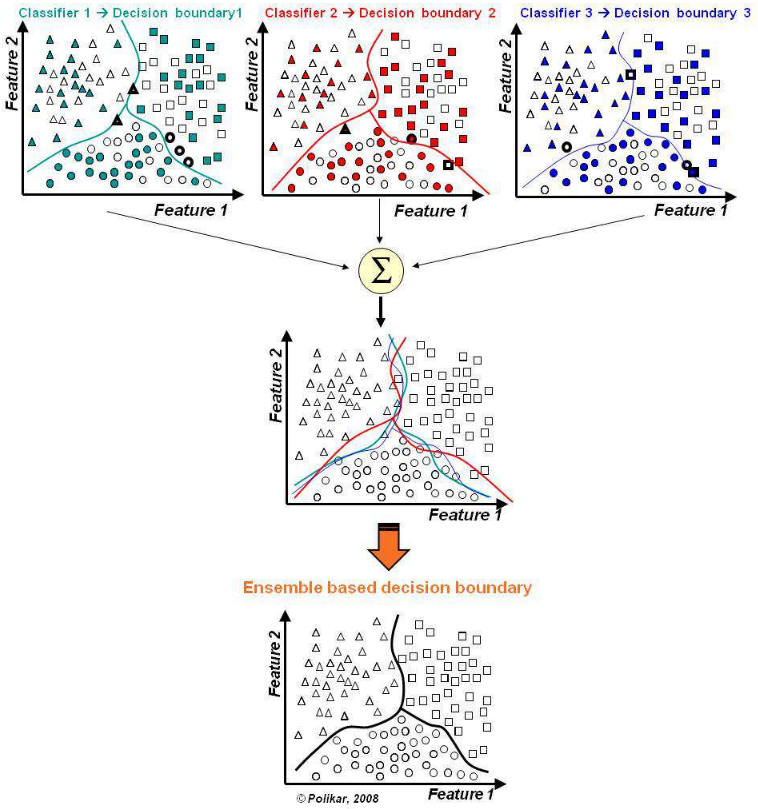

Classifier 1 → Decision boundary1
Classifier 2 → Decision boundary 2
Classifier 3 → Decision boundary 3
Feature 2
Feature 1
Σ
Ensemble based decision boundary
© Polikar, 2008

Figure 2.3: Combining classifiers with different decision boundaries to reduce error and/or model selection. The original figure refers to [28].

while the second one is exploited to conquer it. In Fig. 2.3, there is an example of ensemble learning to show that the combination of the classifiers provides the best decision boundary and reduces the overall error more than the individuals. In addition, the original figure refers to [28].

To put it simply, there are two principal reasons to make the methods of ensemble learning successful in many applications, such as the Viola-Jones algorithm for fast face detection [29]. The first theoretical reason is explicitly explained in Schapire's work [30] that it is likely to convert a weak learning algorithm into one that achieves an arbitrarily high accuracy. In this work, through a boosting procedure, some weak classifiers that perform only slightly better than random guessing could be combined to a strong classifier in a probably approximately correct (PAC) [31] sense, which is correct on all but an arbitrarily small fraction of the instance. The other reason probably from an empirical aspect is that the diversity of ensemble could be improved by the combination of the individual models. From the perspective of hypothesis space, improving the diversity of an ensemble system could reduce the bias of the hypothesis space, which means that the optimal combined model is much closer to the real optimal solution. In [32], the authors proved that the generalisation error of the ensemble model is guaranteed to be less than, or equal to, the average generalization error of the component individuals. For a more comprehensive review of the diversity issues in the ensemble learning, see [33]. The two most related ensemble learning methods, which the topics in Chapter 3 and 4 are based on, are introduced in the following.

On the one hand, most machine learning algorithms can learn from data that are assumed to be drawn from a fixed, but unknown, distribution. Taking the tracking problem as an example, however, this assumption is invalid. Traditional machine learning methods applied to the tracking problem will fail when there is a "concept drift" in the NSE. That is because the learnt function on a fixed sample set previously collected may not reflect the current state of nature due to a change in the underlying environments. Learn++ [34], which is an ensemble of classifiers originally developed for incremental learning, can be adapted for solving the "concept drift" problem in the NSE and to information/data fusion applications. It specifically seeks the most discriminative information from each data set through sequentially generating an ensemble of classifiers. The classifiers trained on individual data sources are fine tuned for the given problem (concept drift). Learn++ can still achieve a statistically significant improvement by combining them, if the additional data sets carry complementary information.

On the other hand, winner-take-all can be considered as a case of competitive learning and has very wide applications including max-pooling in machine learning, political voting and commercial investment. In the area of machine learning, WTA is a computational principle that can be implemented using different types of models [35]. In [36], Lee et al. show that the contrast gain, orientation and spatial frequency of an image can be activated by a winner-take-all competition among overlapping visual filters. In [37], by simulating the principle of virtual cortex, a novel MAX-like operation on inputs to certain cortical neurons is designed for visual object recognition. Inspired by the simulation, the convolutional neural network [38] also adopts the max-pooling to generate values in the first few layers. In [39], image synthesisability relevant features are learned to select the case-optimal method among several existing alternatives for texture synthesis. To realise $100,000$ object classes detection, the WTA hash [40] is applied in [41] to replace the dot-product kernel operator in the convolution operation. The WTA strategy can also be used for model selection [42] and action selection [43].

2.3.2 Online Learning

Online learning normally refers to a group of machine learning methods, which process one datum at a time [44]. The counterpart of online learning is the methods of batch learning, which instead consider the dataset as a whole in the stage of optimisation. Compared with the batch scheme, online learning has its potential advantages and could be used in some cases where it is unsuitable for the batch scheme. Roughly speaking, the scope of online learning methods could be divided into two groups: 1) handling large-scale data; 2) learning to predict the streaming of time-series data. In general, the strategies exploited in the two groups are similar to each other since both are required to process a datum at a time. However, they are separately applied for two different purposes. More particularly, the first group of methods normally are used to speed up the optimisation or to avoid the case of memory overflow when faced with a large-scale dataset, whilst the second group of methods are especially used to solve the problems in which samples are not obtained in advance but become available over time, usually one at a time. The second case is very common in the numerous real-world applications, especially for the time-series data, e.g., object tracking and weather forecasting. The object tracking will be discussed in Chapter 3 and 4, using online schemes.

On the one hand, in the early stage of machine learning, most of methods are designed in some online schemes. For example, in Rosenblatt's Perceptron machine [45], this algorithm is a supervised learning method for binary classification and updates the linear function (weight vector) model in an additive form, whenever a new sample is misclassified, by adding this new sample to the original function. After several years, in [46], Novikof theoretically proved that the Perceptron machine would be converged after a finite step of updating, for a linearly separable classification problem. The Perceptron machine pioneered the research of machine learning and gave huge expectations for artificial intelligence, which would eventually replace human intelligence. The most distinctive method of online learning to speed up the classical method, which adopts a batch scheme for training, is the Sequential Minimal Optimization (SMO) [47]. SMO transfers the large QP problem into a series of smallest time-consuming QP problems. In [48], Bottou and Cun argued that suitably designed online learning algorithms asymptotically outperform any batch learning algorithm for a large-scale problem. Recently, to learn a measure of similarity between pairs of objects, [49] proposed an online dual approach, using the passive-aggressive family of learning algorithms. In [50, 51], the online hashing approaches were proposed to address the problems of large-scale streaming data.

On the other hand, there are many potential applications for online learning, including online email categorization and spam filtering, object tracking and stock price prediction, etc. In [52], the author stated that, compared with other off-line methods, relaxed online SVMs actually achieve similar classification results on online spam filtering in large benchmark data sets. A comparison of early methods between batch and online learning on spam filtering is given in [53]. In [54], the authors demonstrated that, in accordance with email relevance, an online label ranking algorithm automatically classifies their messages into user defined folders. Object tacking, which is also investigated in this thesis, is also a traditional area of online learning methods. In [55], online feature selection is firstly exploited for object tracking. In 2008, incremental PCA [56] is used to adaptively learn the intrinsic subspace feature for visual tracking. Based on Adaboost tracking [20] introduced by Avidan, an online version of Adboost [57] is used to feature selection through weighting the weak classifiers trained on different features. In addition, online multiple instance learning [58], online structural output learning [59] and online Learn++ all are exploited for object tracking. In addition, to make profit-maximised decisions and investment, using online machine learning algorithms to analyse and predict financial time series is also an area of active interest. In [60], by measuring the correlation between the stock market events and the features both in the micro-blogging platform and an induced interaction graph, an online market simulation system that can be used to guide stock traders. Bollen [61] also pointed out that the collective mood states derived from large-scale Twitter feeds are correlated to the stock market. Several online machine learning methods were analysed and validated by Soulas on how to find highly correlated pairs of securities and how to predict foreign exchange rate changes in an online fashion [62].

2.3.3 Graph-based Embedding

Graph-based Embedding methods generally transform the objective data from a original space of a high dimensionality to a low dimensional space, preserving as much of the significant structure as possible, such as linear structure (e.g., principal direction variance [63], Euclidean distance [64, 65]) and nonlinear geometric characteristic (e.g., local tangent [66], local linearities [67], local heat kernel [68], geodesic distance [69], diffusion distance [70]). In Chapter 5 of this thesis, the two sets of projections which preserve both the intra-modality variance and inter-modality covariance are learned to embed the images captures by different cameras into a one common space. Based on this idea which only used for person re-identification, in Chapter 6, an more advanced framework of hetero-manifold regularisation is explored to project samples from multiple modalities into a common Hamming space, with preserving both high-order intra-modality and inter-modality structures. Simply, these methods could be categorised into unsupervised learning, semi-supervised learning and supervised learning.

Unsupervised Algorithms: Principal Component Analysis (PCA) [63] and metric multidimensional scaling (MDS) [64] are the two representative unsupervised approaches for linear dimensionality reduction. As for nonlinear dimensionality reduction algorithms, the representative methods include local tangent space alignment (LTSA) [66], locally linear embedding (LLE) [67], Laplacian eigenmaps (LE) [68], isometric feature mapping (Isomap) [69], and DM [70], etc. These algorithms are generally named as manifold learning which is an emerging and promising approach in nonlinear dimensionality reduction. A manifold is a topological space that is locally Euclidean. LTSA [66] obtains the low intrinsic manifold by global minimization of the reconstruction error of the set of all local tangent spaces in the data set. LLE [67] and LE [68] focus on the preservation of local neighbor structure. Isomap [69] seeks the subspace that best preserves the geodesic distances between any two data points. DM method relates the spectral properties of Markov processes on a weighted graph (G, W) and preserves the diffusion distance introduced in DM [70]. These linear and nonlinear unsupervised methods are mainly designed to embed high dimensional data into low dimensional space with preserving geometric information. Such methods only utilize the geometric relationship between samples, such as linear structure and nonlinear geometric characteristic. Mostly, such geometric information is not sufficient to discriminate different samples especially they are very close in the transformed spatial space. Consequently, the introduced label information can play an important role and provide useful information for accurate and robust classification.

Supervised Algorithms: Linear discriminant analysis (LDA) [65] is a well-known linear supervised algorithm. LDA maximizes the ratio of inter-class variance to the intra-class variance to guarantee maximal separability. LDA projects data into low dimensional space with preserving Euclidean distance and the label information are used as constrains. In recent years, many dimensionality reduction algorithms which preserve different kinds of geometric information with label constrains have been proposed.

To the supervised algorithms, it only exploits the geometric and label information of the labeled samples. Fukumizu et al. [71] presented a novel kernel method for dimensionality reduction with Reproducing Kernel Hilbert Spaces in the setting of supervised learning. In [72], a general framework of supervised dimensionality reduction was proposed, which viewed both features and class labels as exponential-family random variables, and allowed to mix-and-match data- and label- appropriate generalized linear models for classification and regression. In [73], an improved version of Isomap, namely S-Isomap, was proposed. S-Isomap utilizes class information to guide the procedure of nonlinear dimensionality reduction which was not sensitive to noise. Kouropteva et al. [74] and Li et al. [75] also built the supervised based extension of LLE and LTSA, respectively.

In [76], Sajama and Orlitsky presented a method based on maximum conditional likelihood estimation of mixture models which ensured that the selected subspace retained maximum possible mutual information between feature vectors and class labels. Liang and Li [77] developed a general regularization framework for dimensionality reduction by allowing the use of different functions in the cost function. The framework can be used as supervised learning with prior knowledge of label information. In [78], most popular subspace learning algorithms, unsupervised or supervised, were unitedly explained as instances of a ubiquitously supervised prototype.

Semi-supervised Algorithms: These supervised algorithms are very effective for learning the low dimensional representation of labeled samples. But from an engineering point of view, it is clear that collecting labeled data is generally more difficult than collecting unlabeled data [79]. As a result, some data sets include a small amount of labeled samples and a large number of unlabeled samples. To use the geometric and label information contained in data sets more effectively, a few semi-supervised frameworks were proposed for dimensionality reduction. Some methods use label information based on the framework of LDA for defining the different similarity metrics or neighborhoods [80, 81, 82, 83]. In [80], Zhang et al. defined the cannot-link and must-link constraints as prior information corresponding to the between-class and within-class matrices of LDA, respectively. Zhang et al. [81] and Sugiyama et al. [82] presented a similar framework which defined within and between similarity based on LDA for global preserving, and local similarity based on LPP [84] for local preserving. In [83], Song et al. proposed a method which defined the within-manifold, between-manifold and total-manifold scatter matrices similar to that in LDA. Xu and Yan [85] presented a semi-supervised subspace learning algorithm by integrating the tensor representation and the complementary information conveyed by unlabeled data.

There are some other methods which consider dimensionality reduction as a regression algorithm from a high dimension space to a low dimension space [86, 87]. They assume that the low intrinsic coordinates of a part of trained samples are known. Yang et al. [86] showed that classical unsupervised algorithms could be modified by taking into account prior information on exact mapping of certain data points. They reformulate the minimization problem of classical methods using the label information, so that the global low dimensional coordinates could be computed by solving a linear set of equations. In [87], Gong et al. converted the classical minimization problem with a special kernel to an optimization problem with equality constraints, and the final solution could be obtained by diffusion from the labeled data points.

2.3.4 Hash Function Learning

Despite the promising performance achieved by the metric learning related methods in various areas, all of them suffer from a huge computational burden in the test stage. Recently, the hashing techniques have been widely adopted to solve the problem in many vision applications, especially in indexing large-scale data. This is because hashing methods can map high-dimensional features to compact binary codes that are efficient to match and robust in preserving original similarity [88]. In this thesis, the two hashing based methods will be explored to fast cross-camera person re-identification in Chapter 5 and efficiently cross-model retrieval in Chapter 6.

After the well-known Locality Sensitive Hashing (LSH) [89], two types of methods have been developed to generate hash codes. The data-independent methods [90] are generally based on random projections whilst the learning-based hashing methods [91, 92] make use of the data distribution. On the one hand, LSH scheme based methods which are normally data-independent are proposed to preserve various kinds of distances or similarities. The distances which need to be kept are generally defined on a general space (any kind of features) and thus the hash functions are normally generated according to some specific distributions. Nice theoretic properties of distributions [40, 93, 94, 95, 96, 97] guarantee that certain distance or similarity can be preserved by the hash projection. On the other hand, learning to hash, which is, in contrast, data-dependent, aims to learn more compact binary codes by preserving some data-driven measurements in the original feature space specified by a dataset X. Among them, the binary codes refinement has also drawn much attention for some special purposes or some specific conditions. By joint optimization of search accuracy and search time simultaneously, the hashing buckets were perfectly balanced [98]. Using the spherical hashing scheme [99], both balanced partitioning for each hashing function and the independence between any two hashing functions were achieved. In [100], by maximizing the consistency between semantic distance and hashing-based Hamming distance, the pre-computed hashing bits could be reused. Recently, several interesting methods including ensemble learning based hashing [101], cross-modal hashing [102] and hashing for distributed data [103] are also explored. In the following, three groups of methods about affinity-based loss, quantilization loss and cross-modal retrieval are briefly reviewed.

Affinity-based Loss: Some methods consider the affinity or distance distortion as a kind of loss. when they are too far apart (or too close). In [104], each hash function was designed to correct the errors made by the previous one. Instead of the neighbour related pairs defined in [104], the learned hash function (hyperplane) was required to cross the sparse region of data samples [105]. In [106], a hinge-like loss function is advocated to control the ratio of the slopes of the penalties incurred for similar (or dissimilar) points. Li et al. [107] introduce a triplet based hinge loss to encode the relative comparison relationships in the data. In [108], the graph hashing problem is cast into a discrete optimization framework which directly learns the binary codes. Although the authors call affinity or distance distortions as losses, these quantities play the same role as the items of smoothness regularizations, manifold regularizations and pair-wise constraints in classification. Actually, the higher-order relationships including listwise supervision [109], top rank [110] and semantic multi-Label [111] could be also considered to learn hash functions. More details about similarity preserving in hashing could be found in [112].

Quantilization Loss: Self-taught hashing [113] firstly decomposes the learning procedure into two steps and, then, the idea is extended in [88] to accommodate many different affinity-loss functions. The first step of binary codes learning can typically be formulated as binary quadratic problems, and the second step of hash function learning can be accomplished by training standard binary classifiers. [114] formulate the hashing framework as a multi-class classification, where the learned binary codes (surregate labels) are expected to be optimal for classification. To improve the two-step hashing, several works [115, 116] consider to optimize the binary codes and the hash functions, alternatively. Another group methods follow the similar scheme of two-step hashing but, the first step considers the affinity-based loss as a smooth problem and the second step minimizes the quantilization loss by optimally finding the thresholds for the learned continuous variables. Iterative Quantization (ITQ) [115] minimize the quantization error of mapping the PCA-projected data to vertices of the binary hypercube. Similarly, Isotropic hashing [117] can produce embedded dimensions for the PCA-projected data with isotropic variances thus, to some extent, reduce the quantization error as well. Followed the alternative quantilization, various techniques including locally linear reconstruction weight [118], graph Laplacian matrix [119] and bilinear projection [120] are applied into the first step. Actually, the nature of these methods is to minimize the loss of mean squares between the binary codes and auxiliary continuous variable.

Moreover, the hinge loss [102, 121, 122] could be used to minimize the risk of the auxiliary continuous variable close to 0. In [121], a hinge loss is used to learn hash function one by one with considering the similarity-similarity difference. Recently, the hinge loss is used to learn a set of hash functions in cross-modal setting [102, 122]. In [102], Zheng et al. prove that, by incorporating the hinge loss, the discrete optimization problem could be minimized certainly by minimizing an differentiable upper bound. In [123], a orthogonal transformation is searched so that the sum of cosine similarities (Angular quantization loss) of each transformed data point and its corresponding binary landmark is maximized.

Cross-modal Hashing: More recently, the hash function learning is extended to multi-modal data or multiple information sources, especially for cross-view retrieval, such as text and image. Composite Hashing with Multiple Information Sources (CHMIS) [124] and the Cross View Hashing (CVH) [125] extend the SH [91] from different aspects, respectively. The boosting algorithms are adopted to embed the input data from two arbitrary spaces into a same Hamming space by Cross-Modality Similarity Sensitive Hashing (CMSSH) [126] and an anchor-supported method [127]. Considering the maximum margin, Predictable Dual-view Hashing (PDH) [24] explores a joint formulation for learning binary codes of data from two different views. Collective Matrix Factorisation Hashing (CMFH) [128] is based on the assumption that the interlinked data should have the same latent factors and the hash codes can be learned from these factors. Moreover, local functions [129] and correlation-maximal mappings [130] are exploited to learn the common binary codes.

2.4 Object Tracking

Generally, according to the type of samples used to build the model, the online adaptive algorithms can be divided into two groups: generative methods which only use positive samples to infer the relationship between them, and discriminative methods which use both positive and negative samples to train a classification hyperplane. Moreover, from the perspective of development history of object tracking, there are four stages: optical flow to match two consecutive frames [17], particle filter to model the underlying dynamics of a motion system, tracking by detection and multi-expert model.

A most basic concept of object tracking is direct image patch matching. Following this basic idea, there are several well-known methods, such as: Lucas-Kanade tracker [17], fragments-based tracker [131] and mean shift tracking [132]. However, the target in these methods is not updated according to the appearance change of the object. Thus, an essential step forward is to build a generative appearance model to capture the variation over time, such as online subspace learning [56] and sequential Monte Carlo sampling [133]. Recently, sparse coding based methods catch much attention in the community of object tracking. Reported in the two experimental evaluation results [3, 134], the two sparse coding based methods [135] and [136] achieve high performance. However, despite the superior performance on partial occlusion, in a survey [137], the authors state that their experimental results have shown that visual tracking may not be a sparse representation problem. Moreover, generative methods would easily fail with a cluttered background.

Considering the significant role of discriminative information from background, pioneered by support vector tracker [19] and ensemble tracker [20], various discriminative algorithms have been built to model the difference between the foreground and the background. Collins et al. [55] explore a mechanism which adaptively selects the most discriminative features from a set of different colour spaces. In addition, the random projection is used in compressive tracking (CT) [138]. However, CT is a data-independent method which guarantees that no noise is introduced but lacks flexibility. Moreover, numerous methods exploring the different properties of samples and relationships between samples, including P-N learning [1], semi-supervised SVM [139], semi-supervised boosting [140], multiple instance learning [58], weighted reservoir sampling [141] and semi-supervised transfer learning [142], have been also proposed to improve the performance of trackers. Recently, in [143], the confidence of a classifier is considered as a probability which can be analysed using Gaussian Processes regression. Structured output tracking with kernels (Struck) [59], using the windows as input, explores the training data with the form of appearance and translation. The experimental survey [144] concludes that Struck performs well on all aspects but one, the change of scale, bringing it to the number one position over their entire dataset.

Futhermore, in the past few decades, numerous methods integrating multiple components are proposed to solve the various challenges. Intuitively, the diversity can be improved by using the information or knowledge from multiple sources. According to the stage of different components, we can categorise these methods into three groups: combination of features, ensemble of classifiers in a same hypothesis space and multi-expert trackers.

Combination of features: Collins et al. [55] explore a mechanism which adaptively selects the most discriminative features from a set of different colour spaces. [145] fuses multiple observation models with parallel and cascaded evaluation. Yoon et al. [146] adopt two steps: tracker selection and interaction to fuse multiple features. In [147], three different levels of features are modelled to enable robust model relearning.

Ensemble of classifiers: The co-tracking algorithm [139] trains multiple SVM classifiers using different feature types and combines their tracking results to achieve robust tracking. A set of random ferns is adopted to explore comparative features in [1]. Visual tracker sampler [148] incorporates a process of sampling trackers into the framework of particle filtering [18], without differentiating the trackers. Randomised ensemble trackers [149] consider the weights of classifiers as a non-stationary distribution. Three Struck [59] based trackers with different features [150] are combined to select the best tracking result among the three forward trackers. In [151], several online SVM algorithms are used as the base classifiers and a minimum entropy criterion is designed to evaluate the members.

Multi-expert trackers: In [152], the Lucas-Kanade (LK) [17] method and one random forest based classifier are combined for target tracking. Similar to [152], Yan et al. [153] design an ensemble framework for the optimal selection of detectors and trackers to do multi-target tracking. Recently, a complex system with multiple components [154]: a short-term Integrated Correlation Filter (ICF) processing, a short-term key points processing, a long-term memory updating, an output controller and a ICF updater, is proposed to produce sensitive and stable responses to complex situations.

All these methods require various updating schemes to capture the continuous deformations of the objects. As a consequence, they tend to drift by incorporating wrong information. To avoid the drifting, diverse strategies are adopted, such as: different update rates [155], data-independent knowledge [138] and selectively updating the parts [156]. However, the essential reason why drifting occurs is that classical trackers have not considered the object tracking as a "concept drift" problem and tried to solve different challenges by only one super-power model. In fact, the differences between various challenges are very large. Thus, we can see the limitation of the classical trackers comes from the basic i.i.d. assumption in machine learning, on which most of tracking-by-detection methods depend. The drifting problem is not very obvious and can be partially solved by the classical methods in short sequences but it is still quite difficult for the long-term tracking [144]. In the tracking-by-detection methods, recovering from drift may also prove a useful way to make tracking robust but the update of wrong information will destroy the structure of the classifier.

2.5 Person Re-identification

Recently, inspired by [23], person re-identification catches much attention of researchers. Although several researchers start to consider situations of multiple-shot [157, 158] or video based [159] person re-identification, most of methods still focus on one-shot situations where, to some extent, the tasks are more difficult because less information is available for each person. To address the challenge of person re-identification, many efforts have been made along the two directions: learning discriminative features and learning the metric functions. Moreover, both aspects are considered to further improve the performance in [160].

On the one hand, the learned features are generally invariant to the view changes and simple metrics are used for matching. Compared with [23], a stronger feature representation (SCNCD) [161] is proposed, in which the colour distributions over colour names in different colour spaces are obtained. A different feature transform for a pair of configurations is learned in [162]. The visual features are projected to a common feature space and matched by a local expert. [163] points out that certain appearance features can be more important than others in describing an individual from other people. Unsupervised salience learning [164] is used to extract distinct features but suffers from the computational burden because the feature of one image is related to a large scale reference dataset. Similar to salience learning, part-based template matching [165] is exploited to person re-identification by using the candidacy graph and cluster sampling mechanism. In [166], deep learning is exploited to automatically learn features for the re-identification task and the deep framework has been improved in [167] by incorporating neighbouring locations of other images. Moreover, in [168], deep neural networks are used to learn the features with maximising the relative distance.

On the other hand, complex distance metrics are learned to rank the pairs of observations from different views. Support Vector Ranking (PRSVM) [169] is firstly used for ranking the feature differences. Zheng et al. [170] introduce a relative distance comparison (PRDC) model to maximise the likelihood of a pair of true matches having a relatively smaller distance than that of a wrongly matched pair. By using the equivalence constraints, Köstinger et al. [171] propose an efficient metric learning method to reduce the training time. In [172], the unsupervised and supervised dimensionality reduction methods are combined to learn the intrinsic representation and distances are computed in the learned lower space. Very recently, ensemble metrics, such as a mixture of similarities [173] and an ensemble of distances [174], are exploited to discover multiple matching patterns.

In fact, very recently, some researchers have started to focus the vehicle re-identification which is similar to the person re-identification. Some strategies proposed for person Re-ID could be directly used for vehicle but most methods cannot be applied. This is because the vehicle is a rigid object whist the person normally is non-rigid. Moreover, the shapes between different vehicles are more likely to be similar and the appearances of vehicle are also simpler. In [175], a large-scale benchmark dataset for vehicle Re-Id in the real-world urban surveillance scenario is collected. This dataset contains over 40,000 bounding boxes of 619 vehicles captured by 20 cameras in unconstrained traffic scene. At the same time, another dataset named as “CompCars” [176] is collected, in which covers not only different car views, but also their different internal and external parts, and rich attributes. Zapletal and Herout [177] introduced a simple but effective model which uses color histograms and histograms of oriented gradients by a linear regressor for verifying the pair of vehicle images captured by different cameras. Based on the dataset [175], in [178], the appearance attributes of vehicle for a coarse filtering and the Siamese Neural Network for license plate verification to accurately identify vehicles are combined to facilitate progressive vehicle Re-Id.

2.6 Cross-modal Retrieval

The cross-modal similarity is generally established by mapping multi-modal data into a common space. The projection based method is motivated by the fact that multi-modal data are used to represent common objects. For example, in [179], a non-linear dimension reduction technique is introduced for cross-modal retrieval, where bimodal data are represented in a common low-dimensional Euclidean space and the cross-modal similarity is defined by using the Euclidean distance in the learned space. Mao *et al.*[180] propose a cross-modal retrieval algorithm based on parallel field alignment in which heterogeneous data are mapped into a common Euclidean space to measure the similarity between heterogeneous data. Deep learning [181] is also employed to learn a common feature space which could be shared by heterogeneous data. Similar to classical discriminant analysis methods, in [182], two pairwise sets (must-link and cannot-link) on the cross-modal samples are considered to learn a similarity function. More references can be found [12, 183, 184, 185, 186, 187].

The Hamming space is more attractive than the Euclidean space because of its efficiency of searching in a large-scale multi-modal dataset. Some existing cross-modal search algorithms, such as [124, 125, 126], adopt an ideal hash coding restriction that heterogeneous data representing common objects share the same hash coding. Others, such as [24, 128, 188], accept a more relaxed hash coding restriction that heterogeneous data representing common objects share similar binary codes which means the Hamming distance of their binary codes, should be small enough.

Many works of cross-modal search adopt the manifold concept to model multi-modal data, however, the motivations of constructing the manifold are different. Firstly, multi-modal data are treated as an ensemble of homogeneous data, which are modelled as multiple homogeneous manifolds, such as [124, 180, 189, 190]. For example, Gao *et al.*[190] constructed a similarity graph matrix for each uni-modal feature or label feature, and then learned an optimal similarity graph matrix for the given multi-modal data by fusing the similarity information of uni-modal similarity graph matrices and the label information with semi-supervised learning. Secondly, a cross-modal manifold is constructed whereas uni-modal manifolds are omitted, such as [179]. In [179], Mahadevan *et al.*focused on using covariance between the labels of different modal data to measure the similarity between cross-modal data. Lastly, both uni- and cross-modal manifolds are adopted but the information of the uni- and cross-modal manifolds cannot be simultaneously used during the training process. For example, Masci *et al.*[191] use two uni-modal manifolds and one cross-modal manifold to represent bi-modal data; however, the information of these two uni-modal manifolds cannot be used at the same time because of the usage of gradient based optimisation. Zoidi *et* *al.*[192] employed a high-order similarity matrix (similarity tensor) to represent the similarity information of uni- and cross-modal data. Amiri and Jamzad [193] modeled the similarity information of multi-modal data with a supergraph in which the similarity information of uni-modal data is represented by a subgrahp of the supergraph and the similarity information between cross-modal data is modeled by the connected weights between subgraphs.

Besides manifold-related methods, other techniques are also explored for cross-modal retrieval. For example, Masci *et al.*[191] proposed a novel deep learning framework to simultaneously learn multiple hash functions for preserving multi-modal similarity. Song *et al.*[194] proposed another deep learning framework for integrating semi-supervised similarity learning and hash function learning. Lai *et al.*[195] proposed deep neural networks for simultaneous feature learning and hash functions learning. Zhu *et al.*[183] proposed a cross-modal dictionary learning framework for representing multi-modal features with common sparse codes. Pereira *et al.*[12] paid more attention on the role of semantic correlation matching in multi-modal retrieval. More references on hash code learning for similarity search can be seen in [196].

The methods, such as [124, 125, 180, 184], support our view that exploiting the manifold structure is very important for boosting the performance of cross-model retrieval. However, no general frameworks for multi-modalities are available, no higher-order relationships have been considered, and, except for CHMIS[124], all existing methods can hardly be extended to more complex multi-modalities.

2.7 Discussion

The reviewed topics in this chapter range from the general learning framework, learning data association and that the three special applications of data association are relatively broad and support all the proposed works in this thesis. More specifically, a general framework of learning data association is from two aspects: firstly, associating in the same feature space and in multiple sources. Based on such a framework, the classical learning methods, including recognition, regression and ranking can be easily formulated to associate samples for different applications. Secondly, some basic knowledge, which is closely related to the proposed novel methods in the following chapters, is reviewed. Finally, the related works in the three applications, including object tracking, cross-camera re-identification and cross-modal retrieval are reviewed and discussed.

However, despite the results achieved by the exiting methods, there are still many problems that need to be solved and thus automatic data association is far from being established. In the following chapters, novel methods, which are used to overcome the challenges in different directions and levels, are proposed.

Firstly, Chapter 3 introduces a novel method, which is derived from Learn++ methods, to simultaneously tackle the various challenges of object tracking in a single camera. Then, based on the discoveries in Chapter 3, a more advantageous method, which adopts a multi-expert strategy is proposed to improve both the performance and efficiency of existing methods in Chapter 4. Afterwards, Chapter 5 presents a cross-view hashing method to deal with the problem of cross-camera person re-identification. Lastly, by exploiting a framework of a hetero-manifold regularisation, the most difficult task of cross-modal retrieval to associate samples captured by different sensors or platforms are achieved in Chapter 6.

Chapter 3

Sampling Competitive Classifiers for Robust Object Tracking

Object tracking is a traditional and fundamental topic and has wide-ranging applications in designing various systems in computer vision such as surveillance, augmented reality, robotics and human-computer interaction. Recently, adaptive tracking-by-detection approaches [20, 55, 56, 58, 152] based on machine learning methods, which treat the tracking problem as a classification task, are proposed to overcome difficulties in the non-stationary environment (NSE). In some of these methods, several on-line learning tricks are adopted to update the representation of the target [55] or the parameters of a classifier [56] to adapt to NSE. Another type of method is the ensemble learning for tracking [20, 152] which adapts to the NSE through sequentially training the classifiers.

However, firstly, most methods typically can only solve certain challenges but are less effective for others - there is no single tracker that is perfect for all challenges. In addition, to date, developing an effective and efficient method for robust target tracking is still challenging, due to the non-stationary environment (NSE) [197] such as presence of occlusion, background clutter, varying viewpoints, illumination changes, scale changes and camera motion etc.. Moreover, our empirical analysis proves that object tracking is a non-i.i.d. (independent identity distribution) sampling and small dataset problem, which limits the performance of classical machine learning based methods. Motivated by the above three points, this chapter will propose an ensemble learning based tracker in which, most importantly, the members keep the independence with each other for signal target association. The flowchart of visual data association in a view of signal camera is illustrated in Fig. 3.1.

The rest of this chapter is organised as follows. We first give the hypotheses and motivation of this work in Section 3.1. A fast and compact descriptor for image patches is introduced in Section 3.2. The empirical analysis is given Section

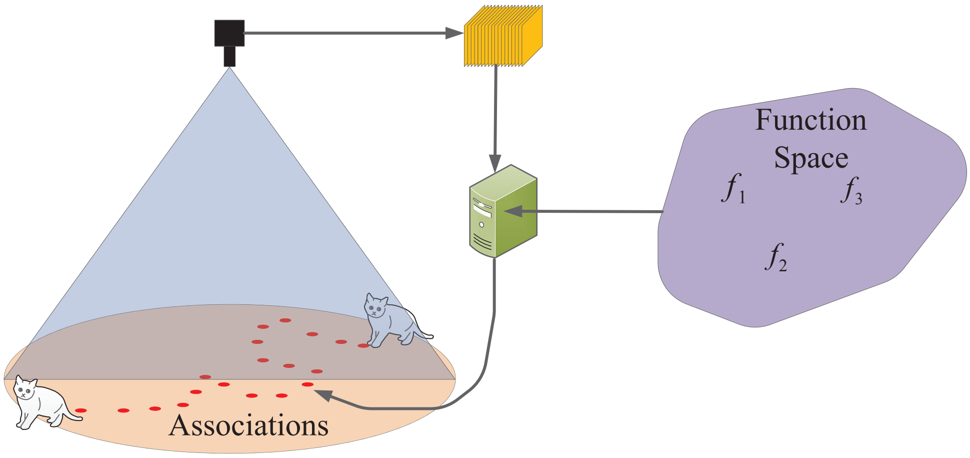


Figure 3.1: Visual data association in a signal camera using multiple classifiers sampled from a same function space.

3.3. Section 3.4 details the proposed Learn++ based method for visual tracking. Experimental results are reported and analysed in Section 3.5. Finally, summary are drawn in Section 3.6.

3.1 Preliminaries

3.1.1 Hypotheses

- The basic hypothesis of object tracking is that very few labelled samples can be given in the first frame while many unpredictable variations and changes are allowed for the object and corresponding environment in the following fames. Hence, the object tracking is a small sample-set problem (few labelled samples) and a un-stationary environment problem (unpredictable variations). The sample distribution changes over time, according to the future surrounding scene and the status of the object.

- We also assume that, before obtaining the first frame, the location and status of target are totally unknown. This is a general assumption in object tracking. At present, there are indeed a few of models which are trained offline to track some popular objects. However, this strategy plays the same role as detection and, in most cases of tracking task, it is impossible to obtain any information of the target.

- Given a dataset with ground truth locations, the empirical analysis about the changes of the sample distribution can be discovered. Moreover, the empirical investigations can be used to reveal the nature of object tracking then to guide the design of more powerful trackers.
- By training on the different groups of samples, the classifiers could be relatively independent with each other and then be utilised to deal with different difficulties.

3.1.2 Motivation

Most machine learning algorithms can learn from data that are assumed to be drawn independently from a fixed but unknown distribution (i.i.d.). However, the i.i.d. assumption cannot be valid in case of the tracking problem. This is because, in object tracking, the classifiers can only be trained on a small sample set and these samples are generated over time. Thus, traditional machine learning methods applied to the tracking problem will fail when there is a "concept drift" in the NSE. The knowledge in this small sample-set which is under-complete is insufficient to describe the overall distribution. The term "***concept drift***" can be used to represent ***changes*** in the underlying ***distribution of samples***. In object tracking, the distribution of samples changes a lot due to the deformation of the object and the change in the underlying environment. It is worth to mention that the sample distribution relies on both the target and the environment and, in most cases, the environment is more likely to be completely unpredictable. Especially during the transition between different difficulties (subproblems), such as from occlusion to varying viewpoints, the samples in the two different situations differ significantly. Thus, the function learned on a fixed sample set previously collected may not reflect the current state of nature and the separability of adopted features will decrease in the new situation. If x are the samples and $y \in \{1, -1\}$ are classes, the whole distribution of the problem at time t, which is characterised by the joint distribution $p^t(x, y)$, can be represented by: the unconditional probability density function $p^t(x)$ and posterior probabilities $p^t(y|x)$. Then, the "concept drift" can be defined as any scenarios where the posterior probability changes over time:

$$KL(p^{t+n}(y|x), p^t(y|x)) < \tau_d \tag{3.1}$$

where t is the time, n is the time step of drift, τ_d is a small value and the Kullback-Leibler (KL) divergence describes the dissimilarity of the two distributions.

In this chapter, we first empirically investigate the "concept drift" problems in object tracking. Then, according to the findings of analysis, a Learn++ (LPP) tracker is proposed to dynamically sample competitive classifiers for robust and

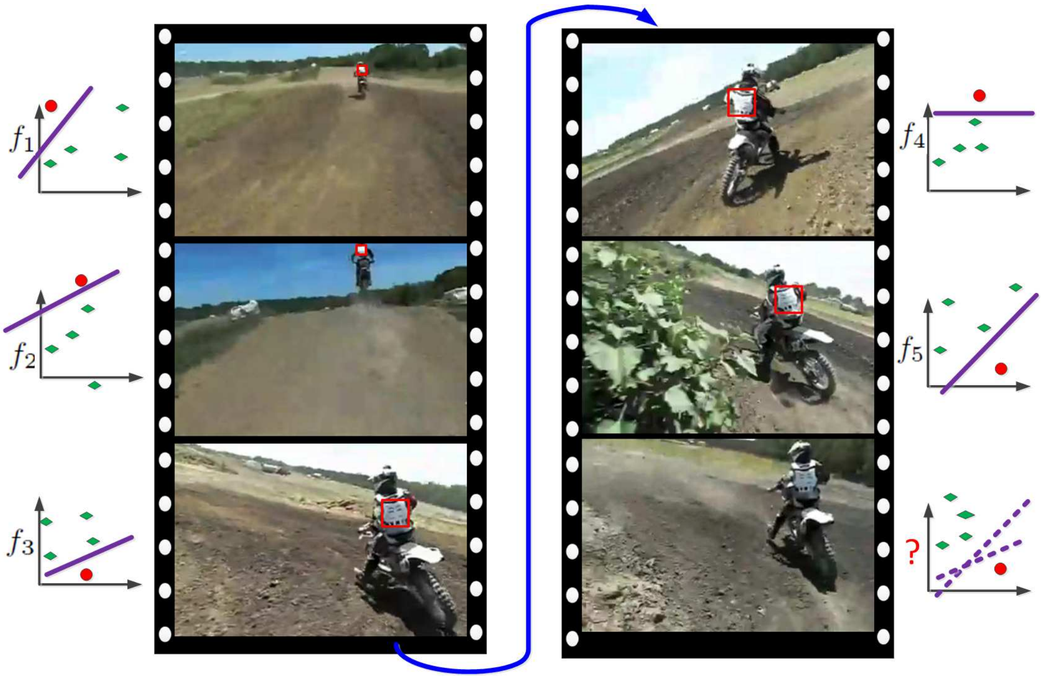


Figure 3.2: Assuming that the best classifiers for the previous frames are available, which classifiers should be used in the current frame (bottom right)? f_2, f_5 or their combination? Also, when the target moves out of view then comes back, which classifiers are the best to be used? This chapter tries to solve these problems in object tracking.

long-term object tracking. In this chapter, "competitive classifier" is used to denote those function which can achieve more advanced performance than others. To increase the efficiency and stability for the model, a competitive strategy is adopted to separately solve various "concept drift" challenges appeared in a same video sequence. Learn++ [198] is a new group of machine learning methods to learn additional information from new data without accessing the original samples and can be used for recognition tasks in very complex situations where new classes would join in. The structure of an optimal classifier in Learn++ is very similar to that in AdaBoost but there are several key differences: AdaBoost runs on a single database and is based on the assumption of an i.i.d. distribution; Some classifiers whose errors are larger than 0.5 will be discarded; Most importantly, the later weak classifier in AdaBoost depends on their previous one; And, the weight distribution of samples is updated using the error of current weak classifier. However, Learn++ can address a set of databases in which the samples are generated by different distributions. To increase the diversity of the model,

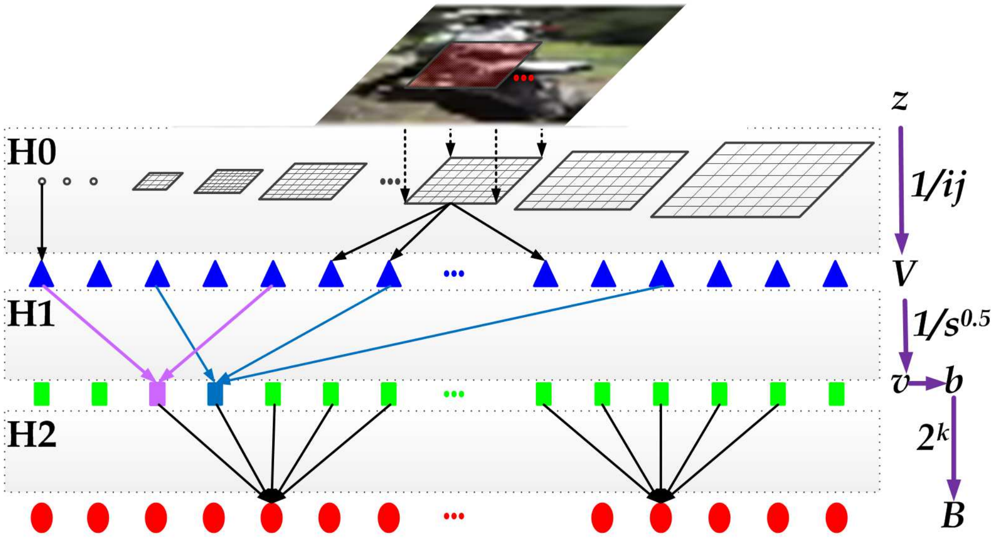


Figure 3.3: The framework of fast structural representation can be chosen by our proposed LPP tracker.

Learn++ keeps all the classifiers as long as they can achieve good performance on a subset. In learn++, the weights of samples are updated using the ensemble performance.

The proposed LPP tracker dynamically maintains a set of basic classifiers $f_i \in \Omega_e^t$ which are trained sequentially on a small sample set. The "concept drift" problems can be solved by adaptively sampling the most suitable classifiers named as competitive subset $\Omega_a^t \subset \Omega_e^t$ as shown in Fig. 3.2(a). These basic classifiers are independent from each other and used to address different sub-problems. For each challenge, the democratic mechanism can be adopted, where all classifiers should compete with each other to be added into a competitive subset to suit the present environment. Next, the optimal classifier $\mathbf{f}^t$ in the present environment can be fast searched in a function space linearly spanned by these basic classifiers in the competitive subset. After the detection guided by motion constraints, the most important samples will be collected to update the classifiers which are trained in the same situation.

3.2 Structural Representation for Image Patch

In object tracking, the two important factors of appearance representation are efficiency and discriminativeness. At present, lots of powerful features have been explored for various tasks but most of them cannot be directly used in object tracking, because they are either computationally expensive or not discriminative in NSE. In this section, a fast structural representation (SR) is introduced to represent an image patch as shown in Fig. 3.3. The advantage of scale invariant SR is that the optimal projections and filters can be decided by selecting the third-layer nodes using the specially designed classifier. As a result, the proposed tracker can keep a good balance between efficiency and performance. SR has a three-level hierarchy: H0-virtual stage of filtering, H1-stage of random projection, and H2-stage of encoding.

H0: filtering. Given a patch $z \in R^{I\times J}$ in which I and J denote the numbers of rows and columns respectively, a set of rectangular smoothing filters $\{h_{i\times j} \in R^{i\times j}, 1 \leq i \leq I, 1 \leq j \leq J\}$ are defined, for which all entries of each filter $h_{i\times j}$ equal $1/(i \times j)$. In total, there are $I \times J$ filters, each of which is convolved with the entire patch and produces $I \times J$ values. So, the dimension $n_V = (IJ)^2$ of the original feature $V \in R^{n_V}$ is very high and much information is redundant.

H1: random projection. Next, a sparse random matrix is used for dimensionality reduction, which is defined as: $P(i,j) = 1$ with the probability $1/(2s)$, $P(i,j) = -1$ with the probability $1/(2s)$ and $P(i,j) = 0$ with the probability $1 - 1/s$, where p is the probability. In [199], Achlioptas pointed out that this matrix with $s = 2$ or 3 satisfies the Johnson-Lindenstauss lemma. Compressive sensing theory ensures that the extracted features preserve almost all the information of the original image patch. In this chapter, we set $s = n_V/4$.

Thus, the value $v_k \in R$ projected by each row of the random matrix is: $v_k = P(k,\cdot)V$. The stage of filtering can be considered as virtual. This is because most of the entries are zeros so that a large proportion of the filter needs not to be calculated. We only need to store the nonzero entries and the positions of their corresponding rectangular filters in an image. Moreover, v_k can be efficiently computed by using $P(k,\cdot)$ to sparsely measure the rectangular features which can be efficiently computed using the Integral Image. For patches with a different size $z^* \in R^{I^*\times J^*}$, the number of rectangular features will be different. In fact, we need not to resize the patch. Applying a scale $IJ/(I^*J^*)$ to the locations of elements in V^* will be feasible to realize scale invariance. For each value v_k, its mean μ_k and variance σ_k of positive samples will be computed when its corresponding classifier is trained.

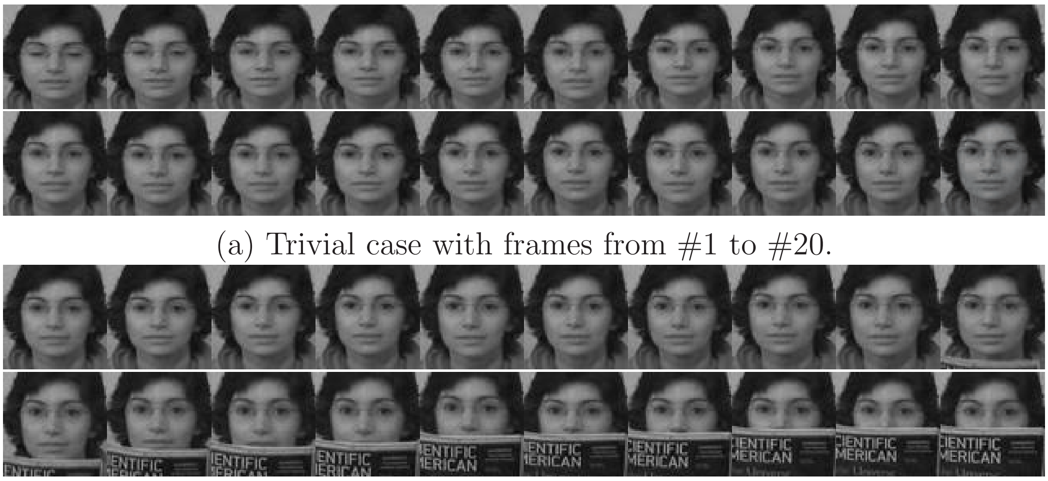
H2: encoding. The third layer is constructed similarly to Fern [200], in which a feature was calculated by comparing two randomly selected pixels in a patch. However, directly comparing two pixels is very sensitive to noise, especially when the two pixels are located around an edge. Normally, to eliminate this drawback, filters will be used firstly. Instead of comparing the pixels, the value v_k can be considered as basic cues. Thus, for each projected value v_k, a binary feature can be defined as: $b_k = \Gamma\lfloor v_k \in [\mu_k - \sigma_k, \mu_k + \sigma_k]\rfloor$, where $\Gamma\lfloor\rfloor$ is the indicative function. Each node in the third layer consists of a set of bits, and in this chapter, the size of the set n_b is set to 7. Thus, the node of the deepest layer in the hierarchy is defined as: $B = \sum_{k=0}^{n_b} 2^k b_k$.

In summary, firstly, SR is a simple but powerful and efficient representation for image patch, which has a similar formulation to the convolutional neural network algorithm [38] in the first several layers. We can see that the first layer is to calculate local mean values with various scales and the second layer, which combines patches in different locations, is to capture the global structure of objects. Thus, SR can extract the local and global features simultaneously and be invariant to partial occlusion. The binary encoding in the third layer, to which a naive Bayes classifier can be directly used, plays the same role as activation function. Secondly, if the values of sparse projection are fixed so that adjacent rectangles are combined and no binary coding is used, then SR extracts the Harr-like features. Next, if the size of a filter is fixed, the number of nonzero entries of random projection is set to two, and the two weights are also opposite numbers, then SR will become the classical framework of Fern [200]. Moreover, besides the natural properties of intensity, scale and partial occlusion invariance, SR is very computationally efficient because of the usage of sparse projection and binary coding. Finally, each node in the third layer can capture a certain internal structure of objects. By selecting the nodes, different information can be used. Therefore, various challenges in object tracking can be handled by selecting the abundant and diversified features.

3.3 "Concept Drift" in Object Tracking

In this section, the "Concept Drift" problem in object tracking is investigated on a public dataset which contains 50 sequences [21]. From the theoretical definition of "Concept Drift", to investigate the "drift", the necessary steps are firstly to fit the **sample distribution** and then to calculate the **changes** of the distribution over time. However, it is impractical to directly obtain stable distributions, because, normally, the dimension of a sample is very high. Therefore, we seek to calculate the distribution of Euclidean distance between any pair of samples. The intuition behind is that the distribution of distances or similarities can reflect the distribution of samples. For example, if two variances x_1 and x_2 are independent, standard normal random variables, then the quantity $||x_1 - x_2||^2$ is distributed according to the one degree chi-squared distribution.

Moreover, we observe that most classical methods can handle the translation in simple situations (trivial cases shown in Fig. 3.4 (a)) without large deformation or other challenges, but are unable to cope with complex "concept drift" situations (drift cases shown in Fig. 3.4 (b)). However, at present, the differences of appearance features between the trivial and the drift situations have not been



(a) Trivial case with frames from #1 to #20.

(b) Drift case with frames from #13 to #32.

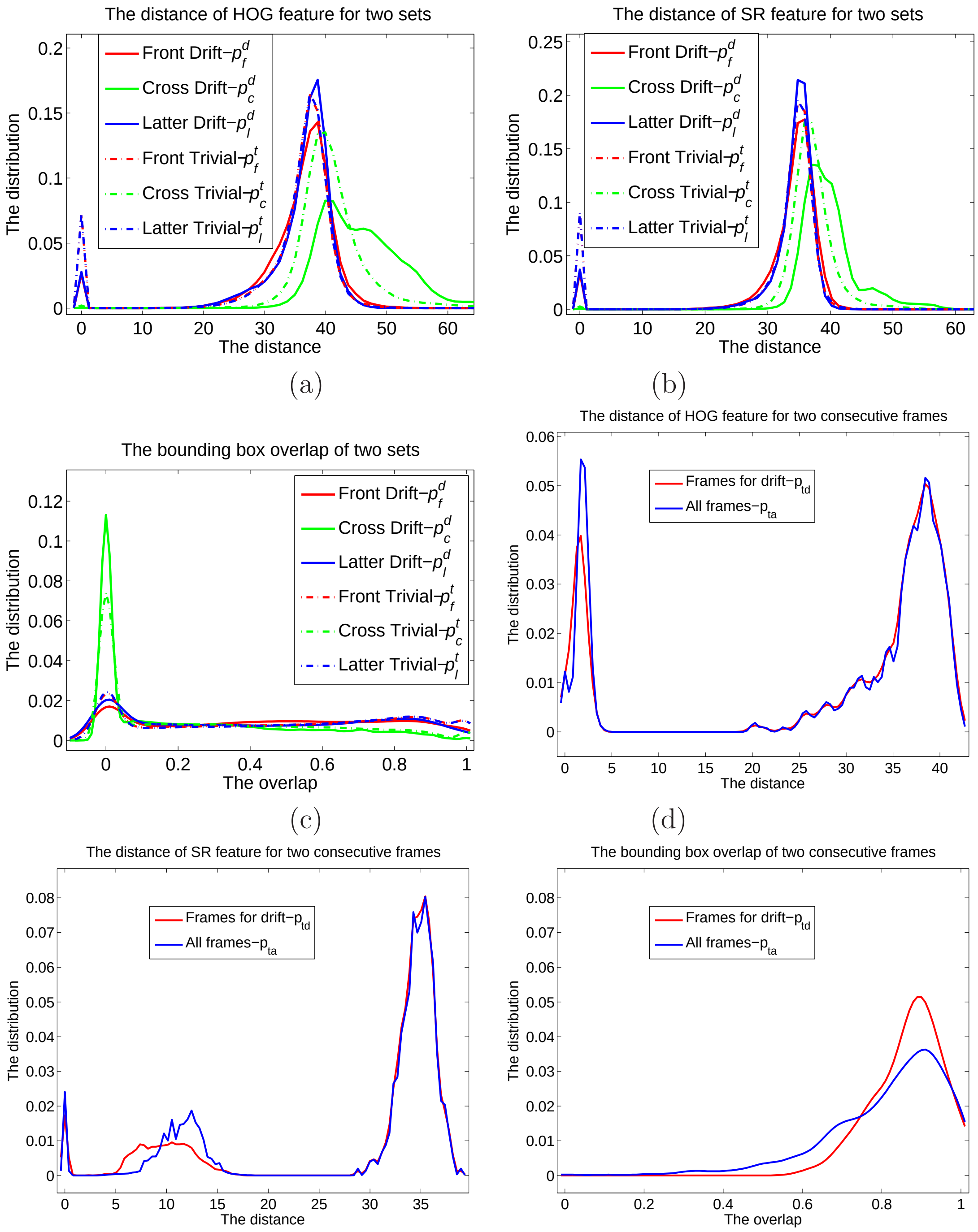
Figure 3.4: Two short videos in FaceOcc1 [21] to show the difference between the trivial case (a) and drift case (b). Each short video consisting of **front set** (first row) and **latter set** (second row) is annotated by three numbers. Take the second video for example, the first number #13 is the index of the starting frame, the second one #22 denotes the moment when the object begins to change and the third one #32 is the index of the ending frame. The two cases are just used to show the differences between them. In fact, in the procedure of real tracking, it is impossible to know what kinds of challenges occur in the following frames.

studied. The study of the differences can facilitate understanding the nature of object tracking and guide the design of more robust methods. To achieve this, 100 short videos (A sample is shown in Fig. 3.4 (b)) with the challenges excluding fast motion[1] from the dataset are manually identified and other 984 short videos (A sample is shown Fig. 3.4 (a)) are randomly selected from the rest of the dataset. Thus, we can see that the short videos are divided into two parts by the three numbers: the front set before drift and the latter set after the drift.

We illustrate the experience analysis in Fig 3.5[2]. Firstly, based on the two sets, three groups of feature distances $d = ||x_1 - x_2||$ can be calculated depending on which sets the pairs of samples come from. Then, for each group of distances,

[1] This problem can be solved easily by dense sampling using the tracking-by-detection methods.

[2] (a) The distance distribution of the HOG feature between the two sets. (b) The distance distribution of the SR feature between the two sets. (c) The bounding box overlap distribution between the two sets. (d) The distance distribution of the HOG feature for two consecutive frames. (e) The distance distribution of the SR feature for two consecutive frames. (f) The bounding box overlap distribution of two consecutive frames.



(a) (b)

(c) (d)

(e) (f)

Figure 3.5: Statistical analysis of the two situations in object tracking: "concept drift" and the trivial case.

a distribution of them can be obtained. For the "drift" cases, p_f^d denotes the distribution of distances between two samples both within the front set, p_l^d denotes the distribution of distances between two samples both within the latter set and p_c^d denotes the distribution of distances across the two sets. For the trivial cases, we can also calculate the three distance distributions including p_f^t, p_l^t and p_c^t. In this analysis, two features, HOG [201] and SR, are used. The comparisons of six distributions are shown in Fig. 3.5 (a) and (b) for HOG and SR, respectively. Moreover, besides the feature distance, the distribution of the overlap of bounding boxes in the two sets is also investigated. The comparisons of six distributions for the overlap of bounding boxes are given in 3.5 (c). In contrast, for the feature distance and overlap of bounding boxes between any two consecutive frames, distributions of them are also calculated and the comparisons are referred to 3.5 (d), (e) and (f), respectively. Finally, the quantitative analysis $\int_d |p_1(d) - p_2(d)| \bigtriangledown d$ between two distributions are given in Table 3.1.

From the statistical analysis given in Fig. 3.5 and Table 3.1, the following observations can be obtained:

- The six distributions of the two types of features HOG and SR are similar. It indicates that the statistics reflect the nature of variations of objects and the environment.

- The feature distances across the two sets are obviously larger than the ones within a set. Next, from Table 3.1, we can see the distribution discrepancy between p_c^d and p_c^t both across the two sets is almost three times larger than the one between p_f^d and p_f^t within the front set. Furthermore, the distribution discrepancy between p_f^d within set before drift and p_c^d across the two sets is quite large. Because the distance distribution can reflect the intrinsic structure of samples, the analysis demonstrates that the sample distributions of the two sets distinguish from each other, especially for the cases of "concept drift" situations. To some extent, the assumption that the samples are generated by different distributions over time is verified.

- In contrast, the distributions of feature distance between the two consecutive frames for the drift and trivial conditions are similar and the distribution discrepancy is also small, no matter which type of representation is adopted.

- Furthermore, the overlap distributions of bounding boxes between the two consecutive frames are also similar for the drift and trivial situations.

In brief, by the empirical study, we can conclude that the object tracking is a non-i.i.d. sampling and small set problem. Therefore, the ability of classical machine learning is limited. And, it is unsuitable to classify the samples in the following set after the "concept drift", by a function which is trained on a small sample set generated by different latent distributions. However, the procedure of tracking-by-detection can be connected together using the motion information (two consecutive frames). In this chapter, a competing strategy is adopted to select the most suitable classifiers to address the current special problem and the motion constraint between two consecutive frames is used to supervise the training and updating of classifiers.

Table 3.1: The discrepancy $\int_d |p_1(d) - p_2(d)| \bigtriangledown d$ between the two distributions $p_1(d)$ and $p_2(d)$ shown in Fig. 3.5. The largest discrepancy approximates 2.

Comparison	p_f^d p_f^t	p_c^d p_c^t	p_l^d p_l^t	p_t^d p_{ta}	p_c^d p_{td}
HOG feature	0.2326	0.7488	0.1835	0.1408	1.2590
SR feature	0.2142	0.5976	0.1548	0.1817	1.1441
Bounding box	0.1579	0.2037	0.1423	0.3518	0.6858

3.4 Learn++ for Solving the Problem of "Concept Drift"

Based on the above observations, in this section, a LPP tracker is learned to solve the numerous problems of "Concept Drift" which is guided by motion constraints. Assume the classifier set Ω_e^t consists of a competitive set Ω_a^t and its complementary set Ω_c^t. We have $\Omega_c^t \bigcup \Omega_a^t = \Omega_e^t$, $n_e^t = |\Omega_e^t|$ and $n_a^t = |\Omega_a^t|$, where $|\cdot|$ denotes the number of members of the set. W_i^t denotes the historical weights of all existing $f_i \in \Omega_e^t$.

3.4.1 Motion constraints

The similarity transform is used as the motion model for our system. There are four parameters: (1) horizontal and vertical coordinates; (2) horizontal and vertical scales. a is the state of target describing its motion.

Based on the observation in two consecutive frames, Optical flow (OP) [17] is used to construct the motion model in our framework. OP follows the movement of the target frame by frame and has very high flexibility because no historical information is considered. The motion model $p(a^t|a^{t-1})$ reflects the motion characteristic of the target by predicting the current state a^t based on the previous state a^{t-1}. If $p(a^t|a^{t-1}) > \tau_1$, then a^t is a valid result no matter whether "concept drift" occurs or not. The P-N constraints [1], $p(a^t|f_i)$, were proposed to estimate the confidence of the ith classifier f_i. If $p(a^t|f_i) > \tau_2(i)$ where $f_i \in \Omega_a^t$, the outcome of classifier f_i is validated. This triggers the application of the P-N constraints that exploit the structure of the data. From the manifold perspective, P-N constraints maintain a purified sub-manifold for positive and negative samples. All samples far from such a sub-manifold will be ignored. So, P-N constraints are used by LPP tracker to guarantee the stability of each classifier. If $p(a^t|\Omega_a^t) > \tau_2$, where $p(a^t|\Omega_a^t) = \max_i p(a^t|f_i)$, there is no occurrence of drifting. The classifier constraints are designed based on the observations of the two sets in the analysis of "concept drift".

3.4.2 Objective function

In frame t, x_l^t and y_l^t denote the structural representation and the label of image patch z_l^t, respectively. Each entry $x_l^t(i)$ contains the n_B number of nodes $\{B_{i,j} : j = 1, \cdots, n_B\}$, for the classifier f_i. Also, X^t is the set of collected samples and $n_X^t = |X^t|$. The distribution of samples D^t will be calculated according to the results of old classifiers and used to describe the importance of samples. For simplicity, we define $f_i(x_l^t) = f_i(x_l^t(i))$, $w^t = (w_1^t, \cdots, w_{n_a}^t)$ and $\Omega_e^t = (f_1, \cdots, f_{n_a})^T$.

Our goal is to find an optimal classifier $\mathbf{f}^t$ with most discriminative features in the function space $\mathcal{H}^t$ linearly spanned by a set of classifiers Ω_e^t which are trained in previous frames, where $\mathcal{H}^t = \{h^t : h^t = w^t \Omega_e^t\}$. Moreover, to improve the efficiency of the system, the weights w^t are required to be sparse so that most basic classifiers are not used in the current frame. Thus, in theory, the objective function is defined as:

$$\mathbf{w}^t = \arg\min_{w^t} \sum_l L(h^t(x_l^t), y_l^t) + \lambda \left|\left|w^t\right|\right|_0 \tag{3.2}$$

where L and λ are the loss function and regularization parameter, respectively. Therefore, we obtain the hypothesis as:

$$\mathbf{f}^t = \mathbf{w}^t \Omega_e^t \tag{3.3}$$

The optimal classifier $\mathbf{f}^t$ can be used to detect the object in the current frame (new environment). The final classification for each image patch x_i^t is achieved as: $y_l^t = sign(\mathbf{f}^t(x_l^t))$. Eqn. 3.2 cannot be optimised directly, due to that the true label y_l^t of image patch x_l^t is unknown. However, based on the assumption of the "concept drift" (Eqn. 3.1), we can approximate to the optimal solution using the classifiers that have yielded good performance in recent n frames or in the same situations by Learn++. Learn++, which is an ensemble of classifiers originally developed for incremental learning. It specifically seeks the most discriminative information from each sample set (sub-problem) through sequentially generating an ensemble of classifiers which are trained on individual data sources and carry complementary information. It can still achieve a statistically significant improvement by combining these classifiers which are finely tuned for the given problem. In this chapter, the "concept drift" problem in the NSE is solved through two steps: (1) selection of the active subset $\varOmega_a^t$; (2) optimal approximation $\mathbf{f}^t = \mathbf{w}^t \varOmega_a^t$. In the following, how to train basic classifiers f_i and how to approximate the optimal classifier $\mathbf{f}^t$ by calculating w_i^t will be introduced.

3.4.3 Basic classifier

The Naïve Bayesian are used as the basic classifiers in our proposed system. These classifiers will be trained on different datasets thus the parameters of classifiers will be different to capture varied information. Thus, f_i can be defined by posterior probabilities by combining n_B nodes (assuming an uniform prior $p(y)$):

$$f_i(x_l) = \arg\max_y p(y|x_l(i)) \tag{3.4}$$

where $p(y|x_l(i)) \propto \prod_j^{n_B} p(x_l(i,j)|y)$. Therefore, for each f_i, the posterior probabilities will be trained and updated to adapt to the changes of the environment and the object by calculating and updating the class conditional distribution $p^t(B_{i,j}|y)$ of each Fern.

Training. The parameters of SR for each classifier f_i will be generated randomly. Once generated, these parameters will be fixed during the whole lifespan of the classifier f_i. At frame t, based on a set X_1^t with all positive samples and 2000 negative samples in the set X^t and distribution D^t, we can define two quantities which are used to train or update classifiers: $N^t(y, B_{i,j}) = \sum_l D_l^t \Gamma \lfloor x_l^t(i,j) = B_{i,j} \rfloor \Gamma \lfloor y_l^t = y \rfloor$ and $N^t(y) = \sum_l D_l^t \Gamma \lfloor y_l^t = y \rfloor$, where $x_l^t \in X_1^t$. Other negative samples are used to evaluate the classifier. Therefore, $\tau_2(i) = max_{a_l^t} p(a^t|f_i)$, where a_l^t denotes the corresponding states of negative samples $x_l^t \in X^t / X_1^t$. Thus, the class conditional distributions for f_i are calculated by:

$$p^t(B_{i,j}|y) = \frac{1 + N(y, B_{i,j})}{1 + N(y)} \tag{3.5}$$

where $N(y, B_{i,j}) = N^t(y, B_{i,j})$ and $N(y) = N^t(y)$.

Learning. If f_i has been used in frame t. The set X^t can be used to update the class conditional distribution $p^t(B_{i,j}|y)$ so as to adapt to the changes by:

$$N(y, B_{i,j}) \Leftarrow N(y, B_{i,j}) + N^t(y, B_{i,j}); N(y) \Leftarrow N(y) + N^t(y) \tag{3.6}$$

By recalculating Eqn. 3.5, the updated class conditional distributions are obtained.

3.4.4 Tracking by detection

At the beginning ($t = 0$), random Ferns f_1 needs to be trained according to the selected target in the first frame and we can directly jump to the sample collection step. At frame $t(t > 0)$, assume that $\mathbf{f}^t(x_l^t)$ is the ensemble learned at time $t-1$ and the location $\mathbf{a}^{t-1}$ has been determined. The goal is to detect the location of the target and evaluate the state of the target. To achieve this, the following steps which are similar to most tracking-by-detection approaches are processed sequentially. First, by applying the sliding window method to the current frame, the classifier in the competitive subset is used to classify each patch of this frame. Second, the OP method is used to compare the two targets in the two successive frames. Third, the probabilities $p(a^t|\mathbf{a}^{t-1})$ and $p(a^t|\Omega_a^t)$ are calculated. Fourth, all states classified as positive samples by $\mathbf{f}^t$ will be fused and the optimal state $\mathbf{a}^t$ with the highest confidence in the current frame will be obtained. Finally, the classifiers will be updated according to the present performance. The entire procedure is organised as in Algorithm 1. According to the results of current frame, how to search optimal classifier for next frame will be given in the following section.

Algorithm 1 LPP tracker

```
Initialization Define a target in the first frame and build a classifier $f_1$.
Repeat $t = 1, \cdots$
(0) Capture a new frame. If no frame: Exit.
(1) Run each classifier $f_i \in \Omega_a^t$ of the competitive subset on the present frame.
(2) Combine the results $\mathbf{f}^t$ according to Eqn. 3.3 and obtain the best results: $\mathbf{a}^t$.
(3) If $\mathbf{a}^t$ is valid target: Compute the probabilities $p(a_t|\mathbf{a}_{t-1})$ and $p(a_t|f_i)$.
(4)     If $p(\mathbf{a}^t|\mathbf{a}^{t-1}) > \tau_1$: Collect and weight samples $X^t$,
(5)         If $p(\mathbf{a}^t|\Omega_a^t) > \tau_2$: Update the old classifiers $f_i$,
(6)         Else If $p(\mathbf{a}^t|\Omega_c^t) > \tau_2$: Revive a classifier from $\Omega_c^t$;
(7)         Else: Train a new classifier $f_{n_a^t+1}$.
        End
    End
(8) Resampling and evaluate the classifiers.
Return Update the classifier $\mathbf{f}^{t+1}$ and set the state $\mathbf{a}^t$, Go To (0).
```

3.4.5 Collecting and weighting samples

If $p(\mathbf{a}^t|\mathbf{a}^{t-1}) > \tau_1$ is satisfied, it means that the tracked target is valid and can be used to update the set of classifiers. Otherwise, when no valid target is in the current frame, we can directly jump to the classifier sampling step.

Collecting. The sample set X^t is constructed as follows: If the overlap of $\mathbf{a}^t$ and a_l^t exceeds 0.5, the patch z_l^t of state a_l^t will be considered as the positive

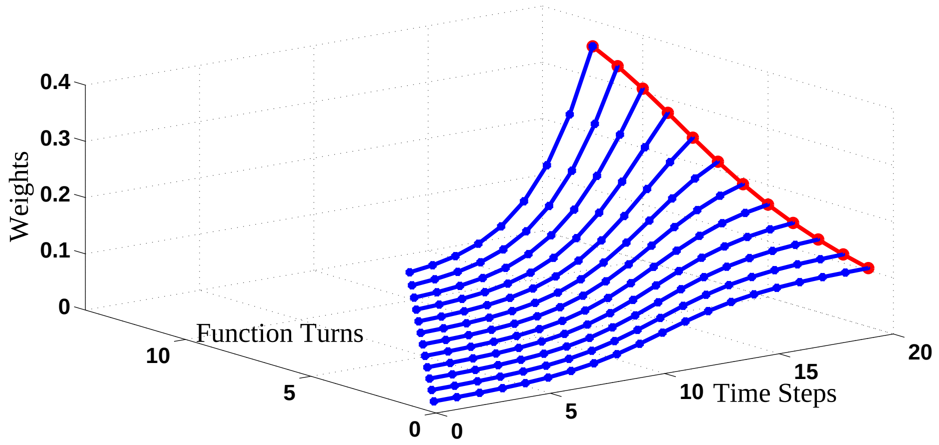


Figure 3.6: Signoidal weights used in Eqn. 3.9. λ_1, λ_2 and η are set to 0.5, 10 and 8, respectively.

sample; otherwise if the overlap of $\mathbf{a}^t$ and a_l^t is lower than 0.2, it is considered as the negative sample. Also, according to the fused results $\mathbf{a}^t$, 400 positive samples will be generated by the affine warping of the selected patch $\mathbf{z}^t$ to increase the richness of positive samples.

Weighting. At the beginning ($t = 0$), the distribution of samples D^t used to train the first classifier is set to be equal to $1/n_X^t$. If ($t > 0$), the distribution of patches in the tth frame will be computed. Firstly, the current ensemble $\mathbf{f}^t$ is evaluated on the new patches X^t: $E^t = \frac{1}{n_X^t}\sum_{l=1}^{n_X^t}\Gamma\lfloor sign(\mathbf{f}^t(x_l^t)) \neq y_l^t\rfloor$. Secondly, sample weights D_l^t of x_l^t are defined by:

$$D_l^t = \begin{cases} E^t, & sign(\mathbf{f}^t(x_l^t)) = y_l^t; \\ 1, & otherwise. \end{cases} \tag{3.7}$$

Finally, set $D_l^t \Leftarrow D_l^t/\sum_{l=1}^{n_X^t} D_l^t$. Normalizing the error weights by their sum then provides us the updated penalty distribution. Samples of the new environment x_l^t, which are not recognised by the existing knowledge base $\mathbf{f}^t$, are identified.

3.4.6 Sampling the classifiers

In this section, how to approximate the optimal classifier based on Learn++, according to the recent performance of the ensemble, will be introduced. The strategies include learning the new samples to the existing classifiers, reviving the old classifiers, training a new classifier and sampling all of them. If the current competitive subset can deal with the changes, the optimal classifier in the next frame has the same basic classifiers. To increase the adaptivity, the new samples will be learned by existing classifiers. For each $f_i \in \Omega_a^t$, if $p(\mathbf{a}^t|f_i) > \tau_2(i)$, then f_i will be updated by the samples X^t according to their distribution D_l^t following Eqn. 3.6. Otherwise, reviving the old classifiers or training a new classifier will be considered. This step of selecting the samples is important for keeping the stability of the classifiers. From the analysis of "concept drift", we can see that most classifiers are trained on a small dataset. If some samples generated by different distributions are used to update the classifiers, the intrinsic structure of the classifiers will be easily destroyed. Thus, the model will fail to deal with any challenges even if the classifiers have incorporated the history information.

Reviving. If $p(\mathbf{a}^t|\Omega_a^t) < \tau_2$ and $p(\mathbf{a}^t|\mathbf{a}^{t-1}) > \tau_1$, due to the "concept drift", it means that the current ensemble cannot deal with the changes. So, a new set of basic classifiers need to be built. New classifiers will be added into the ensemble so that the optimal classifier will be searched in a new set of classifiers. Firstly, all existing classifiers $f_i \in \Omega_c^t$ will be used to check whether the current appearance can be recognised or not by old classifiers. If a similar "concept drift" has occurred before, an old classifier can be revived. This procedure is efficient to compute because no sliding window is needed. If $p(\mathbf{a}^t|\Omega_c^t) > \tau_2$, where $p(\mathbf{a}^t|\Omega_c^t) = \max_i p(\mathbf{a}^t|f_i)$, there exists one classifier f_i that can recognise the current state. Thus, this classifier f_i will be revived directly without adding a new one. Otherwise, a new classifier will be trained and added to the ensemble following Eqn. 3.5.

Resampling. No matter whether the valid target has been detected in the current frame or not, some classifiers killed before will be revived through the resampling procedure according to the historical weights W_i^t. This will increase the diversity and avoid the local optimal solution. The adaptive rejection sampling method [202] is employed to realize this step.

Evaluating. After learning, reviving or training, the competitive set of basic classifiers Ω_a^{t+1} is fixed. For finding the optimal classifier for the next frame, evaluating all classifiers $f_i \in \Omega_a^{t+1}$ on the new data X^t is necessary. Firstly, the error of each $f_i \in \Omega_a^{t+1}$ on weighting samples is defined as:

$$\varepsilon_i^t = \sum_{l=1}^{n_X^t} D_l^t \Gamma \lfloor f_i(x_l^t) \neq y_l^t \rfloor \tag{3.8}$$

Thus, $\varepsilon_i^t \Leftarrow \varepsilon_i^t/(1-\varepsilon_i^t)$. ε_i^t can be considered as the performance of the function. If f_i contributes mostly to the error of the ensemble classifier $\mathbf{f}^t$, ε_i^t will be larger than others. Secondly, for incorporating the performance on recent frames, a sigmoidal error weight is defined as: $\gamma_i^t(m) = 1/(1 + \exp(\lambda_1 m - \lambda_2)), \{m = 0, \cdots, n_e^t + \eta - i\}$, where λ_1, λ_2 are two parameters, η is the time step and i is

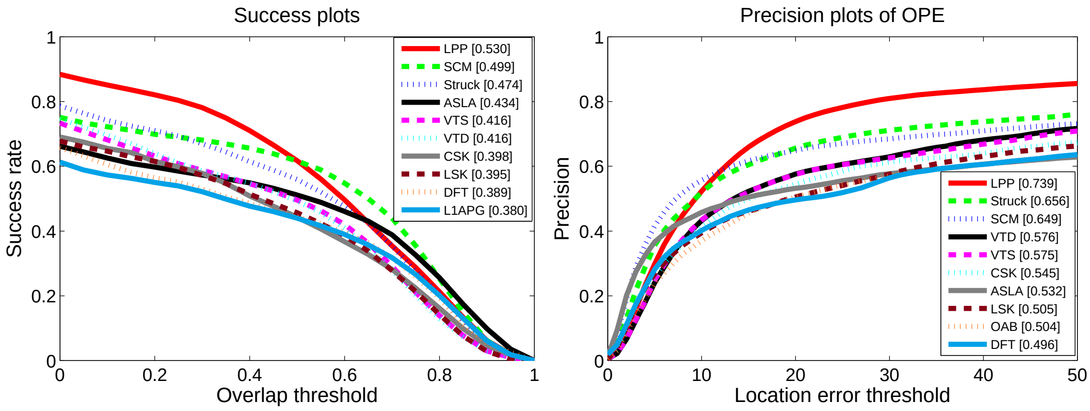


Figure 3.7: The success plots and AUC rankings of 10 tracking methods.

the index of the function in the ensemble. Thus, the weights are normalised so that $\gamma_i^t(m) \Leftarrow \gamma_i^t(m)/\sum_m \gamma_i^t(m)$ (see Fig. 3.6). Finally, the error of $f_i \in \Omega_a^{t+1}$ is weighted with respect to time so that recent competence (error rate) is considered more heavily for categorizing knowledge. The weighted errors are defined by:

$$\beta_i^t = \sum_{m=0}^{n_e^t+\eta-i} \gamma_i^t(n_e^t+\eta-i-m)\varepsilon_i^{t-m} \tag{3.9}$$

Thus, we calculate the classifier voting weights: $w_i^t = \log 1/\beta_i^t$ and normalise them: $\mathbf{w}_i^{t+1} \Leftarrow w_i^t/\sum_i w_i^t$. The instant voting weights can be used to update the historical weights according to $W_i^{t+1} \Leftarrow (1-\alpha)W_i^t + \alpha\mathbf{w}_i^t$, where α is the updating rate and is set to 0.05.

Optimal approximation. To balance the increase of the diversity of the ensemble and efficiency of the model, the following conditions will be considered: (1) For any $f_i \in \Omega_a^{t+1}$ with $\mathbf{w}_i^{t+1} < \tau_3$, the classifiers will be killed and moved to Ω_c^{t+1}; (2) For any $f_i \in \Omega_c^{t+1}$ with $W_i^{t+1} < \tau_3$, the classifiers will be deleted for ever. Because the size of Ω_a^{t+1} is much smaller than Ω_e^{t+1}, the weights $\mathbf{w}^{t+1}$ are sparse. Therefore, the optimal approximation classifier used in the next frame will be defined by:

$$\mathbf{f}^{t+1} = \sum_{f_i \in \Omega_e^{t+1}} \mathbf{w}_i^{t+1} f_i \tag{3.10}$$

3.5 Experiments

In our experiments, the greyscale images are taken as input. τ_1, τ_2 and τ_3 are set to 0.75, 0.9 and 0.05, respectively. LPP tracker will be compared with 13 state-of-the-art methods, including Struck [59], SCM [136], VTS [148], VTD [203], CSK [204], ASLA [135], DFT [205], L1APG [206], LSK [207], MIL [58], OAB [57], Frag [131] and CT [138]. Most of them were recently proposed and ranked in the top list in the experimental comparison in [21]. Our experiments follow the setting in [21]. Each sequence is repeated 5 times with different random seeds by LPP tracker, and the median results are reported. To compare with numerous methods, two types of metric are used to evaluate the different methods. (1) Center location distance: following [21], if the distance between the center of the tracked patch and the center of ground truth is within 20 pixels, the estimated target is considered as correct. Thus, the precision can be defined as the proportion of the correctly tracked frames to the total number of frames. The precision rankings of the 10 methods on the 21 videos are given in Table 3.2. (2) Bounding box overlap: the success plot shows the ratios of successful frames at the thresholds varying from 0 to 1. The area under curve (AUC) [21] of each success plot is used to rank the tracking algorithms. Wu et al. [21] pointed out that AUC is fairer and more accurate because it measures the overall performance. Both success plots and AUC rankings are shown in Fig. 3.7 and some screenshots are shown in Fig. 3.10.

3.5.1 Comparison with state-of-the-art methods

Firstly, taking the sequence *singer1* (624×352) for example, CT, LPP tracker and Struck take the average time per frame of 17ms, 55ms and 209ms respectively on a Dell M4600 (Intel Core 2.8GHz and 8G RAM). Thus, LPP tracker can address most real-world problems in real-time (more than 18 FPS).

Secondly, from Fig. 3.7, LPP tracker can achieve the best performance among all the 13 methods (Only top 10 are shown in the figure) on most of the challenges, from the perspectives of both center location distance and bounding box overlap. From this figure, we can see that LPP outperforms other methods obviously both at the overlap range from 0 to 0.6 and the location distance range from 15 to 50. Generally, template matching based methods including SCM and ASLA achieve better results both at higher overlap rates and nearer pixel distances. In fact, at this situation, LPP tracker performs very closely to the best method and the performances of all methods are nearly the same.

Thirdly, the performance comparisons on different challenges are given in Table 3.2, Fig. 3.8 and Fig. 3.9. On the one hand, using the bounding box overlap criterion, LPP tracker has a great advantage over the other methods except for the challenges of scale variation and low resolution. Although LPP tracker is not the best method for these two challenges, it still ranks in the top three. It is also meaningful to point out that LPP tracker achieves much better results on the challenges of fast motion, motion blur and out of view than other methods. In fact, the first motivation of LPP tracker is to address the challenges of appearance

Challenges	LPP	Struck	SCM	ASLA	CSK	L1APG	OAB
IV	0.677	0.558	0.594	0.517	0.481	0.341	0.388
OPR	0.725	0.597	0.618	0.518	0.540	0.478	0.503
SV	0.720	0.639	0.672	0.552	0.503	0.472	0.541
OCC	0.719	0.564	0.640	0.460	0.500	0.461	0.483
DEF	0.734	0.521	0.586	0.445	0.476	0.383	0.470
MB	0.663	0.551	0.339	0.278	0.342	0.375	0.360
FM	0.679	0.604	0.333	0.253	0.381	0.365	0.416
IPR	0.677	0.617	0.597	0.511	0.547	0.518	0.471
OV	0.698	0.539	0.429	0.333	0.379	0.329	0.454
BC	0.693	0.585	0.578	0.496	0.585	0.425	0.446
LR	0.501	0.545	0.305	0.156	0.411	0.460	0.376
Challenges	VTD	VTS	DFT	LSK	CT	MIL	Frag
IV	0.557	0.573	0.475	0.449	0.359	0.349	0.326
OPR	0.620	0.604	0.497	0.525	0.394	0.466	0.444
SV	0.597	0.582	0.441	0.480	0.448	0.471	0.407
OCC	0.545	0.534	0.481	0.534	0.412	0.427	0.475
DEF	0.501	0.487	0.537	0.481	0.435	0.455	0.468
MB	0.375	0.375	0.383	0.324	0.306	0.357	0.288
FM	0.352	0.353	0.373	0.375	0.323	0.396	0.346
IPR	0.599	0.579	0.469	0.534	0.356	0.453	0.401
OV	0.462	0.455	0.391	0.515	0.336	0.393	0.355
BC	0.571	0.578	0.507	0.504	0.339	0.456	0.421
LR	0.168	0.187	0.211	0.304	0.152	0.171	0.163

Table 3.2: The precision rankings of 14 tracking methods on challenging sequences.

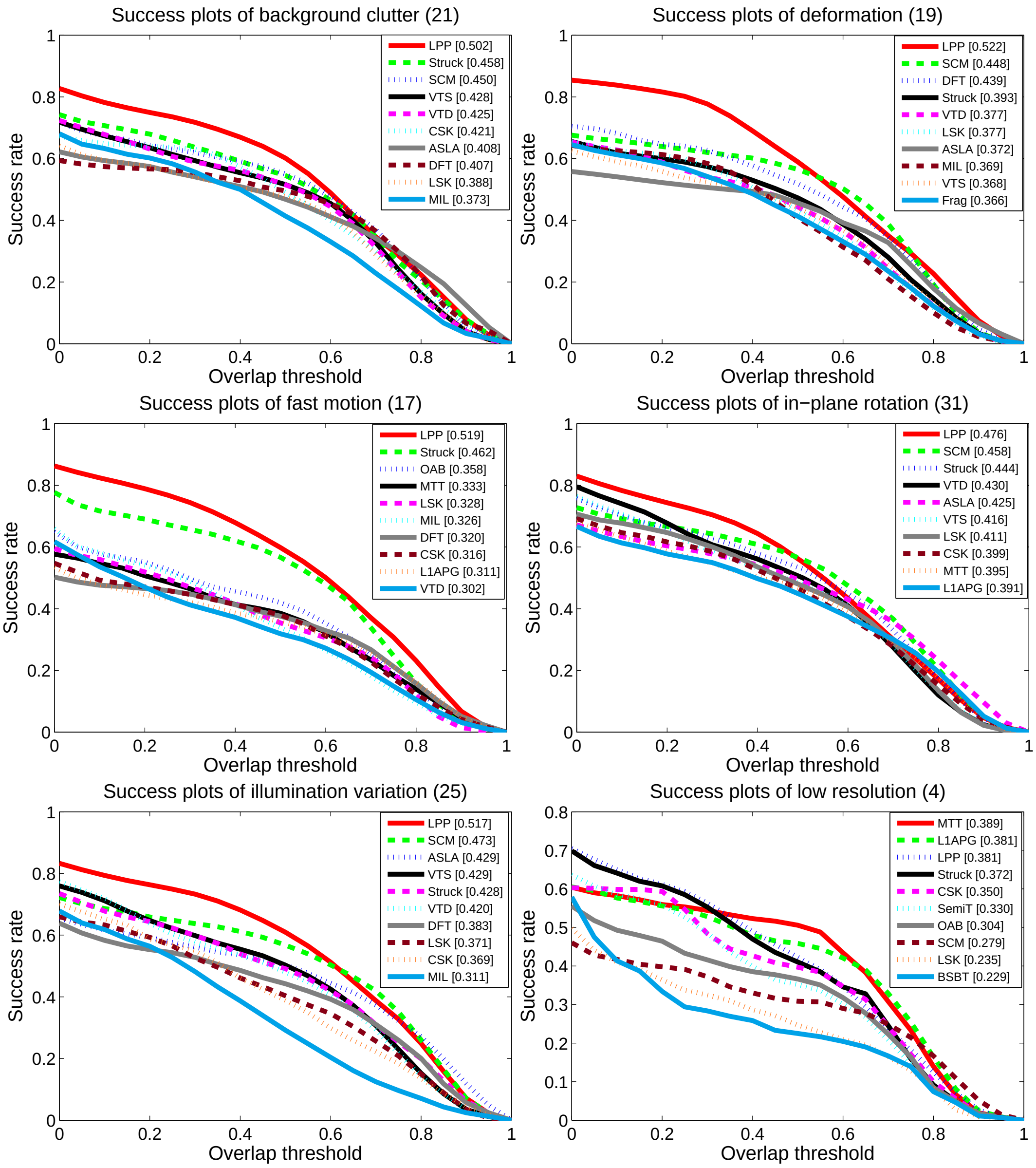


Figure 3.8: The success plots and AUC rankings of 10 tracking methods on challenging sequences.

changes during movement. For example, by updating the selected classifiers, the "concept" can be reflected by the independent particular classifier. If the object leaves out of view then comes back, LPP tracker can still track it using a particular classifier as long as the appearance has been learned in a certain time. On the other hand, using the center location distance criterion, LPP tracker outperforms

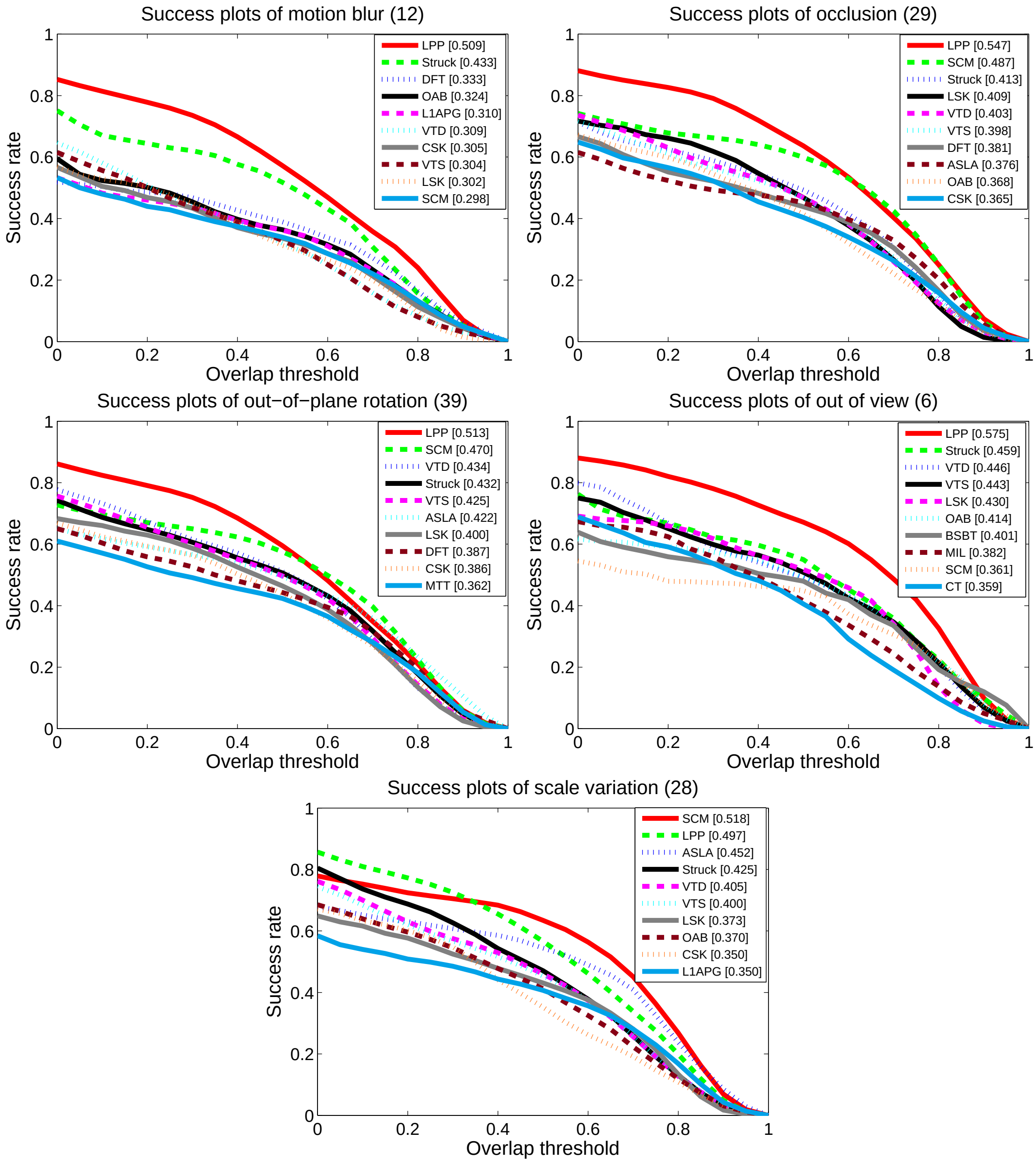


Figure 3.9: The success plots and AUC rankings of 10 tracking methods on challenging sequences.

all other methods on most challenges only except for the low resolution challenge. The advantages using pixel distance criterion are more obvious than the overlap rate criterion. Unlike template matching methods, in some difficult situations including out-of-plane rotation and occlusion, LPP tracker can track the part of object without containing the pixels from environment. However, at such

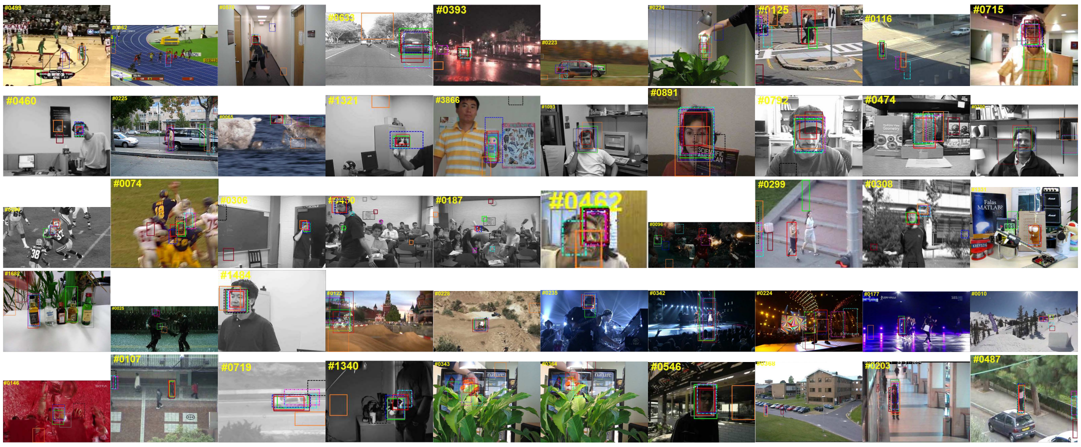
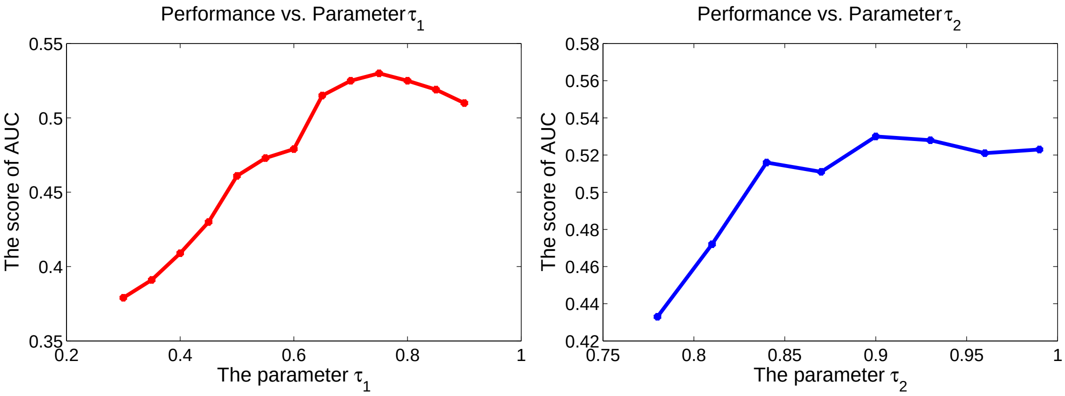


Figure 3.10: Screenshots of top 9 tracking methods (AUC ranking in Fig. 3.7) on challenging sequences, including LPP (red), Struck (green), SCM (blue), ASLA (yellow), VTD (pink), VTS (cyan), CSK (dark red), LSK (orange) and DFT (turquoise).

conditions, the ground truth annotations of video samples in the dataset normally contain the pixels from the environment.

In total, LPP tracker gains nine firsts, one second and one third by the AUC ranking, and it gains ten firsts and one second by the precision ranking. The difference between the two rankings is scale variation. That is because LPP tracker can build a new classifier for one part of the object when there are some large deformations in the remaining part in these challenges, where the object has been tracked by LPP tracker but the score of overlap is relatively low. In addition, there are only four sequences in the dataset containing the challenge of low resolution. The plots in Fig. 3.8 for low resolution are very unsmooth because the samples are not sufficient. At present, no method can solve this challenge very well and the highest score is only 0.389 which is much lower than the scores for other challenges. For LPP tracker, although filters which are robust to noise are used in the image patch representation, when the present classifier focuses on the local part of the object, the descriptor is also influenced by the challenge of low resolution.

There are two parameters of motion constraints τ_1 and τ_2 to guide the learning of LPP tracker. When we investigate one parameter, other parameters will be set to default (same values for all videos). In Fig. 3.11 (a) and (b), the overall performance on all the videos vs. the different settings for the two parameters are given. We can see that the parameter τ_1 achieves the best performance around 0.75 while the parameter τ_2 achieves the best performance around 0.9. If the two parameters are set too small, the model will become more flexible but less stable.



(a) (b)

Figure 3.11: (a) The AUC performance vs. parameter τ_1. (b) The AUC performance vs. parameter τ_2.

More erroneous information will be added into the model and the performance will deteriorate. However, if the two parameters are set too large, the model cannot adapt to the new environment and the performance of the model will decrease as well. Moreover, the scores of AUC are relatively stable around the best values of the two parameters, which means they are not very sensitive.

3.5.2 More analysis on the long-term sequences

To further demonstrate the capabilities of our system, we compare LPP tracker with Struck (the best method in evaluations [21] and [144]) on three more challenging long-term sequences named *motorcross*, *panda* and *sheep*. There are several difficulties, which are normally not considered by other methods before: (1) the target makes a complete rotation; (2) the target moves out of view and gets back with a totally different appearance and location; (3) the video is very long and various challenges appear simultaneously. To some extent, the assumptions of smooth motion and smooth variation necessary for most methods are not valid anymore in such sequences. The three sequences with the above three difficulties will be good examples to test the flexibility and stability of a model. Firstly, Struck fails at frames 30, 1016 and 828 for sequences *motorcross*, *panda* and *sheep*, respectively, when the target starts to move out of view. However, LPP tracker can successfully reject the learning from wrong samples and keep its stability. Secondly, from Fig. 3.12, we can see that LPP tracker can tackle all these problems simultaneously because LPP tracker builds one classifier for each problem. Finally, Fig. 3.13 demonstrates the weights of classifiers on all the frames of sequence *sheep*. When no valid target is detected, LPP tracker will sample

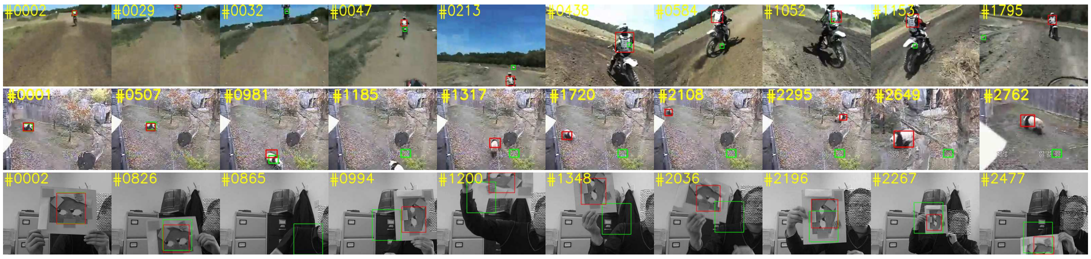


Figure 3.12: Comparsion of tracking results on three more challenging sequences between LPP tracker (Red) and Struck (Green). *motorcross* (top row), *panda* (middle row) and *sheep* (bottom row) have 1800, 3000 and 2532 frames, respectively.

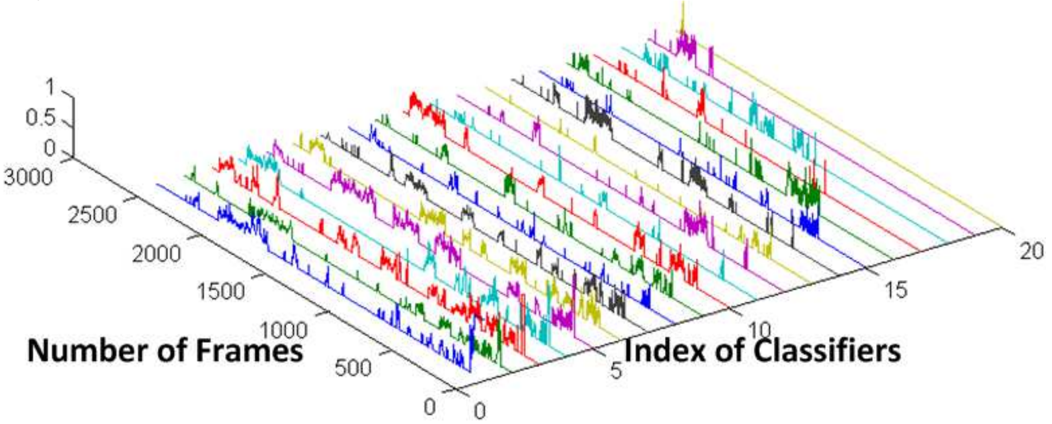


Figure 3.13: The weights for the optimal classifier $\mathbf{f}^t$.

the classifiers according to their historical weights. Once the predefined target appears in view, LPP tracker will select the most effective classifier to track the target. From Fig. 3.13, we can see that the weights are very sparse and just a few members will be run for each frame.

3.6 Summary

In this chapter, we have proposed a Learn++ based tracker for visual tracking. By means of automatically adjusting the members of classifiers, a democracy mechanism is adopted by LPP tracker to solve numerous challenges appearing in the scenarios, simultaneously. Moreover, LPP tracker achieves an optimal balance between flexibility and stability of the classifiers and between the efficiency and performance of the model as well. In future work, it is worth considering using other constraints to guide the sampling of classifiers. Moreover, for abrupt deformation of the target when typically $n < 5$, LPP tracker may refuse to add a new classifier to the ensemble. How to define an adaptive quantity to tackle such a situation is under investigation. In this chapter, the classifiers are generated in a same function space and based on a same type of feature but trained using different filters and samples. In fact, there is evidence that performance of tracking may be increased by combining different successful models. Thus, to further improve the diversity of the tracker, it is meaningful to investigate how to combine various successful models in the future.

Chapter 4

A Winner-take-all Strategy for Improved Object Tracking

As pointed out in Chapter 3, visual tracking is a fundamental task in computer vision and the goal is to estimate the locations or motion states of a predefined target in video. It has many potential applications in surveillance, human-computer interaction, reality augmentation and robotics. In Chapter 3, a set of independent classifiers sampled in a same function space are dynamically maintained and updated according their recent performance and environment. However, the essential reason why online object tracking is very challenging is that it is an under-sample and incomplete-dataset problem. Thus, from the perspective of statistical learning theory, the problem leads to overfitting and low generalisation to tackle the various unpredictable changes. Therefore, to further improve the diversity of the system in this chapter (See Fig. 4.1), a Winner-take-all strategy will be exploited for online object tracking. The differences between the model LPP introduced in Chapter 3 and the model in this chapter are:

- The model in the last chapter selects a set of classifiers from a same functional space but the parameters of them are different. These classifiers which are trained using different datasets and at different time are used to conquer various challenges. However, in this chapter, the tracker members with diverse considerations are from different functional space and thus possess more diversity.

- A set of competitive classifiers are chosen to solve current problem whist, in this chapter, only one winner tracker is selected to improve both the performance and the efficiency of system simultaneously.

The rest of this chapter is organised as follows. The hypotheses and motivation of this work is introduced in Sec. 4.1. In Sec. 4.2, how to build a performance

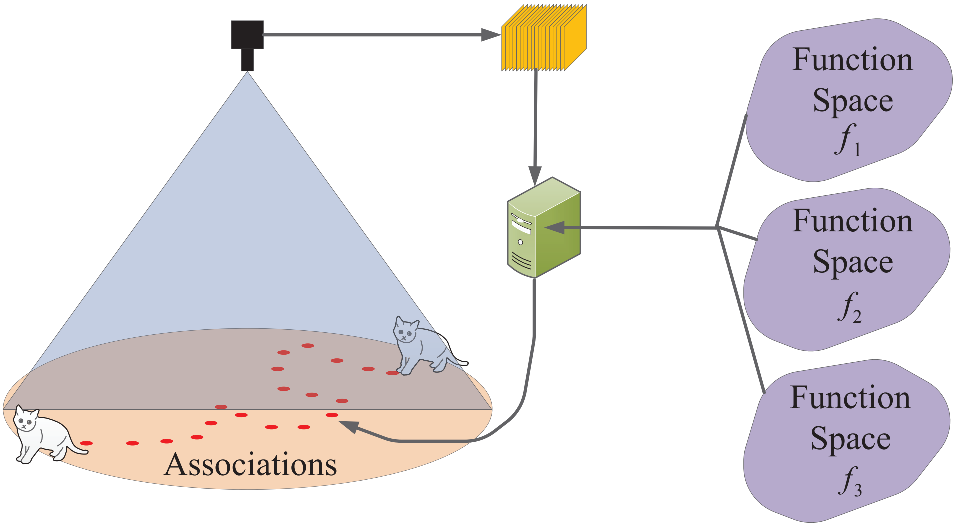


Figure 4.1: Visual data association in a signal camera using a winner tracker selected from different function spaces.

prediction model and how to online select a winner tracker is detailed. Sec. 4.4 presents experimental results. Section 4.5 draws summary.

4.1 Preliminaries

4.1.1 Hypotheses

Beside the basic hypotheses of object tracking introduced in the last chapter, this chapter has the following hypotheses as well:

- The diversity of a system could be improved by incorporating different models. Moreover, the diversity of these models have the difference with each other so that they are complementary.
- It is assumed that, for a particular application, a set of trackers owning diverse properties can be selected from existing methods. Furthermore, a relationship between the performance of these selected trackers and motions (challenges) of a target can be modelled off-line on a large labelled dataset.

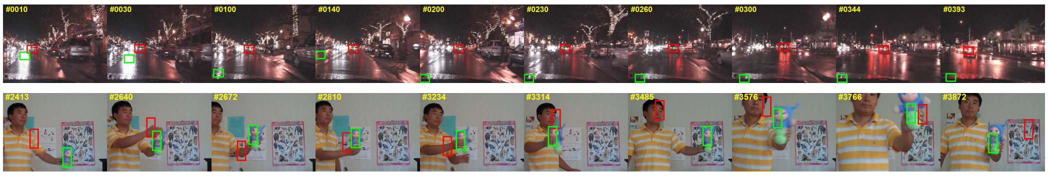


Figure 4.2: Comparison of two trackers Struck [59] (Red) and CT [138] (Green) on sequences carDark and Doll in the TB-50 dataset [21]. Struck outperforms CT on the first sequence (first row) but the performance is obviously lower on the second sequence. The question is: when and how to combine the two trackers without sacrificing the efficiency? This chapter will answer this question from the perspective of a winner-take-all strategy.

4.1.2 Motivation

For decades, numerous algorithms are proposed but different models achieve dissimilar results for different difficulties. For example, part-based models [208] are more robust to partial occlusion, comparative features [138] are more invariant to illumination changes and the tracking-by-detection methods [1] have a stronger ability to tackle the out-of-view problem. Fig. 4.2 shows that two trackers Struck [59] and CT [138] perform very differently on two sequences. And, if several different challenges occur in a long video sequence, most methods will fail to track the target because a single method cannot deal with all the challenges. In general, it is difficult to say which existing tracker can completely outperform all other methods in any environment.

To avoid the overfitting and improve the generalisation, the easiest way is to directly fuse the results from an ensemble [20, 149], which amplify the diversity of the system. However, this strategy naturally increases the computational complexity. Complicated trackers [59] normally perform better on very complex situations than some simple models but the computational complexity of these complicated methods is very high so that they are far from real-time. For example, from the findings in [134], Struck [59] achieves at least 54.9% (47.4 vs. 30.6) higher overall accuracy than CT [138] but its time complexity is more than triple.

Moreover, several existing evaluation reports [3, 134, 137, 144] give a comprehensive investigation of the performance of recently proposed trackers and several datasets for evaluating different trackers are built. In these works, the strengths of various methods and their robustness to different challenges are analysed in detail. The datasets and analysis are very valuable and beneficial to understand the intrinsic principle of object tracking. If the knowledge can be exploited, the performance of a new object tracking system can be improved.

Motivated by the above three observations, in this chapter, a winner-take-all strategy is exploited to select a **winner** tracker which is most suitable and efficient to tackle the current challenge, according to features extracted from the current environment and an efficiency factor. To fast extract features in a tracking environment, a dense trajectories based motion feature is designed to describe characteristics and challenges of the movement of an object and its surrounding. Based on a large public dataset, a prediction model of performance for different trackers on various challenges can be obtained off-line. Then, the learned structural regression model can be directly used to efficiently select the winner tracker online. To increase the flexibility of all members, the tracked results of the winner will be used to update other trackers. The advantages of the proposed WTA tracker are reflected from the following three aspects: 1) By exploiting the knowledge off-line, the performance of trackers can be carefully identified on a large sample set. We can consider the knowledge transferred from a dataset into a new testing sequence. 2) By incorporating the powerful and complementary abilities from multiple trackers, the diversity of the model is improved so that the WTA tracker can tackle various unpredictable difficulties. 3) Since, at any time, only one suitable tracker in terms of both accuracy and speed is executed, the WTA tracker will be much faster than the slowest one. The best cases are that the fastest tracker can be chosen for simple situations most of the time and a complex and accurate tracker will be occasionally used only when there is a difficult challenge.

4.2 Learning to Predict the Performance

Our proposed WTA tracker contains two parts: off-line tracker evaluation and online tracking. On the one hand, based on the public dataset, the off-line evaluation is to learn a model to predict the performance of the trackers in the selected set. It is worth to point out that this step is not to pre-train the tracker but rather to build an evaluation criterion. On the other hand, online tracking is used to choose a winner tracker (most suitable) according the motion feature and the efficiency factor in the current situation and then to update all trackers.

Consider the following general setting. Suppose that there are K trackers available in the selected set $f_k \in \mathcal{T}$, indexed by k. To balance the performance and efficiency of the system, a quantity of speed statistics $\mathcal{S} = \{s_k : f_k \in \mathcal{T}\}$ for trackers is also considered. We assume that I^t is the current frame in which the state of object needs to be estimated. Therefore, we have

$$\begin{aligned} \bar{\alpha} &= max_{\alpha,k} p(\alpha, f_k | I^t_{t-\delta}, \mathcal{S}) \\ &= max_{\alpha,k} p(\alpha | f_k, I^t) p(f_k | I^{t-1}_{t-\delta}, \mathcal{S}), \end{aligned} \tag{4.1}$$

where $I^t_{t-\delta} = \{I^t, \cdots, I^{t-\delta}\}$ is the previous frames and δ is a fixed parameter.

It is worth to point out that, if the operation is replaced by a summation, thus the framework becomes an ensemble of trackers. In this situation, the system will be very computationally expensive. The key of our WTA tracker is that the best tracker can be selected according to current motion features and it is not necessary that all trackers should be executed.

4.2.1 Evaluation criterion

Given a frame I_i in a video V for which the ground truth state $\hat{\alpha}_i$ of a motion is labelled, we can obtain state α_i^k for tracker f^k. Then, a rate can be defined to describe the performance of different trackers by using the intersection.

$$y_i^k = \frac{Area(\hat{\alpha}_i \cap \alpha_i^k)}{Area(\hat{\alpha}_i \cup \alpha_i^k)} \tag{4.2}$$

For different trackers f_k, the overlapping rate will be dissimilar because the abilities of trackers are very different. It is obvious that y_i^k is between 0 and 1 and when the value is larger, the tracker is more powerful. In this chapter, a rectangle which encloses the object will be considered as the state and the overlapping rate is defined as the overlap between the rectangles of the ground truth and the one estimated by tracker f_k. Our aim in the evaluation step is to build a relationship between the overlapping rate y_i^k and the motion in several previous frames $I_{t-\delta}^{t-1}$.

4.2.2 Motion features

Motion is a most informative cue for tracking and most difficulties are induced by the motion of the target, including deformation, in-plane rotation, out-of-plane rotation, fast motion, motion blur, scale variation and occlusion. Moreover, due to the diverse shape of objects and the cluttered environment, the motion will become more complex. How to design a robust representation of motion is still a very challenge task. Recently, dense trajectories [209] has been shown to be effective for action recognition in a cluttered environment. The trajectories are extracted by tracking densely sampled points using KL tracker [17]. Following [209], action localisation proposals from dense trajectories is proposed in [210] by using an efficient proposal generation algorithm. KL tracker is normally used to assist the algorithms using the idea of tracking-by-detection, such as [1, 152]. KL tracker is very efficient when the sampled points are sparse.

In this chapter, we adopt the dense trajectories to describe the motion for choosing the most suitable tracker. If the current frame is I^t, then several previous frames $I_{t-\delta}^{t-1}$ are available and the corresponding states $\alpha_{t-\delta}^{t-1} = \{\alpha^{t-\delta}, \cdots, \alpha^{t-1}\}$ of the target have been estimated. We divide the rectangle α^{t-1} in the last frame I^{t-1} into 9×9 grids and thus 100 points need to be back tracked. A common

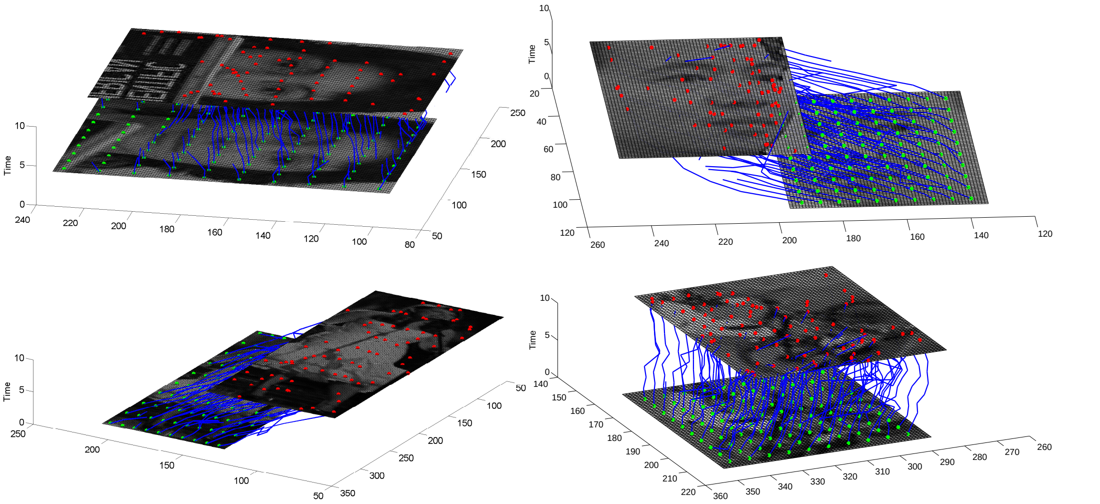


Figure 4.3: Four cases of challenges captured by the motion flow. Top left (FaceOcc1): some points are vanished due to the occlusion; Top right (Mhyang): out-of-plane rotation; Bottom left (Singer1): scale change and camera motion; Bottom right (MountainBike): in-plane rotation and camera motion.

phenomenon is that, due to the motion of the target, some points will disappear and others forming new appearances will appear in the scene (e.g., recover from partial occlusion). However, this has no negative influence on the representation of motion, as long as the traces of selected points in the last frame are correctly estimated. If one point cannot been found in some frames, the location will be set to the same as the location which has been already traced. Also, we can estimate the locations in the current frame by using KL tracker.

Motion flow calculated in a region centered around the object is used to capture the variations and encode local motion patterns induced by the environment and object. Camera motion can be considered one of the challenges and can be characterized by the motion feature similarly as other challenges. Fig. 4.3 shows how to calculate the motion flow and two cases (third and fourth) contain the challenge of camera motion. Both in training and testing stages, the motion flow is computed inside the rectangles based on the past δ labelled or tracked states and used to predict current performance. More clearly, the feature x contains two parts: $x = (x_m, x_b)$. We have $x_m = (\mathbf{S}_1, \cdots, \mathbf{S}_{100})$ where $\mathbf{S}_j = (\triangle P_j^t, \cdots, \triangle P_j^{t-\delta+1})$ and $\triangle P_j^t = (\triangle \mathbf{x}_j^t, \triangle \mathbf{y}_j^t)$. $\triangle \mathbf{x}_j^t$ is the horizontal-coordinate displacement of point j in the image space between two frames. Also, we have $x_b = (\triangle B^t, \cdots, \triangle B^{t-\delta+1})$ where $\triangle B^t$ is the average brightness difference of the points between two frames.

4.2.3 Structural regression modal training

Through the reviews [134, 144], we can see that different trackers give very dissimilar performances for different challenges. In this section, we will build a relationship between the performance of various trackers and the challenges.

Dataset collection: The first step is to collect samples and give the annotations to the samples for each tracker. We use the ALOV300 dataset collected in [144], which contains 315 sequences and the ground truth of each sequence is annotated. Suppose that there are K trackers available in the selected set $f_k \in \mathcal{T}$. Then, each tracker will be tested on each sequence individually and we can obtain the estimated state α_i^k for every frame. To achieve this, each tracker will be initialised in the first frame, and used to track the target in following frames and also updated ontime normally.

Because the sequences are annotated, the set $\alpha_{t-\delta}^{t-1}$ consists of the states of ground truth in the previous frames $I_{t-\delta}^{t-1}$. Therefore, according to dense trajectories, the motion feature x_i of the target for the current frame I_i can be obtained. For frame I_i, we have the state of ground truth $\hat{\alpha}_i$ and the states α_i^k estimated by every tracker individually. Thus, the overlapping rate y_i^k for tracker f_k can also be calculated. Therefore, for each motion feature x_i, an overlapping rate vector $y_i = (y_i^1, \cdots, y_i^K)$ can be obtained. If all the items are close to zero which means the object and environment keep still (The current situation can be easily solved by all the trackers), the pair of samples (x_i, y_i) will be ignored.

Once the dataset $\mathcal{D} = (\mathcal{X}, \mathcal{Y})$ is constructed, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$, a relationship can be built to predict the performance according the motion feature. Therefore, our aim is to learn a compatibility function $L : \mathcal{X} \times \mathcal{Y} \rightarrow R$ over the pairs of motion feature and performance of each tracker. By maximising L over the response variable for a specific given input x_i, we can derive a prediction function as:

$$F(x; w) = max_{y \in \mathcal{Y}} L(x, y; w), \tag{4.3}$$

where w denotes a parameter vector and will be estimated in the optimisation step. In this chapter, we suppose the L is a linear combination of joint features [211] $\Psi(x, y)$:

$$L(x, y; w) = w^T \Psi(x, y), \tag{4.4}$$

where $\Psi(x, y)$ is a feature vector induced by a joint kernel $K(x_i, y_i, x_j, y_j) = (\Psi(x_i, y_i))^T \Psi(x_j, y_j)$. Hence, given a tracker set $\mathcal{T}$ and sample pair (x_i, y_i), we have: $max_{y \in \mathcal{Y}} w^T \Psi(x_i, y) \leq w^T \Psi(x_i, y_i)$. If we define $\Delta\Psi_i(y) = \Psi(x_i, y_i) - \Psi(x_i, y)$, for any i and y, the inequality can be simplified as:

$$w^T \Delta\Psi_i(y) \geq 0. \tag{4.5}$$

$w^T \Delta\Psi_i(y)$ is the linear margin in the feature space.

To find the optimal parameter, a loss function $\Delta(y, y_i) : \mathcal{Y} \times \mathcal{Y} \rightarrow R$ should be defined to quantity the loss associated with a prediction y, if the true performance is y_i. We assume that $\Delta(y, y_i) = 0$ if $y = y_i$; otherwise $\Delta(y, y_i) > 0$. The loss can be considered as the lower boundary of line margin $w^T \Delta\Psi_i(y)$, meaning that the optimal w should maximise the line margin $w^T \Delta\Psi_i(y)$ as much as possible. We are usually not able to find a model that satisfies constraints exactly, hence some slack variables are added to allow examples to deviate from the boundary.

$$w^T \Delta\Psi_i(y) \geq \Delta(y, y_i) - \xi_i. \tag{4.6}$$

It is obvious that the solution is more than one as long as the norm of w is large enough so that all constraints are satisfied. To obtain a unique solution, the norm is limited to be smaller than 1: $||w||_2 \leq 1$. This derives the general maximum-margin framework and the objective function can be defined as:

$$\begin{array}{c} min_w \frac{1}{2}||w||^2 + C\sum_i \xi_i \\ s.t. \forall i, y \\ w^T \Delta\Psi_i(y) \geq \Delta(y, y_i) - \xi_i. \end{array} \tag{4.7}$$

The problem 4.7 can be solved using the SVM-struct library [212]. In this chapter, a tensor product joint kernels [213] is used to represent the joint feature maps and there is $K(x_i, y_i, x_j, y_j) = K_x(x_i, x_j)K_y(y_i, y_j)$. Both motion feature and performance measurement will use a liner kernel to construct the feature maps: $K_x(x_i, x_j) = x_i^T x_j$ and $K_y(y_i, y_j) = y_i^T y_j$. By solving the problem 4.7, we can obtain an optimal w as follows:

$$w = \sum_i \sum_{y \in S_i} \beta_{iy} \Delta\Psi_i(y), \tag{4.8}$$

where S_i is a working violate set for sample x_i and β_{iy} is a parameter. Both will be estimated by solving the dual problem of 4.7. The algorithm of training is given in Alg. 2.

Algorithm 2 WTA training

```
**Input** Trackers $f_k \in \mathcal{T}$ and ALOV300 Dataset.
(a) Run each tracker $f_k$ for all sequences $V$ in the dataset.
(b) Calculate overlapping rate $y^k$ according to Eq. 4.2.
(c) Compute motion features $x$ using the dense trajectories $\mathbf{S}$.
(d) Optimise the objective function 4.7.
(e) Obtain parameters by 4.8.
**Output** The prediction function $F(x; w)$.
```

During training, 5000 annotated samples $\mathcal{D} = \{(x_i, y_i)|i = 1, \cdots, 5000\}$ divided into two parts (4000 used for training and the remaining for validation) are generated and the cross-validation method is used to obtain robust parameters and avoid over-fitting. δ is set to 10 in our experiments, considering the robustness of motion flow.

4.3 Online Winner-take-all Tracking

Once the prediction model of performance is built, we can use it to select the most suitable tracker (winner) according to the current motion feature and the efficiency factor. To keep the flexibility, the result of the winner will be used to update all members. The algorithm of online tracking is given in Alg. 3.

Algorithm 3 WTA tracking

```
**Input** Tracker set $\mathcal{T}$, prediction function $F(x;w)$, distribution $p(f_k|\mathcal{S})$ and state of a target in the first frame.
**Initialisation** Initialise trackers $f_k$ according the ground truth in the first frame.
**Repeat** $t = 1, \cdots$
(0) Capture a new frame. **If** no frame: **Exit**.
(1) Calculate the motion feature $x$.
(2) Obtain the distribution $p(f_k|I_{t-\delta}^{t-1}, \mathcal{S})$ according to Eq. 4.9.
(3) Choose the winner tracker $f_{k^*}$ according to Eq. 4.10.
(4) Run the winner on the current frame.
(5) Obtain the state of target using $p(\alpha|f_{k^*})$.
(6) Collect samples in the current frame.
(7) Update all trackers. **Go To** (0).
**Return** The state of target in all frames.
```

4.3.1 Winner selection

According to the probability framework of WTA in Eq. 4.1, we can see that the simplest way to boost the performance is to firstly execute all the trackers and then choose the state of best tracker as the state of system. However, this strategy is obviously computationally expensive. In general, for most object tracking algorithms, the most time-consuming process is to search the target in the image space, which includes patch representation, classification and fusion. The number of sampled patches, which are generated by dense sampling or local sampling, primarily determines the overall efficiency of the proposed tracker. Therefore, to avoid the time-consuming checking for all trackers, we turn to adopt another way that a most suitable tracker is first selected by calculating $p(f_k|I_{t-\delta}^{t-1}, \mathcal{S})$, then the optimal state will be estimated by this tracker according to $p(\alpha|f_k, I^t)$.

For online tracking, the first step is to calculate the probability $p(f_k|I_{t-\delta}^{t-1}, \mathcal{S})$. We assume that, through the analysis on a big dataset, the prediction function

$F(x, w)$ of performance has be built and the efficiency statistics $\mathcal{S}$ of trackers are obtained. The larger value of s_k means that the efficiency of the corresponding tracker is higher. We can consider it as the prior information to select the trackers according to the distribution: $p(f_k|\mathcal{S}) = s_k / \sum_k s_k$. The quantity s_k keeps relatively stable, because the speed only depends on the searching framework of trackers themselves. For most trackers, the process of online tracking is same for different frames or various difficulties. Thus, the two variables $I_{t-\delta}^{t-1}$ and $\mathcal{S}$ are independent with each other and we have:

$$p(f_k|I_{t-\delta}^{t-1}, \mathcal{S}) = p(f_k|I_{t-\delta}^{t-1})p(f_k|\mathcal{S}). \tag{4.9}$$

In the cases where the predicted performance of most trackers is similar or there are no changes for the target and and environment, the fastest tracker will be the desired one.

To calculate $p(f_k|I_{t-\delta}^{t-1})$, we further assume that the current frame is I^t and the previous frames $I_{t-\delta}^{t-1}$ and their corresponding states $\alpha_{t-\delta}^{t-1}$ are available. Hence, a motion feature x can be extracted from $I_{t-\delta}^{t-1}$. Then, the prediction model of performance $y = F(x, w)$ is used to evaluate the trackers. Therefore, we can calculate the probability $p(f_k|I_{t-\delta}^{t-1}) = y_k / \sum_k y_k$. Finally, a winner tracker will be selected according:

$$k^* = max_k p(f_k|I_{t-\delta}^{t-1}, \mathcal{S}). \tag{4.10}$$

Once the most suitable tracker is selected, the second step of online tracking is to detect the predefined target as usual. It is worth to notice that the most suitable tracker is not the one which can achieve the best performance but rather the most efficient tracker in a set which can solve the current problem. Finally, the state result of frame I^t depends on the selected tracker and different trackers adopt various methods of representation, classification and fusion.

4.3.2 Online updating

According to recent surveys [134, 137, 144], a typical tracking system includes four important components: state sampling, patch representation, matching or recognition and updating. Different algorithms explore diverse schemes to implement every part but all the methods should be updated according to the result from the current frame. Updating is the most significant step so that the system can be adaptive to the changes induced by targets and environments. Fortunately, comparing to state searching, updating is much faster because normally only several patches need to be calculated. To keep the effectiveness of all trackers, we should maintain them using the current result so that all the trackers can adapt to the changes. Hence, the state in the current frame is the connecting

Method	WTA	Struck	CT	TLD	ASLA	KMS	CSK
mFPS	134.2	20.2	64.4	28.1	8.5	316	362
AUC	56.9	47.4	30.6	43.7	43.4	32.6	39.8

Table 4.1: The properties of selected trackers. mFPS means the average FPS of the tracker on all frames.

point and plays a role as a medium. All member trackers can interact mutually at this point and one tracker can learn from others and become more powerful.

Suppose that $\alpha^t = max_\alpha p(\alpha | f_{k^*}, I^t)$ is the optimal state in frame I^t using the selected tracker f_{k^*}. Same to the update procedure in most methods, 50 positive samples will be generated by the affine warping of the selected patch whose overlap with α^t exceeds 0.5. For updating the discriminative methods, 50 negative samples will be generated randomly from the patches whose overlap with α^t is less than 0.1.

4.4 Experiments

4.4.1 Two public datasets

In our experiments, two recent public datasets ALOV300 [144] and TB-50 [21] are used.

ALOV300: This dataset[1] is used to learn the structural regression model off-line. ALOV300 dataset consists of 315 video sequences and the total number of frames is $89,364$. The average length of these sequences is 9.2 seconds with a maximum of 35 seconds. The collection is categorised into thirteen aspects of difficulty. The sequences are annotated by a rectangular bounding box along the main axes of a flexible size every fifth frame. In rare cases, when motion is rapid, the annotation is more frequent. The ground truth has been acquired for the intermediate frames by linear interpolation.

TB-50: This datseset[2] is used to test our WTA tracker online. TB-50 dataset contains 50 sequences and the total number of frames is $29,000$. All sequences are manually tagged with 11 attributes, which represent the challenging aspects in visual tracking. Unlike ALOV300, most sequences are longer and are labelled with at least three difficulties. Every frame in TB-50 is manually annotated by a rectangular bounding box.

[1] http://www.alov300.org/

[2] http://www.visual-tracking.net

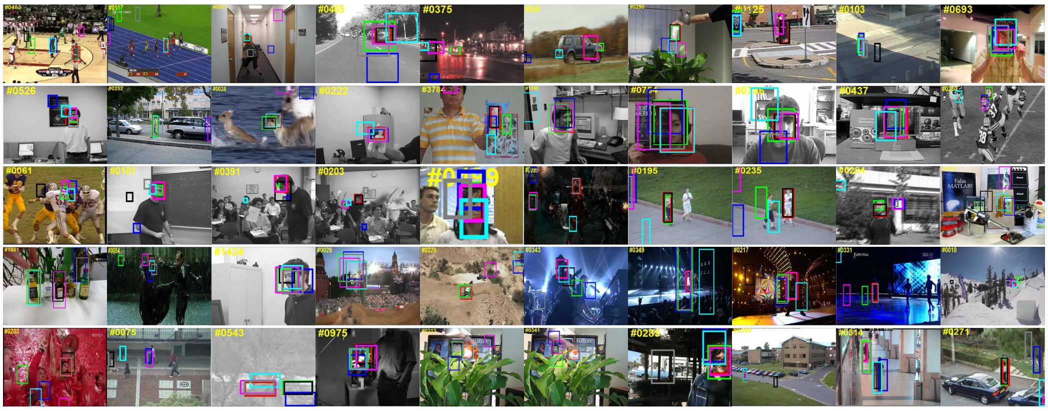


Figure 4.4: Screenshots of the 7 tracking methods including the proposed WTA (Red), Struck (Green), CT (Blue), TLD (Black), ASLA (Magenta), KMS (Cyan) and CSK (Gray) on challenging sequences. In total, 29486 frames are tested. (Zoom in for better viewing.)

4.4.2 Selected trackers

From the report in [21], six methods including **Struck** [59], **CT** [138], **TLD** [1], **ASLA** [135], **KMS** [214] and **CSK** [204] are chosen as the basic trackers in WTA. The mean number of frames per second (mFPS and the overall performance AUC on dataset TB-50 are shown in Table 4.1. The experiments were run on a Dell M4600 (Intel Core 2.8GHz and 8G RAM). The properties of trackers are similar to that in [21]. Struck is a structured supervised classifier based method and is the best tracker according to the overall performance among the six selected trackers. In CT, a set of projections randomly generated is used to embed the features into a low dimensional space. It is fast because the projections are fixed and updating is only applied to the naive Bayesian classifier. ASLA is a representative sparse coding based method. It is relatively slow but generally achieves better results for partial occlusions. KMS uses a kernel-based similarity to define the distance between the target model and target candidates and is very fast. In the top ten methods of the report in [21], CSK has the highest speed where the proposed circulant structure plays a key role. We can see that each selected tracker has strengths and weaknesses. In fact, the selection of trackers is not limited to the six methods and any tracker can be chosen, but in the general rule is any selected tracker contains some characteristics that complement other selected trackers.

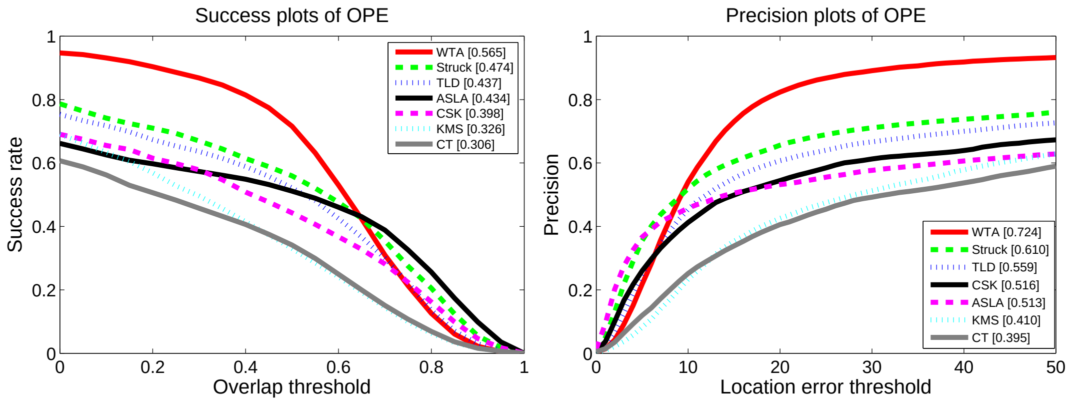


Figure 4.5: Overlap success and distance precision plots over the benchmark using OPE. AUC rankings are given in the legend.

4.4.3 Overall performance on benchmark

To quantitatively illustrate the performance of the WTA tracker, using one-pass evaluation (OPE) [21], two metrics including the centre location error and the overlapping rate are adopted. The centre location error is the Euclidean distance between the centre of the tracking result and the ground truth while the overlapping rate is defined by the ratio between intersection and union of the two rectangles. For fair comparison, other selected algorithms are run individually. The overall performance is shown in Fig. 4.5 and one screenshot for each sequence is given in Fig. 4.4. All the methods are ranked according to the area under curve (AUC) in Fig. 4.5. We can see that the proposed WTA tracker achieves much better results than the compared methods, no matter which metric is used. According to the overall performance reported in [21], Struck is the best method. However, the performance of our WTA tracker is at least 9.1% higher using the overlap metric and 11.4% higher using the centre error metric. Moreover, the precision comparisons on 11 challenges are illustrated in Fig. 4.6 and 4.7. From the AUC scores, we can conclude that WTA can boost the performance over all the challenges. Struck cannot outperform others on challenges of illumination variation and scale variation, but WTA can perform much better than others by incorporating the abilities of different trackers. Furthermore, we can see that the performance of WTA is at least 6.9% higher than others on the challenge of fast motion and 10% higher on most other challenges. Finally, the mean number of frames per second (mFPS) is also computed to show the efficiency of trackers. From Table 4.1, the mFPS score of WTA ranks the third. We can see that WTA is much faster than the trackers including Struck, CT, TLD and ASLA by choosing other two faster trackers KMS and CSK, when there is no challenge.

WTA is compared with the state-of-the-art methods, which is shown in Table

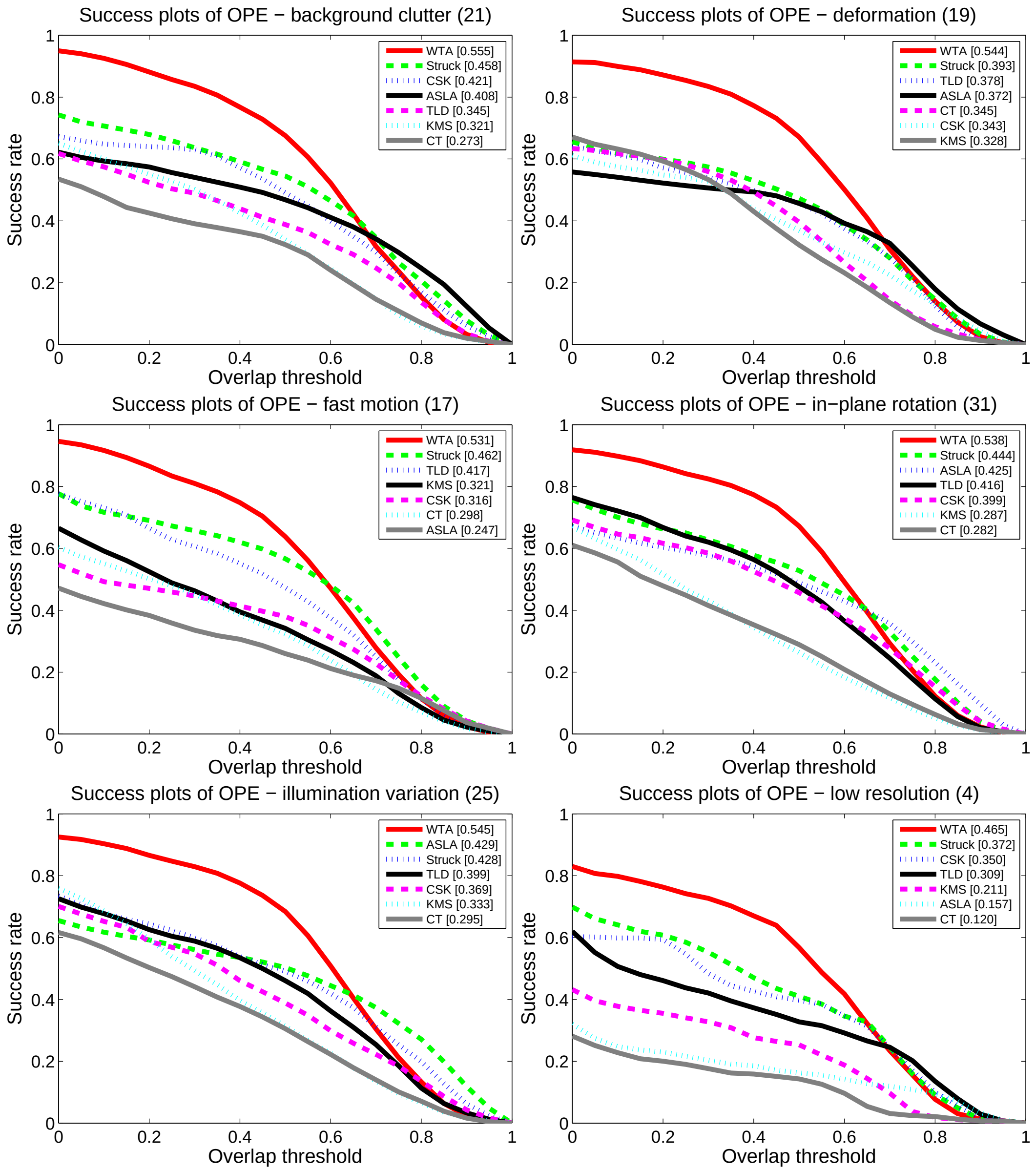


Figure 4.6: Overlap success and distance precision plots for different challenges.

4.2 (including ensemble based methods: MEEM, EBT and Muster). We can see that the proposed WTA is competitive with the state-of-the-art methods. Moreover, the efficiency of KCF is 28% higher but the performance is 25% lower than WTA. And, the performance of WTA is 10% lower than Muster but our WTA is almost 4 times faster than it.

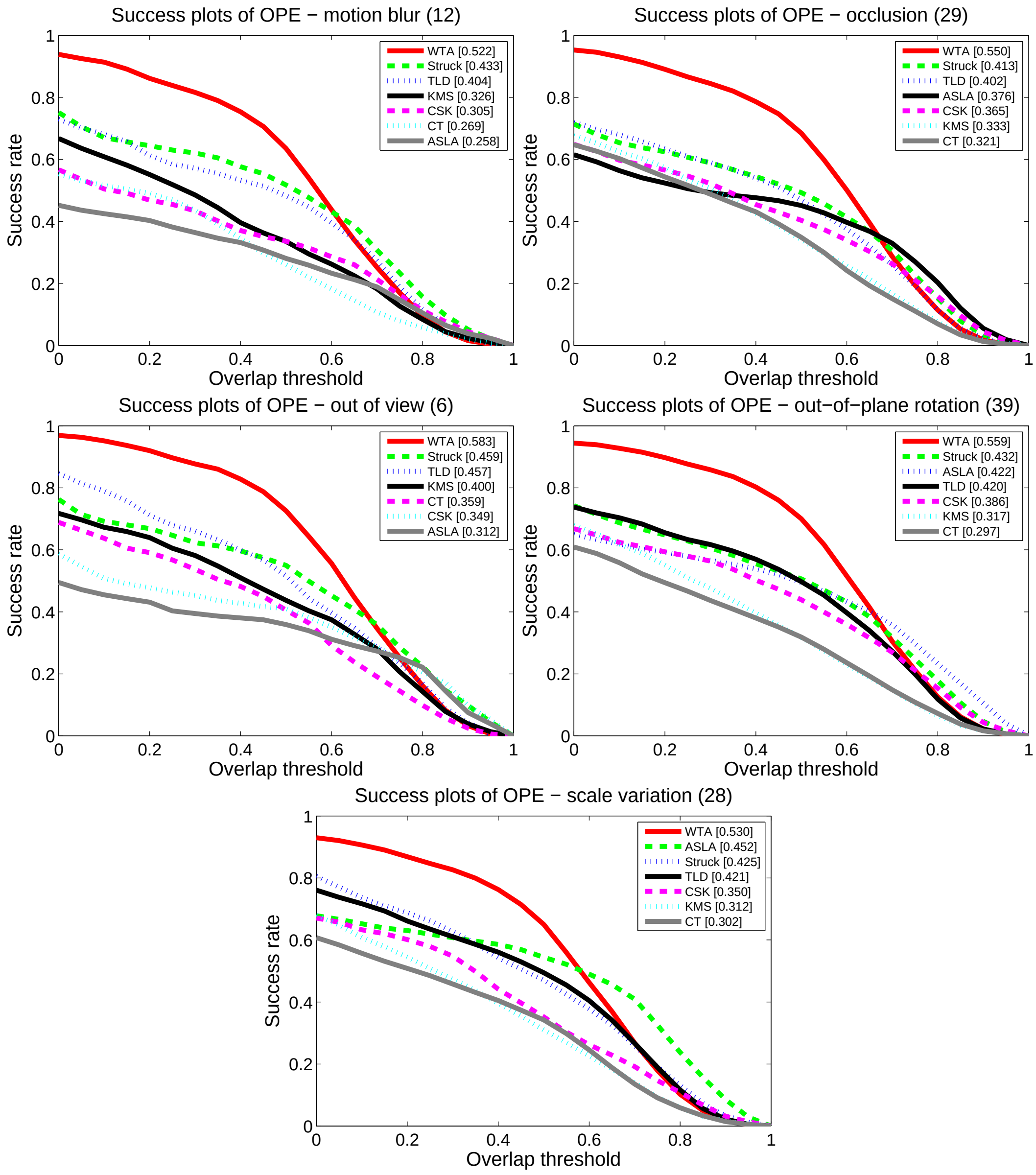


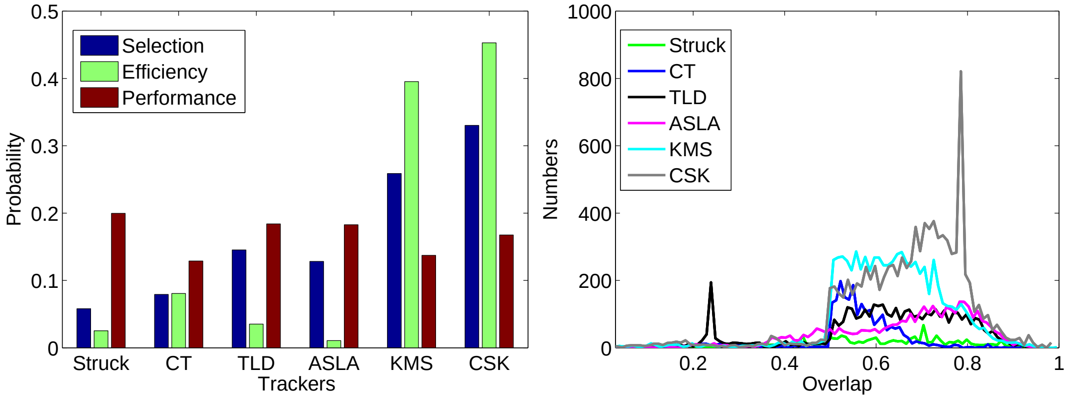
Figure 4.7: More about overlap success and distance precision plots for different challenges.

4.4.4 Component analysis

From the framework of winner-taker-all, we can see that only one tracker will be selected to tackle the problem for each frame. Thus, the selection probabilities shown in Fig. 4.8 (a) are calculated to illustrate how many frames are selected for each tracker. Firstly, the probability is slightly proportional to the efficiency

Method	WTA	KCF	TGPR	MEEM	Muster	RPT	EBT
mFPS	134.2	172	3.5	10	29.5	4	1
AUC	56.9	45.5	53.9	57.2	64.1	57.6	53.8

Table 4.2: Comparison with the state-of-the-art methods including KCF [215], TGPR [216], MEEM [151], Muster [154], RPT [217] and EBT [218].



(a) (b)

Figure 4.8: (a) Selection probabilities comparison to the efficiency and overall performance of the 6 components. (b) Relationship between the overlap score and the number of times the corresponding tracker is chosen.

distribution $p(f_k|\mathcal{S})$ but has less connection to the overall performance of trackers. It does not mean that the performance of each tracker is completely independent of the selection. In fact, trackers are selected mainly according to their instant performance on special challenges. Secondly, from Fig. 4.8 (b), we can see that most trackers are chosen when they can achieve good results (overlapping rate is larger than 0.5). It is interesting to report that TLD is chosen many times at the rate 0.2, because, in TLD, the samples which are divided into two groups at the threshold 0.2 are used to train and update the trackers.

Finally, to further evaluate the contribution of each component, we build our WTA tracker based on 5 trackers. The mean number of frames per second and overall performance are investigated when one tracker is removed from the set. From Table 4.3, comparing with other methods, we can see that the overall performance is influenced much when Struck or ASLA is removed but the efficiency is a little improved. On the contrary, if the fast trackers KMS and CSK are removed, the efficiency is hugely influenced. Therefore, we can conclude that complex trackers normally contribute much to performance and simple, fast models contribute much to efficiency of the proposed winner-take-all framework.

Method	-Struck	-CT	-TLD	-ASLA	-KMS	-CSK
mFPS	263.3	140.6	143.8	150.9	109.7	89.2
AUC	52.8	57.2	54.7	53.1	56.1	55.1

Table 4.3: Component analysis. "-Struck" means that Struck will be removed from the framework of WTA. Hence the corresponding quantities mFPS and AUC are calculated without Struck.

Sequence	S1	S2	S3	S4	S5	S6	S7
MEEM	0.53	0.61	0.14	0.04	0.25	0.32	0.33
MUSTer	0.85	0.25	0.26	0.07	0.99	0.28	0.04
Struck	0.68	0.43	0.15	0.04	0.55	0.63	0.38
TLD	0.98	0.72	0.81	0.33	0.40	1.00	0.40
WTA	0.97	0.83	0.95	0.65	0.75	0.89	0.65

Table 4.4: Precision comparison of tracking results on 7 long-term challenging sequences. S1 and S2 are collected from [1], S3 are from [2] and the last 4 sequences are from [3]. Precision is defined as the centre distance between the truth rectangle and the detected rectangle is less than 20 pixels.

4.4.5 Analysis of long-term tracking

Generally, long-term tracking could be considered as one of the most challenging because of possible different problems presenting in a same sequence. Single tracker which models a certain aspect of challenging normally fail to meet the requirement to tackle these problems simultaneously. To further evaluate the performance of our proposed WTA, 7 long-term sequences, which the average and minimum numbers of frames are 2791 and 2133 respectively, are selected from public datasets and the comparison results between WTA and some methods including TLD, Strack, MUSTer and MEEM, which are thought as the relative powerful for long-term tracking, are reported in Table 4.4 and 4.5. From the two table, we can see that the proposed WTA achieves consistently advantageous results over the state-of-the-art methods. It is worth to point out that WTA is obviously more robust than others on these 7 sequences. For example, MUSTer could completely track the object on the 5th sequence but almost fail on the 4th and 9th sequences. In contrast, WTA could solve the different sets of problems in the different sequences.

Sequence	S1	S2	S3	S4	S5	S6	S7
MEEM	3	1295	23	21	130	20	14
MUSTer	3	173	10	143	82	34	5
Struck	48	1285	8	57	82	25	13
TLD	872	228	8	65	82	2500	7
WTA	1486	1297	2011	356	325	1663	186

Table 4.5: Fail comparison of tracking results on 7 long-term challenging sequences. The number denotes which number of frames the centre distance of one tracker is larger than 20 pixels.

4.5 Summary

In this chapter, to improve the performance and efficiency, a winner-take-all framework has been proposed for object tracking by incorporating the strengths of trackers for different challenges. It proves that different trackers have different characteristics and the combination of them is valuable. In the future, there are two points which need to be further investigated. On the one hand, in this chapter, to guarantee the performance, several advantageous trackers are used. In fact, it is meaningful to consider whether the performance can be hugely improved, when merely simple fast trackers are combined. On the other hand, in this chapter, the trackers interact with each other simply through the result of the current frame. Whether trackers can learn from each other more deeply and grow in a similar way of the crossover and mutation steps in genetic programming is still under investigation.

Chapter 5

Learning Cross-view Binary Identities for Fast Person Re-ID

This chapter will turn to person re-identification (Re-ID) in a cross-camera setting, see Fig. 5.1. Cross-camera person re-identification (Re-ID) is a fundamental solution for automated video surveillance [22]. It addresses the problem of associating people, at different locations and times, observed by the non-overlapping Closed-Circuit TeleVision (CCTV) system. It has various potential applications, such as long-term multi-person tracking, person re-acquisition and forensic search [22]. Thus, the models introduced in previous chapters could be considered as the preprocessing of person Re-ID. Normally, in solving the task of person Re-ID, the single-view person detection and tracking are assumed to be successfully addressed. By combining the procedures of object tracking and re-identification induced in this chapter, a fully trace of a person in a large area could be discovered. Nevertheless, we still need to point out that although the methods proposed in last two chapters could be considered as the pre-stage of person Re-ID when the predefined target is set to a person, it is not limited to a person only and other objects could also be tracked. Furthermore, both single view tracking and cross-camera re-identification are designed to realise the tasks of visual data association. However, the sources or the settings of the two tasks are striking different. For instance, only one positive sample is given in the task of tracking online whist a large number of samples could be collected offline for the task of re-identification. Hence, the potential necessary assumptions behind of the two tasks are different.

Due to the various difficulties including illumination changes, viewpoint and pose variations, inter-object occlusions and low resolution images, person re-identification is still a very challenging task and far from being tackled. Most of the state-of-the-art approaches can be categorised into two groups: learning discriminative features which are invariant to view changes [23, 162, 163, 166]

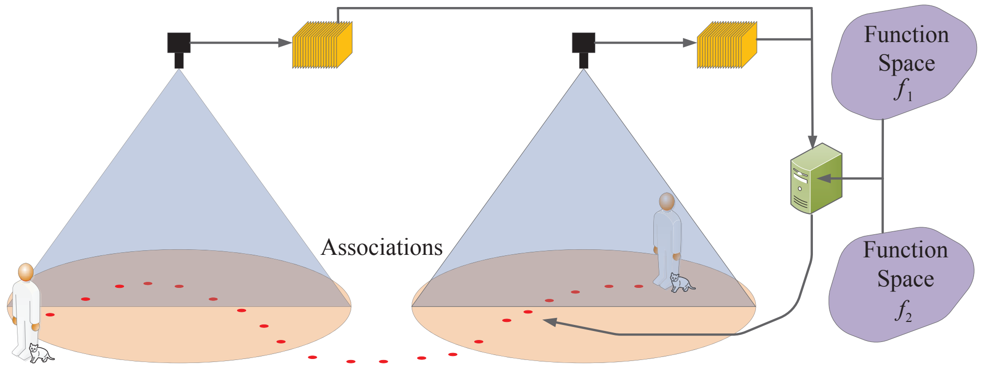


Figure 5.1: Visual data association in a cross-camera setting for person re-identification. The traces of a person in a single view could be addressed using the models introduced in the Chapter 3 and 4.

and learning the metric functions which are used to rank the pairs of observations from different views [169, 170, 172, 219]. However, in spite of their good performance on public datasets, existing methods generally neglect considering the efficiency of the algorithm in the matching stage. In fact, the searching speed of a re-identification algorithm plays a significant role in real-world applications. Therefore, in this chapter, a novel approach, learning Cross-view Binary Identities (CBI), is proposed to reduce the computational burden for person re-identification.

The rest of this chapter is organised as follows. The hypotheses and motivation of this work are introduced in Sec. 5.1. In Sec. 5.2, how to build a model to learn cross-view identities for person re-identification is detailed. Then, Sec. 5.3 presents how to solve the complicated objective function and convergence proof. Sec. 5.4 presents experimental results. Section 5.6 draws summary.

5.1 Preliminaries

5.1.1 Hypotheses

Generally, the hypotheses of cross-camera Re-ID task include:

- The basic hypothesis is that the human body detection and tracking in a single view have been already solved using the methods induced in last two chapters.

- To build a robust system, the model should have an advanced generalization performance. Hence, it assumes that sufficient samples which depict the sample distribution could be collected in advance. This point is very different to the cases of single-view object tracking induced in the previous two chapters. The inherent reason is that the labelled samples in object tracking is very limited and generally given in the first frame, thus it is impossible to describe the sample distribution only using these limited samples.

- To associate the persons at diverse locations and different times, it is assumed that some invariant features about the appearance and structure of a human body can be learned to represent individuals.

- Even if these meaningful features can be discovered by some heuristic methods or learning algorithms, it is still difficult to directly compute the similarity between the images captured in different views, because of the problem of ambiguity and uncertainty. Therefore, the general ways to compare the features suppose that a common feature space, in which a certain effective metric will be used, could be explored by some linear or kernel-based methods.

5.1.2 Motivation

In general, the efficiency of matching mainly depends on two aspects: (1) the number of samples stored in the gallery set; (2) the definition of similarity. As for the first aspect, it is impossible to reduce the number of samples. It is because [22]: (1) A large number of surveillance cameras have been installed in public spaces ranging from transport infrastructures, shopping centers, sport arenas to residential streets. These places are always assembled with hundreds of thousands of persons, even in a day. (2) Re-identification in open environments can potentially scale to arbitrary levels, covering huge spatial areas spanning not just different buildings but also different cities, or countries, leading to an overwhelming quantity of "big data". (3) Furthermore, person re-identification can be extended from multi-camera networks to distributed Internet spaces, necessarily across multi-sources over the Internet taking images from, for instance, the Facebook profiles, Flickr and other social media. Therefore, with an explosive growth of images, speeding up the matching stage of a re-identification system by designing a more advantageous similarity criterion is an essential and non-replaceable option.

In this chapter, a novel approach, learning Cross-view Binary Identities (CBI), is proposed to reduce the computational burden for person re-identification. In fact, hashing has been widely used for nearest neighbour search in computer

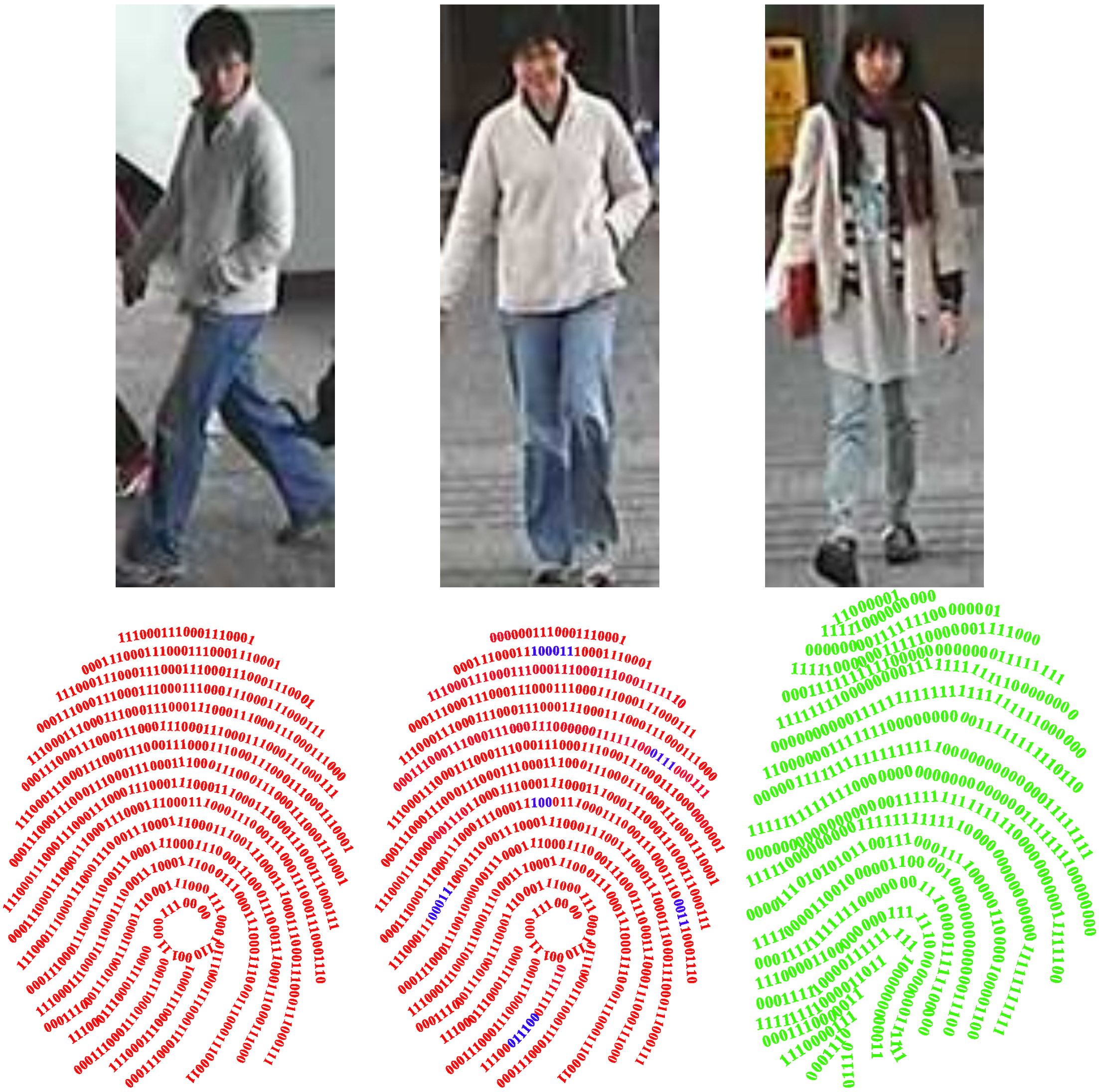


Figure 5.2: Assuming that the left and middle images are of one person and the middle and right images are of different persons but captured by one camera. Our aim is to learn two sets of hash functions (one for each view) which embed the images to binary codes (IDs) so that the IDs (second row) of a same person are similar with each other and the IDs of different persons are quite dissimilar. As illustrated in this figure, the learned binary codes play a same role as fingerprints.

vision areas, such as image retrieval, object recognition and image matching, but it has been seldom used in re-identification. Using the hash functions, various special properties can be preserved in the learned codes, such as locality, variance and affinity. For two observations x_a and x_b of one person in two different views, CBI can learn two similar codes, which are considered as the identity (ID) of that person, as shown in Fig. 5.1.2. The learned binary codes enable efficient similarity search in different views using the Hamming distance between codes. Moreover, compact binary codes are extremely economical for large-scale data storage. Specifically, once the ID of one person in one view is obtained, the ID can be used to search the corresponding person in another view by computing the Hamming distance between two sets of bits y_a and y_b.

5.2 Learning Cross-view Binary Identities

For two different camera views: a and b, we can collect two training datasets $X_a = \{x_a^1, x_a^2, \cdots, x_a^n\}$ and $X_b = \{x_b^1, x_b^2, \cdots, x_b^n\}$, where x_a^i is a column vector observed by view a for person i and n is the number of paired samples (x_a^i, x_b^i). Our aim is to find K hash functions $F = \{f_v^1, \cdots, f_v^K\}$ for each view $v \in \{a, b\}$ and $y_v(k) = f_v^k(x_v)$. In this chapter, the hash functions are constructed by a set of linear hyperplanes: $W_v = \{w_v^1, w_v^2, \cdots, w_v^K\}$. Thus, for dataset X_v, we obtain $Y_v = \{y_v^1, y_v^2, \cdots, y_v^n\}$ by using $y_v^{ik} = sign((w_v^k)^T x_v^i)$. It is obvious that $y_v^i \in \{-1, 1\}^K$. For simplicity, we can write it as: $Y_v = sign(W_v^T X_v)$.

For a person with an image in the probe view, the first step is to calculate the ID by using the learned projections of the probe view. Next, the ID can be used to retrieve the images of persons with similar IDs in the gallery view. The IDs of persons in the gallery view can be obtained in advance. Finally, the person re-identification can be achieved by ranking the Hamming distances. Because the learned pairs of projections can embed the images of a same person into a same ID, the top list of ranking will conclude the ones corresponding to the probe image.

5.2.1 Maximising the variance of bits

We want to produce an efficient code for each view v, in which the variance of each bit is maximised and the bits are pairwise uncorrelated [115]. Thus, we need to do this by maximising the following objective function:

$$\begin{array}{rc} & \mathcal{J}_v = \sum_k var(f_v^k(x_v)) \\ s.t. & cor(f_v^{k_1}(x_v), f_v^{k_2}(x_v)) = 0, \\ & cor(f_v^k(x_v), f_v^k(x_v)) = 1, \end{array} \tag{5.1}$$

where $k_1 \neq k_2$. However, the requirement of exact balancedness makes the above objective function intractable. By signed magnitude relaxation, we get the following continuous objective function based on dataset X_v:

$$\begin{aligned} \mathcal{I}_v = \textstyle\sum_k E(||(w_v^k)^T x_v||_2^2) &\approx \tfrac{1}{n} \textstyle\sum_k (w_v^k)^T X_v X_v^T w_v^k \\ &= \tfrac{1}{n} tr(W_v^T X_v X_v^T W_v) \\ s.t. \quad & W_v^T W_v = I. \end{aligned} \tag{5.2}$$

We relax the constraints as: $((w_v^{k_1})^T w_v^{k_2})^2 < \delta_v, k_1 \neq k_2$, without considering the norm of each linear projection. δ_v is a minimal positive value. In fact, in the following, we can see that it is not necessary to require the unit norm constraints if the linear functions satisfy the hinge loss constraint.

5.2.2 Minimising the Hamming distance

In a single-view problem, the main consideration is that the learned codes are discriminative to represent all the samples by preserving some special properties. However, it is not enough in a multi-view problem, such as person re-identification. Our main goal, in this chapter, is to learn K hash functions for each view so that two observations of each person have the most similar binary codes (IDs). That is to say, the Hamming distance between two sets of codes of one person should be minimised. For a pair of sample sets (X_a, X_b) collected under the two views a and b, the Hamming distance between them is defined as:

$$\mathcal{L}_h(X_a, X_b) = \sum_i D_h(y_a^i, y_b^i), \tag{5.3}$$

where D_h indicates the Hamming distance. The Hamming distance D_h is equal to the number of ones in $y_a^i \oplus y_b^i$, where $\oplus$ is a logical operation that outputs true whenever the inputs differ.

However, despite its efficiency, minimisation of the Hamming distance is generally intractable, because it is non-differentiable to the linear functions. Thus, we seek to minimise an alternative item, which guarantees the Hamming distance will be minimised simultaneously. Fortunately, Proposition 1 shows that we can achieve this, when the linear hash functions satisfy the hinge loss constraint defined as follows.

Definition 1: Hinge loss constraint. *For any sample x_v^i in one view v, if the linear function w_v^k is satisfying*

$$y_v^{ik} (w_v^k)^T x_v^i \geq 1 - \xi_v^{ki}, \tag{5.4}$$

where ξ_v^{ki} is a minimal non-negative value and $y_v^{ik} = sign((w_v^k)^T x_v^i)$, thus w_v^k is the hinge loss constraint satisfied function.

The hinge loss function is used for "maximum-margin" classification, most notably for Support Vector Machines (SVM) [220]. It penalises the items satisfying $y_v^{ik}(w_v^k)^T x_v^i < 1$ so that all items can be correctly classified and the classification score should keep stable as well. In our framework, we hope all the samples can be projected outside of $[-1, 1]$ by each linear function so that the learned codes are relatively stable for all the samples. Moreover, if W_a^k and W_b^k are hinge loss constraint satisfied functions, the Hamming distance between the learned codes are constrained by the Euclidean distance.

Proposition 1: *If two sets of linear projections W_a and W_b for two views are the hinge loss constraint satisfied functions and their corresponding binary codes are defined by $y_v^i = sign(W_v^T x_v^i), v \in \{a, b\}$, thus the inequality can be established when satisfying $\forall k, \xi_a^{ki} + \xi_b^{ki} \leq 1$:*

$$D_h(y_a^i, y_b^i) < ||W_a^T x_a^i - W_b^T x_b^i||_2^2. \tag{5.5}$$

Proof. The Hamming distance between two binary codes y_a and y_b is defined by:

$$\begin{aligned} D_h(y_a, y_b) &= \textstyle\sum_k y_a^k \oplus y_b^k \\ &= \textstyle\sum_k \mathbf{1}(sign((w_a^k)^T x_a) \neq sign((w_b^k)^T x_b)), \end{aligned}$$

where $\mathbf{1}(\cdot)$ is an indicator function. Thus, for any k, we consider two conditions: (1) If $sign((w_a^k)^T x_a) = sign((w_b^k)^T x_b)$, it is obvious that

$$y_a^k \oplus y_b^k = 0 \leq |(w_a^k)^T x_a - (w_b^k)^T x_b|.$$

(2) If $sign((w_a^k)^T x_a) \neq sign((w_b^k)^T x_b)$, we assume that $sign((w_a^k)^T x_a) = 1$ (Otherwise, same conclusion can be also obtained). There must be $sign((w_b^k)^T x_b) = -1$. Since the two linear projections are both hinge loss constraint satisfied functions, we have $(w_a^k)^T x_a \geq 1 - \xi_a^k$ and $(w_b^k)^T x_b \leq -1 + \xi_b^k$. So, there is $2 - \xi_a^k - \xi_b^k \leq |(w_a^k)^T x_a - (w_b^k)^T x_b|$. Provided that $\xi_a^k + \xi_b^k \leq 1$, the following inequalities hold:

$$y_a^k \oplus y_b^k = 1 \leq 2 - \xi_a^k - \xi_b^k \leq |(w_a^k)^T x_a - (w_b^k)^T x_b|.$$

In total, provided with $\mathbf{1}(.)^2 = \mathbf{1}(.)$, we obtain the following conclusion by satisfying $\forall k, \xi_a^k + \xi_b^k \leq 1$:

$$\begin{aligned} D_h(y_a, y_b) &= \textstyle\sum_k \mathbf{1}^2(sign((w_a^k)^T x_a) \neq sign((w_b^k)^T x_b)) \\ &\leq \textstyle\sum_k ||(w_a^k)^T x_a - (w_b^k)^T x_b||^2 \\ &= ||W_a^T x_a - W_b^T x_b||_2^2. \end{aligned}$$

□

5.2.3 Overall objective function

To construct our objective function, three points need to be considered: (1) The cumulative Hamming distance should be minimised while the variance of bits should be maximised, thus

$$\begin{aligned}\mathcal{L}(W_a, W_b) &= \sum_i ||W_a^T x_a^i - W_b^T x_b^i||_2^2 - n \sum_v \mathcal{I}_v \\ &= -2tr(W_a^T S_{ab} W_b),\end{aligned} \tag{5.6}$$

where $S_{v_1 v_2} = X_{v_1} X_{v_2}^T, v_1, v_2 \in \{a, b\}$. (2) For conditions $\xi_a^{ki} + \xi_b^{ki} <= 1$, we can sum all of them over samples and functions to obtain the relaxed inequality $\Upsilon = \sum_{ki} \xi_a^{ki} + \sum_{ki} \xi_b^{ki} <= K * n$. (3) To increase the generalisation of the model, it is necessary to penalise each learned projection by maximising the margin of two separated samples $||W||^2 = \frac{1}{2} \sum_{v \in \{a,b\}} \sum_k ||w_v^k||^2$, which is same as an SVM classification model. Therefore, we obtain

$$\mathcal{L} = \lambda_2 \mathcal{L}(W_a, W_b) + C\Upsilon + ||W||^2, \tag{5.7}$$

where λ_2 and C are used to balance the different types of loss. In Eqn 5.7, $\mathcal{L}(W_a, W_b)$ can be considered as a cross-view loss function for matching, Υ is a within-view quantization loss for hashing and $||W||^2$ is a regularization.

Proposition 2: *Substituting $\mathcal{L}(W_a, W_b)$, Υ and $||W||^2$ into (5.7) with considering the conditions, the objective function can be written as:*

$$\begin{aligned}\{W_a^*, W_b^*\} = \arg\min_{W_a, W_b} & -\lambda_2 tr(W_a^T S_{ab} W_b) \\ & + \frac{1}{2} \sum_k ||w_a^k||^2 + C \sum_{ki} \xi_a^{ki} \\ & + \frac{1}{2} \sum_k ||w_b^k||^2 + C \sum_{ki} \xi_b^{ki} \\ s.t. \qquad & \forall v \in \{a, b\}, i, k, k_1 \neq k_2 : \\ & ((w_v^{k_1})^T w_v^{k_2}))^2 \leq \delta_v, \\ & (y_v^{ik} (w_v^k)^T x_v^i) \geq 1 - \xi_v^{ki}, \xi_v^{ki} \geqslant 0.\end{aligned} \tag{5.8}$$

Firstly, we can see that the proposed CBI is related to Canonical Correlation Analysis (CCA) [221], but without minimising the covariance of intra-module. A solution of CCA may be affected by highly correlated but unimportant (in the sense of low variation and/or covariation) variables. However, a preserved large variance will increase the stability and discriminativeness of the learned codes. Secondly, we can see that S_{ab} is the cross-covariance matrix between the two views a and b. Maximum Cross-variance Analysis (MCA) [222] is a typical dimensionality reduction method for two cross sets of highly correlated variables in the low dimensional space. The proposed CBI can also learn the compact, highly correlated binary codes by maximising the cross-covariance in the new space. Thirdly, although PDH [24] also learns the projection by maximising the margins, there are two significant differences between CBI and PDH. On the one hand, both the cross-variance and the variances of bits have been maximised in CBI but neither of them is considered in PDH. On the other hand, PDH obtains the projection by directly using the classical SVM, but, in CBI, a novel dual problem with a first degree item is solved to learn the projections. That is why PDH cannot improve the performance by increasing the number of bits. Finally, the Hamming distance of two IDs of one person in two different views will be the least when the learned linear hash functions are hinge loss constraint satisfied.

5.3 Optimisation

Despite the complex formula in Proposition 2, in general, the problem can be solved by gradient descend with iterative projection. However, we adopt a more efficient way to search the local optimal solution, considering that the objective is convex to each variable with other variables fixed. Following [223], we can iteratively optimise the projections one by one. The training procedure of CBI is summarised in Algorithm 4.

Algorithm 4 CBI training

```
Input: Training dataset $X_a$, $X_b$ and parameters$\lambda_1$, $\lambda_2$, $C$ and $K$.
Output: $W_a$ and $W_b$.
Initialisation
(0) Solve the SVD problem $tr(W_a^T S_{ab} W_b)$.
(1) Initiate $W_a$ and $W_b$ by the first $K$ left and $K$ right eigenvectors.
Repeat  $t = 1, \cdots$
(2) Choose the kth pair of projections using Eqn. 5.16.
(3) Decide the view order of $v_1$ and $v_2$ to be optimised successively.
    (a) Calculate $\Theta_{v_1}^{\mathbf{k}}$ and $s_{v_1 v_2}^{v_2 \mathbf{k}}$.
    (b) Solve the quadratic programming problem in Eq. 5.14.
    (c) Calculate the projection for view $v_1$ using Eq. 5.13.
    (d) Update the codes of view $v_1$ by $y_{v_1}^{ik} = sign((w_{v_1}^k)^T x_{v_1}^i)$.
(4) Assign the codes for view $v_2$ by $y_{v_2}^{ik} = y_{v_1}^{ik}$.
    (a) Calculate $\Theta_{v_2}^{\mathbf{k}}$ and $s_{v_2 v_1}^{v_1 \mathbf{k}}$.
    (b) Solve the quadratic programming problem in Eq. 5.14.
    (c) Calculate the projection for view $v_2$ using Eq. 5.13.
    (d) Update the codes of view $v_2$ by $y_{v_2}^{ik} = sign((w_{v_2}^k)^T x_{v_2}^i)$.
 If satisfy conditions: Exit.
Return  Update the kth binary codes and hash functions.
```

For further simplifying the optimisation, the orthogonal constraint of projections in intra-module has been added into the objective function. Thus, as shown in Proposition 3, we can see that the problem is the same as the classical SVM but only by adding an item of first degree.

Proposition 3: *We fix all other variables except for w_a^k and ξ_a^{ki}. By removing the irrelevant items, we obtain:*

$$\begin{array}{c} w_a^k = \arg\min \frac{1}{2}||w_a^k||^2 + C\sum_i \xi_a^{ki} \\ -\lambda_2 (w_a^k)^T S_{ab} w_b^k + \frac{\lambda_1}{2}(w_a^k)^T Q_a^k w_a^k, \\ s.t. \quad (y_a^{ik}(w_a^k)^T x_a^i) \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0, \end{array} \tag{5.9}$$

where $S_{ab} = X_a X_b^T$ and $Q_a^k = \sum_{j \neq k} w_a^j (w_a^j)^T$.

5.3.1 Dual problem

The problem in Proposition 3 can be reorgnised as:

$$\begin{array}{c} w_a^k = \arg\min \frac{1}{2}(w_a^k)^T \Theta_a^k w_a^k - (w_a^k)^T s_{ab}^{bk} + C\sum_i \xi_a^{ki}, \\ s.t. \quad y_a^{ik}(w_a^k)^T x_a^i \geq 1 - \xi_a^{ki}, \xi_a^{ki} \geqslant 0, \end{array} \tag{5.10}$$

where $\Theta_a^k = Q_a^k + \lambda_1 I$ and $s_{ab}^{bk} = \lambda_2 S_{ab} w_b^k$.

So far, all variables related to view b have been absorbed into the vector s_{ab}^{bk}. For simplicity, we delete the subscripts of views and the index of projections k in this subsection. The optimal parameters w_a^k and ξ_a^{ki} can be obtained by solving the following objective function:

$$\begin{array}{c} w = \arg\min \frac{1}{2} w^T \Theta w - w^T s + C\sum_i \xi^i, \\ s.t. \quad y^i w^T x^i \geq 1 - \xi^i, \xi^i \geqslant 0. \end{array} \tag{5.11}$$

The objective function becomes a classical convex quadratic programming problem. To simplify the optimisation by transferring inequality constraints to equality constraints, a dual problem is designed. Thus, the Lagrange function can be defined as:

$$\begin{array}{c} L(w, \xi, \alpha, \gamma) = \frac{1}{2} w^T \Theta w - w^T s + Ce^T \xi \\ -w^T X^y \alpha + e^T \alpha - \alpha^T \xi - \gamma^T \xi, \end{array} \tag{5.12}$$

where $e = (1, \cdots, 1)^T$, $\xi = (\xi_1, \cdots, \xi_n)^T$, $\alpha = (\xi_1, \cdots, \alpha_n)^T$, $\gamma = (\xi_1, \cdots, \gamma_n)^T$ and $X^y = (y^1 x^1, \cdots, y^n x^n)$. The gradient with respect to the parameters: $\frac{\partial L}{\partial w} = \Theta w - s - X^y \alpha$ and $\frac{\partial L}{\partial \xi} = Ce - \alpha - \gamma$. Then, the optimal values should satisfy the following constraints:

$$\begin{array}{c} w = \Theta^{-1}(s + X^y \alpha); \\ \gamma = Ce - \alpha. \end{array} \tag{5.13}$$

Substituting the above equations into the original Lagrange function, we obtain the dual problem:

$$\begin{array}{c} \alpha = \arg\min_\alpha \frac{1}{2}\alpha^T (X^y)^T \Theta^{-1} X^y \alpha + (s^T \Theta^{-1} X^y - e^T)\alpha \\ 0 \leq \alpha_i \leq C. \end{array} \tag{5.14}$$

Eqs. 5.13 and 5.14 are similar to the equations in the classical linear SVM. However, it is meaningful to point out the two differences between them, which constitute the advantages of CBI and distinguish from PDH. On the one hand, the inverse of Q in the quadratic item forces that the learned projection must be orthogonal to the other projections within the same view. On the other hand, the s in the first degree item forces that the learned projection should be highly related to the corresponding projection within another view.

5.3.2 Greedy selection

Various ways can be used to initiate the projections, such as random generation. However, to speed up the optimisation, we generate the projections by solving the maximum cross-covariance problem firstly. The first K right and K left eigenvectors of cross-covariance matrix S_{ab} have been chosen to initiate W_a and W_b, corresponding to the first K largest eigenvalues.

Then, the problem becomes how to choose a projection which will be optimised at present. Once one projection has been selected, the new optimal projection will be obtained by solving the problem in Proposition 3. Assume the loss of each projection w_a^k, at the present iteration, is defined as:

$$\begin{aligned} \mathcal{L}(w_a^k) = &\tfrac{1}{2}(w_a^k)^T \Theta_a^k w_a^k - (w_a^k)^T s_{ab}^{bk} \\ &+C\textstyle\sum_i \lfloor 1 - y_a^{ik}(w_a^k)^T x_a^i \rfloor_+, \end{aligned} \tag{5.15}$$

where $\lfloor\rfloor_+$ is the hinge loss function. Therefore, greedy selection will be achieved by:

$$\mathbf{k} = \max_k(\mathcal{L}(w_a^k) + \mathcal{L}(w_b^k)). \tag{5.16}$$

We hope the overall loss will be decreased by minimising the items which have a high loss. The next step is to optimise the selected **k**th pair of projections, which are detailed as follows.

First, view v_1 with less loss will be optimised in advance, because the learned binary codes probably approach the optimal ones. The binary codes $y_{v_1}^{ik}$ of the last round will be considered as the initials to optimise the problem in Proposition 3 for view v_1. Next, the binary codes will be updated according to $y_{v_1}^{ik} = sign((w_{v_1}v^k)^T x_{v_1}^i)$ by using the learned projection. After that, the learned codes of view v_1 will be used to optimise the projection in view v_2. This means y_{v_2} is initiated by y_{v_1}. This process is the same as in [24]. Finally, the same optimisation of Proposition 3 will be conducted for view v_2. Thus, the binary codes of view v_2 will be also updated by $y_{v_2}^{ik} = sign((w_{v_2}v^k)^T x_{v_2}^i)$.

The optimisation procedure can be terminated by different criteria, such as difference between two binary codes of two views less than a small positive number or the fixed number of iterations. In our experiments, we observed that when

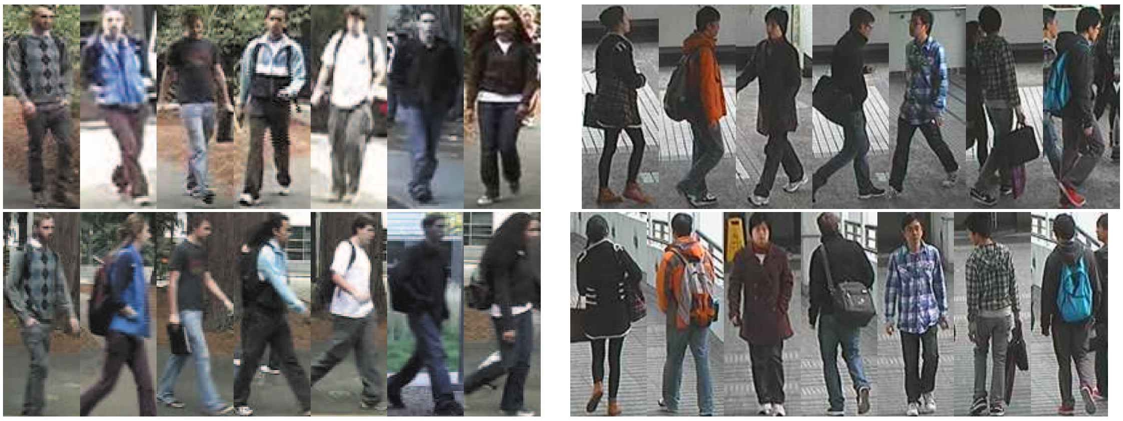


Figure 5.3: Some image samples of the two datasets: VIPeR (left) and CUHK01 (right).

the number of iterations is around the number of projections K, the difference between two binary codes will be the least.

5.3.3 Convergence

In this section, a theoretical analysis is provided by rigorous proof of the convergence of the objective function in Proposition 2.

Proposition 4: *$\mathcal{L}$ in Proposition 2 monotonically decreases with each optimization step for w_a^k and ξ_a^{ki}, and therefore $\mathcal{L}$ converges to a local optimum.*

Proof. Denote $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$ as the objective function in Proposition 3 and R as the remaining which is unrelated to w_a^k and ξ_a^{ki} in Proposition 2, respectively. Then, we obtain the objective function in Proposition 2 $\mathcal{L} = J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n) + R$. At tth step of optimisation, suppose that w_a^k (Otherwise, same conclusion can be also obtained for w_b^k.) has been chosen. Then, we can denote $\mathcal{L}^{t-1}$ as the objective function before optimising w_a^k and $\mathcal{L}^t$ is the function after we obtain the optimum $(w_a^k)^*$ of $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$. Since $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n)$ is a convex problem, there must be $J(w_a^k, \xi_a^{ki} | i = 1, \cdots, n) \geq J((w_a^k)^*, \xi_a^{ki} | i = 1, \cdots, n)$. Moreover, because R is fixed, the following inequality can be established.

$$\cdots \geq \mathcal{L}^{t-1} \geq \mathcal{L}^t \geq \cdots . \tag{5.17}$$

□

5.4 Experiments

We test our proposed CBI for person re-identification on two public datasets: VIPeR [23] and CUHK01 [166]. Some example images of the three datasets are shown in Fig. 5.3. To illustrate the performance and efficiency of CBI, 17 recent algorithms, including 13 person re-identification methods and 4 multi-modal hash function learning methods, are used for comparison.

Image representation: In recent two years, various robust features have been proposed for person re-identification. Especially, the Salient Colour Names based Colour Descriptor (SCNCD) [161] and the Local Maximal Occurrence Feature (LOMO) [160] have achieved promising performance. In this chapter, to reflect the advantage of our CBI to learn binary codes for different descriptors, three types of image representations including SCNCD, LOMO and ELF (Ensemble of Localised Features) which was proposed in [23], are adopted as the basic descriptors. (1) In SCNCD, 16 colour names are used and a colour distribution over the colour names in an image part is computed. SCNCD divides each image into six horizontal stripes of equal size and colour names' distributions of all parts are fused to form an image-level feature. Only the descriptor for the VIPeR dataset is offered by the authors. (2) The LOMO feature analyses the horizontal occurrence of local features, and maximises the occurrence to make a stable representation against viewpoint changes. To handle both the colour constancy and dynamic range compression, a multi-scale Retinex transform is applied. The original dimension of LOMO feature is 26960. (3) ELF descriptor has been used in several methods, such as: [163, 169] and [170]. Each image containing a person was divided into six horizontal stripes. For each stripe, the RGB, YCbCr and HSV colour features and two types of texture features extracted by 13 Schmid and 8 Gabor filters were computed. Thus, each person image was described by a feature vector in a 2784 dimensional feature space. More details are referred to the original paper [23]. CBI is not sensitive to the parameters for the two datasets and we set $\lambda_1 = 2$ and $C = 200$ for all the experiments. However, λ_2 will be set to 0.05, 10 and 5 for ELF, SCNCD and LOMO, respectively.

Evaluation protocol: We randomly partition a dataset into two parts without overlap on person identities, according to a certain percentage. The expectation is reported by conducting 10 trials of evaluation. The parameters of other hashing algorithms are carefully tuned so that the best results are obtained. The results of other person re-identification methods either come from original papers or by running their offered codes, with exactly the same experimental setting. Same as most person re-identification publications, the standard Cumulated Matching Characteristics (CMC) [224] curves and the corresponding Area Under Curve (AUC) are used to illustrate the performance of different methods.

Datasets: The VIPeR contains 632 pedestrian image pairs in an outdoor

Methods	CBI-500	CBI-700	SDALF	KISSME	MLF
Time(s)	1.1e-06	1.4e-06	3.6e+00	9.2e-03	0.98e+01
Methods	PRDC	eSDC	PRSVM	MRank	SCNCD
Time(s)	9.3e-03	1.14e+01	3.2e-03	3.4e-02	4.2e-03

Table 5.1: Time comparison of computing the similarities between one probe sample and all the gallery samples (316) using the compared methods. CBI-500 denotes that only 500 hash codes have been learned.

environment. Each pair contains two images of the same individual taken from two different camera views. Changes of viewpoint, illumination and pose are the most significant causes of appearance change. Each image has been scaled to be 128 × 48 pixels. The experimental setting is the same as [170]. Half of the dataset including 316 images for each view is used for training the algorithms and the reminding (316 pedestrian) is used for testing. The CUHK01 contains 971 pedestrians and is also captured with two camera views in a campus environment but each pedestrian has two images from each camera view. Camera A captures the frontal view or back view of a pedestrian, while camera B captures the side view. All the images are normalized to 160×60 for evaluations. Our two settings follow [166] (100 test persons and 871 persons for training and [225] (486 test persons and the remaining as training samples).

5.4.1 The efficiency of CBI

The Hamming distance comparison of the learned binary codes for two different persons in two views on the VIPeR dataset is shown in Fig. 5.4. Two persons with similar appearances are selected. According to the proposed CBI, binary codes with length 704 for each image are learned and resampled into an image with 22 × 32 pixels so that it is easy to illustrate the difference of learned codes. From this figure, we can see that the Hamming distance between two images of a same person in two views is much lower than that between the images of different persons no matter they are captured in the same view or not.

CBI is efficient for similarity search in the testing stage, since the bit XOR operation is applied when calculating the Hamming distance between binary codes. To illustrate the efficiency of CBI, we compare the time of similarity computation for various methods on the VIPeR dataset. To simulate a real situation, the time includes the feature projection for the probe image but the embedded features of gallery images are obtained in advance. All algorithms are run on a Matlab 7 platform installed on Windows 7 with Intel Core 3.4GHz CPU and 8M memory. The codes of compared methods are provided by their original authors and comparison results are shown in Table 5.1. We can see that the proposed

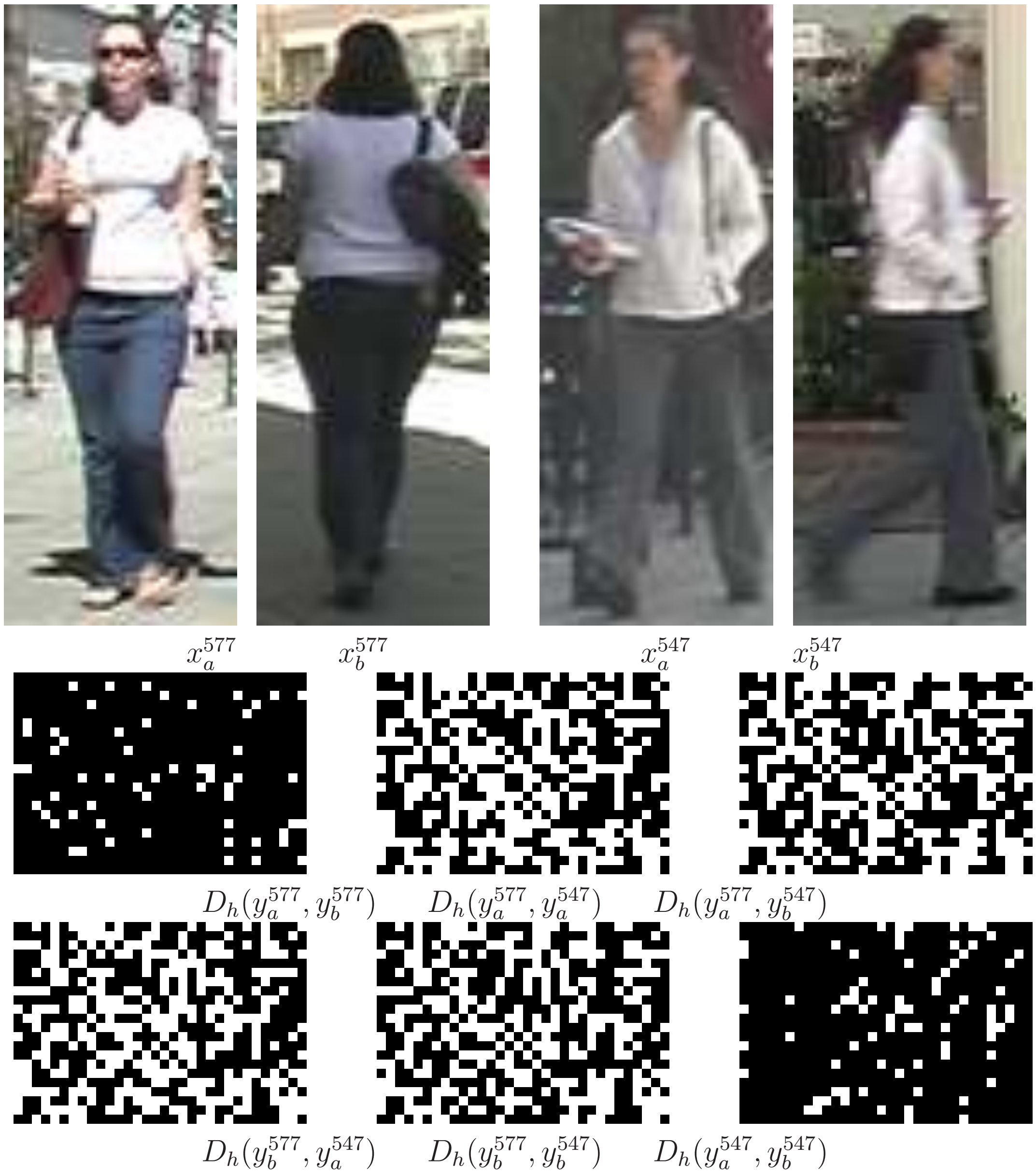

x_a^{577}
x_b^{577}
x_a^{547}
x_b^{547}
$D_h(y_a^{577}, y_b^{577})$
$D_h(y_a^{577}, y_a^{547})$
$D_h(y_a^{577}, y_b^{547})$
$D_h(y_b^{577}, y_a^{547})$
$D_h(y_b^{577}, y_b^{547})$
$D_h(y_a^{547}, y_b^{547})$

Figure 5.4: Top row: the images in the two views of the 577th (left two) and 547th (right two) persons from test sets in the VIPeR dataset. Middle and bottom rows: the Hamming distances between the learned codes for the four samples and the exact Hamming distances are 57, 331, 317, 328, 316 and 78, respectively.

CBI is at least 2200 times faster than other non-hashing methods. It is worth to point out that the local patches based methods, including eSDC [164], MLF [225] and SalMatch [226], achieve advantageous performance (Rank 1: eSDC-26.74% in Fig. 5.6). However, this group of methods exploiting the local patches introduces a huge computational burden and they are 10^7 times slower than CBI.

We set the original dimension of the feature and the number of samples in the gallery set to be n_d and n, respectively. Thus, in CBI, the total complexity will be Kn_d multiplication and $K(n_d - 1)$ addition operations for projection and nK logical operations for computing the Hamming distance. If all the codes are implemented by a low-level programming language, the advantages of CBI in terms of XOR operation will become even more significant. Moreover, if we build the relationship between the learned binary codes and the physical addresses in a real-world application, the addressing time is $O(1)$ and the retrieval task can be achieved without any similarity computation. However, if the Euclidean distance is directly considered as the measurement of similarity, besides the projections, there are more nK multiplication and $2nK$ addition operations. Metric learning is a very popular method for the retrieval and identification tasks, such as PRDC and KISSME [171]. The distance in this group of methods normally has the form: $(\triangle x)^T M \triangle x$, where M should be learned in some cases and can be set as covariance matrix for Mahalanobis distance[1]. Thus, for one test sample, the complexity of retrieval tasks in a gallery set will be $nn_d(n_d + 1)$ multiplication and $n(n_d + 1)(n_d - 1)$ addition operations. Therefore, in theory, the efficiency of CBI is at least $n(n_d + 1)/K$ times faster than metric learning based methods. In general, in a real-world application, the number of samples in the gallery set n is huge and the original dimension n_d is much larger than the number of learned bits K.

5.4.2 Comparison with the state-of-the-art methods

For evaluating on the VIPeR, we compare the proposed CBI for person re-identification with recent published algorithms, including: ELF [23], PRDC [170], PRSVM [169], SDALF [227], CPS [228], Mrank [229], eSDC [164], SalMatch [226], MLF [225], KISSME [171], SCNCD [161] and LOMO [160]. The comparison results are shown in Fig. 5.5 and 5.6. Among them, PRDC, Mrank and PRSVM used the ELF feature. We can see that the proposed CBI achieves much better results than the three methods almost in all ranking. Following [160], a Cosine similarity measure is applied to SCNCD and LOMO. To compare with the three types of original features, we can see that the performance is boosted by CBI

[1] In computing the Mahalanobis distance, the quantity $\triangle x$ is a difference vector between the pair of compared samples and can not be precomputed before the probe sample is obtained.

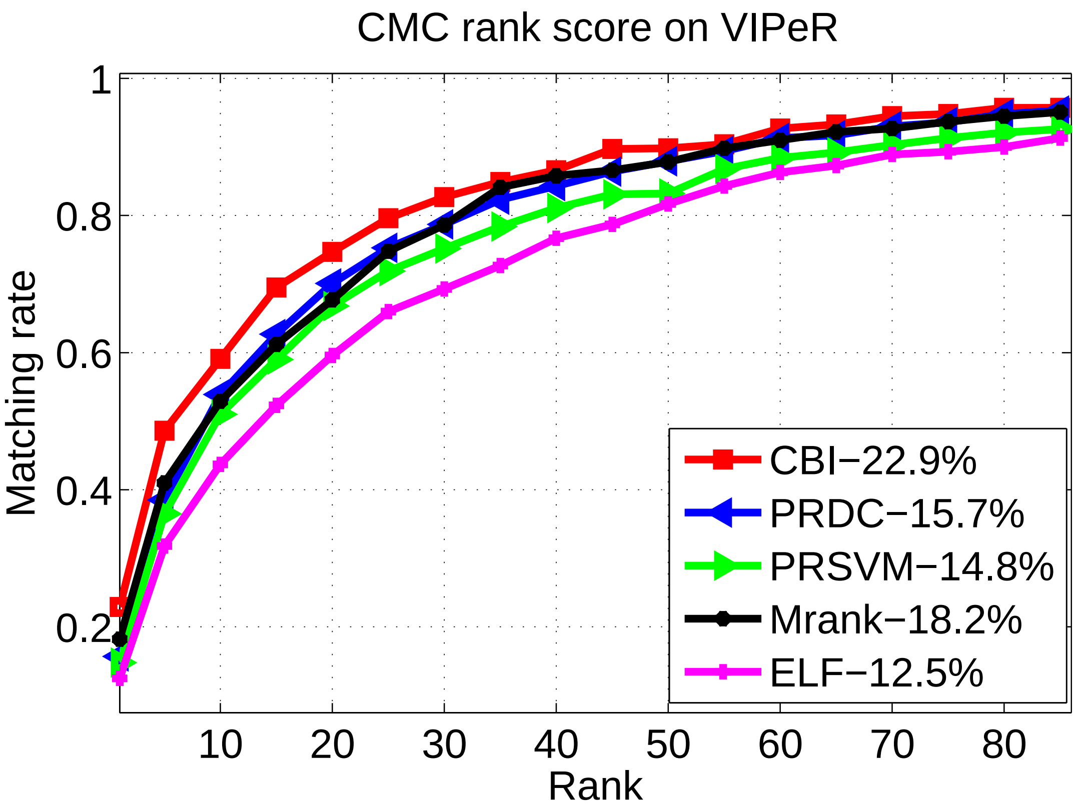


Figure 5.5: The CMC rankings of the methods using the ELF feature on the VIPeR datasets with #316 test persons. Numbers in legend is the Rank-1 accuracy.

for at least 30%. Moreover, using the SCNCD feature, CBI is the best method at rank 1 and is better at low ranks (≤ 30) than other state-of-the-art methods. Finally, by using LOMO feature, CBI has a 29.1% accuracy at rank 1 and outperforms other methods almost at all ranks.

For comparing on the CUHK01, we follow two partitions as in [166] and [225] and the results are shown in Fig. 5.7 and 5.8. For the first partition with 100 test persons, three methods including FPNN [166], eSDC [164] and SDALF [227] are compared with. We can see that CBI can achieve much better results than eSDC and SDALF at all ranks, no matter what features are used. To compare with the deep architecture based method FPNN, using the LOMO feature, CBI can achieve better results at all ranks while, using the SCNCD feature, is only slightly inferior FPNN at rank 1 (1.6 %) but better than FPNN at all other ranks. For the second partition with 486 test persons, the task is relatively more difficult and

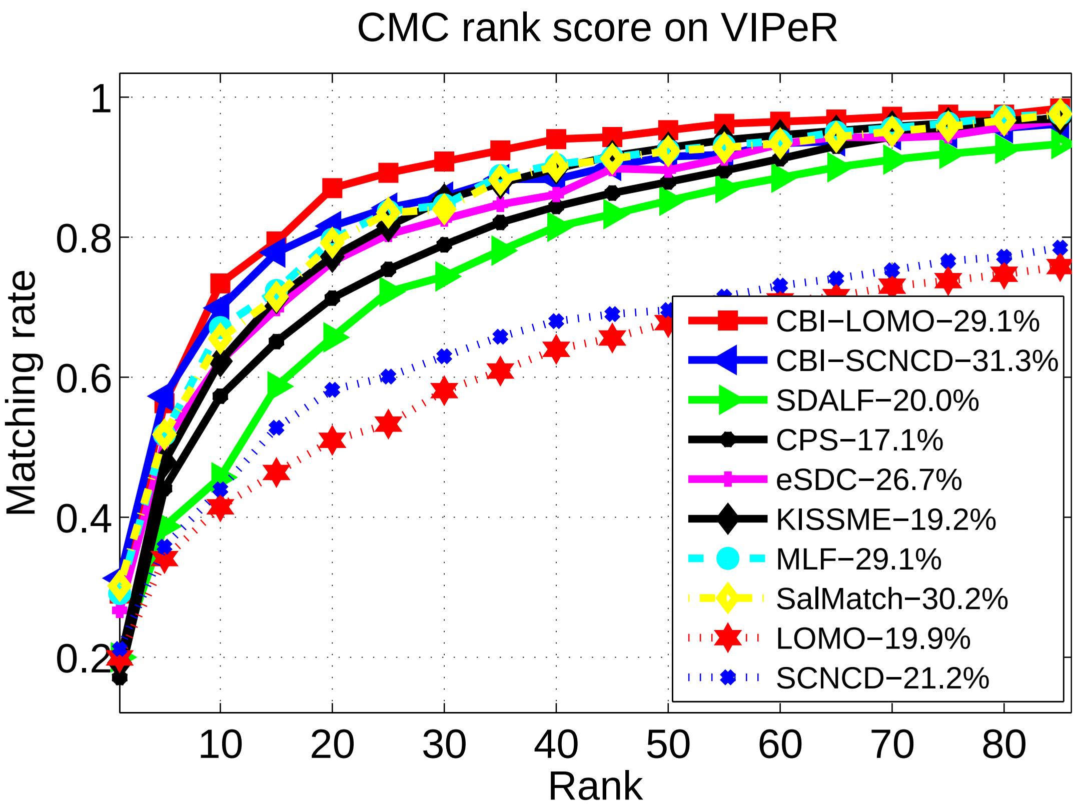


Figure 5.6: The CMC rankings of the state-of-the-art methods on the VIPeR datasets with #316 test persons. The two types of features SCNCD and LOMO are used by the proposed CBI.

four state-of-the-art methods including eSDC [164], SDALF [227], SalMatch [226] and MLF [225] are compared against. In this setting, MLF is the best method but, using the LOMO feature, CBI achieves very similar performance to MLF. In fact, MLF is a local patches based method thus the computational burden of feature calculating and matching is very high.

In total, CBI can achieve competitive performance with the state-of-the-art methods. We have to point out that recent works on improved deep learning [167] and fusion based methods (LOMO+XQDA [160], MLF+LADF [225], mixture of similarities [173] and ensemble of distances [174]) reported higher results. However, since this chapter mainly focuses on the efficiency, the combination of different methods is not considered, because they are computationally very expensive. Naturally, the performance of binary coding, a.k.a. hashing, methods will be lower than their corresponding non-hashing based methods due to the quanti-

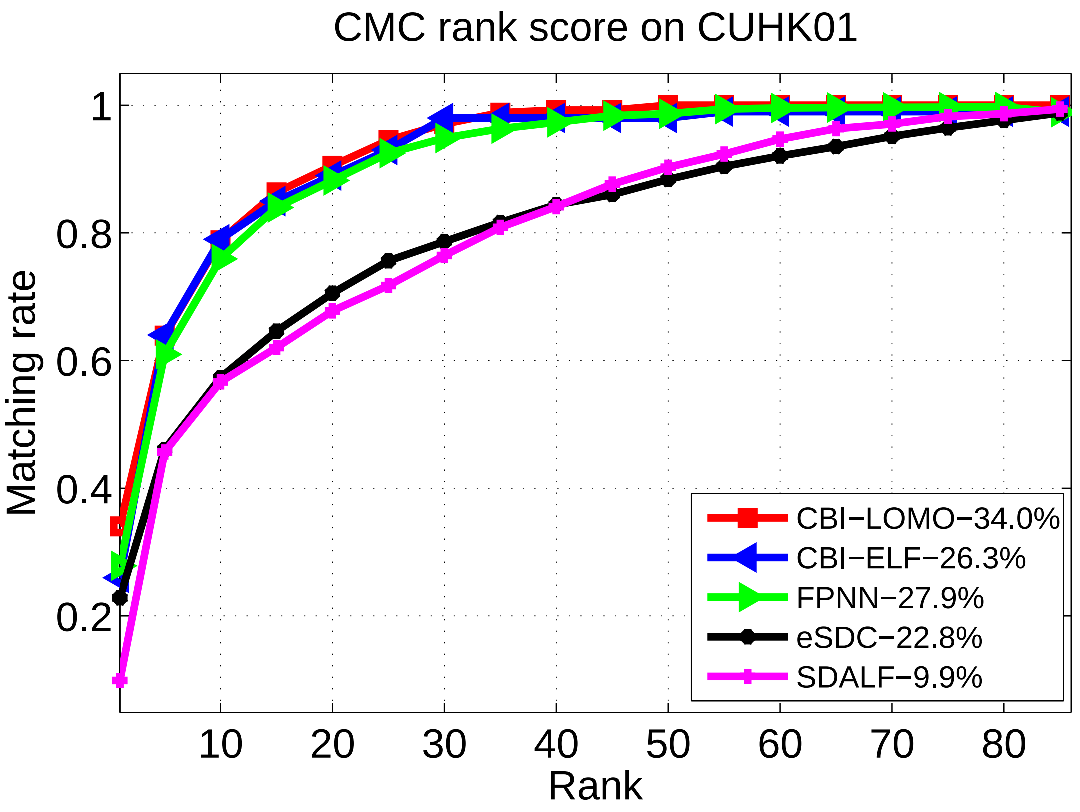


Figure 5.7: The CMC rankings on the CUHK01 datasets with #100 test persons. The two types of features ELF and LOMO are used by the proposed CBI.

zation loss [115]. In the following section, the comparison of CCA [221] and CVH [125], which is a hashing version of CCA, will also prove this point. Therefore, we can conclude that, as a binary coding method for person re-identification, the performance of CBI is acceptable.

5.4.3 Comparison with other hashing methods

We compare our CBI with CCA [221] and recently proposed multi-modal binary code learning methods, including PDH [24], CVH [125], CMSSH [126] and CFMH [128] on the VIPeR and CUHK01 datasets. Most of the compared methods are used for cross-view searching, such as across text and image. Normally, the intra-module or inter-module relationship between the samples has been considered in these methods. In fact, for the problem of person identification, discriminative representation is more important than the preserved local properties, especially for one-shot recognition. In this experiment, half of the persons (VIPeR: 316 test

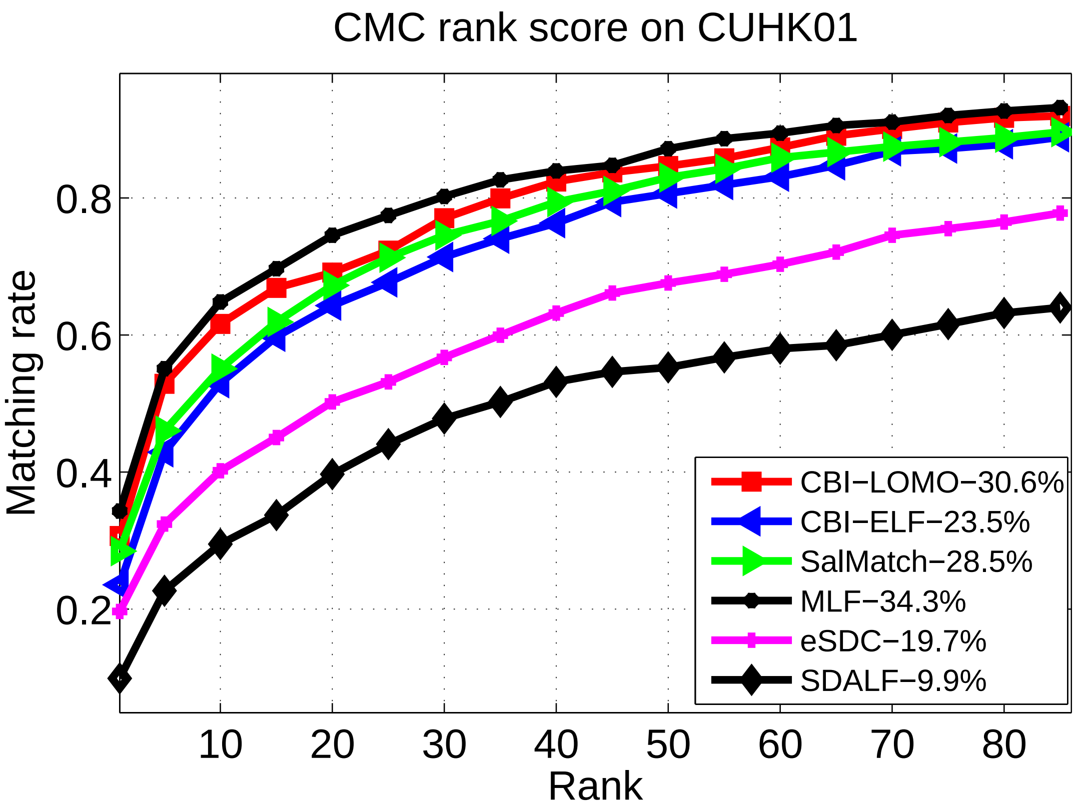


Figure 5.8: The CMC rankings on the CUHK01 datasets with #486 test persons. The two types of features ELF and LOMO are used by the proposed CBI.

persons and CUHK01: 486 test persons) are chosen for test and the remaining persons are used for training. The same features are used as input in all the methods for fairness.

The comparisons of ranking scores vs. dimensions of learned codes at ranks 1, 10 and 20 for the three features including ELF, SCNCD and LOMO are shown in Figs. 5.9, 5.10 and 5.11, respectively. Firstly, due to the finite rank of the variance matrix, the dimensions of the features learned by CCA and CVH are constrained, thus their best performance is poor. Secondly, at very low dimensions, most methods achieve similar results and the performance of CCA is better than others. However, the hashing version CVH of CCA achieves much lower results than CCA and is also lower than other methods in most cases. Thirdly, in general, most methods achieve better when the length of learned codes is longer. However, after a certain length, the ranking scores of other three methods including CMFH, PDH and CMSSH do not progressively increase by the increase of the length of

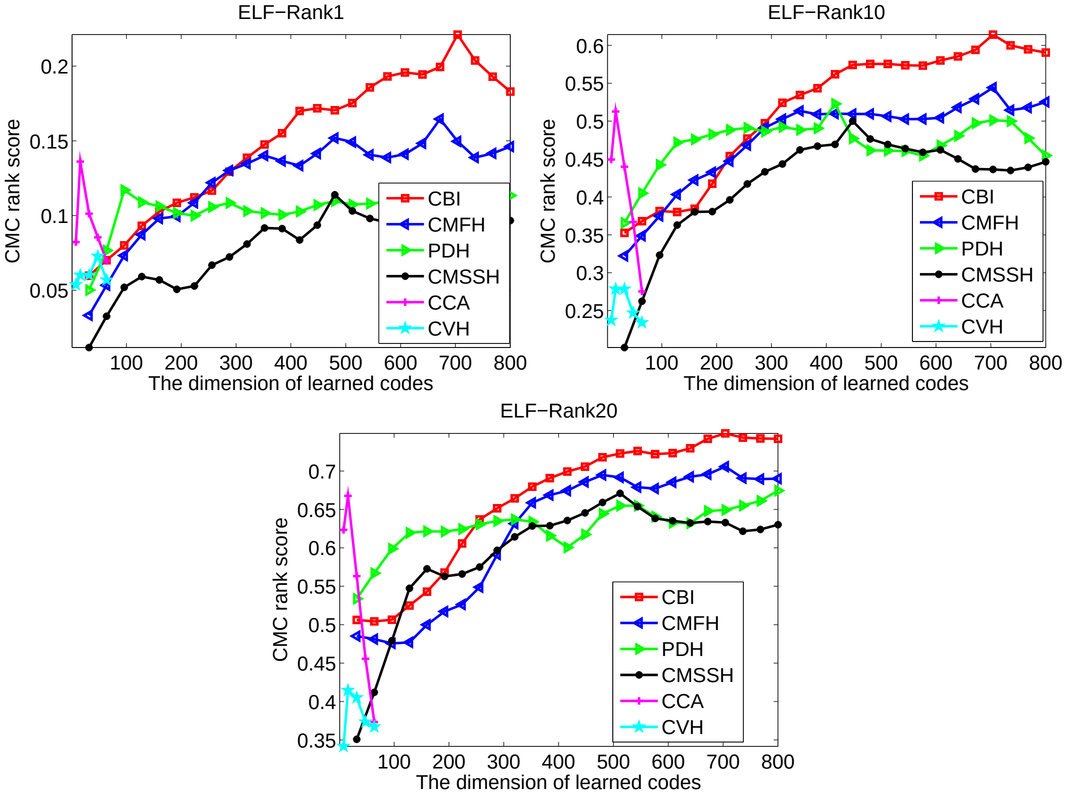


Figure 5.9: Comparison with four multi-modal hashing methods using ELF feature on the VIPeR dataset at ranks: 1 (left of first row), 10 (right of first row) and 20 (second row).

codes. This is because the later learned codes tend to add little discriminative information, due to ignoring the orthogonal constraint between different hash functions. Fourthly, our proposed method achieves much better results than other methods when the code length is over 400. Finally, we can see that features also play an important role for improving the performance and most methods achieve better results when using advantageous features.

Furthermore, the overall AUC performance for ranks 1 to 85 and the ranking accuracies at ranks 1, 5, 10, 15 and 20 for the three features on the VIPeR dataset are shown in Tables 5.2, 5.3 and 5.4, respectively. The CMC matching scores in the tables are computed at the dimension when the methods achieve the best performance. From the tables, we can see that CCA achieves much better results than the corresponding hashing version CVH at all ranks and, in some cases, the accuracies of CCA are almost twice to CVH. Moreover, no matter what features are adopted, we can observe that CBI outperforms all other binary code learning

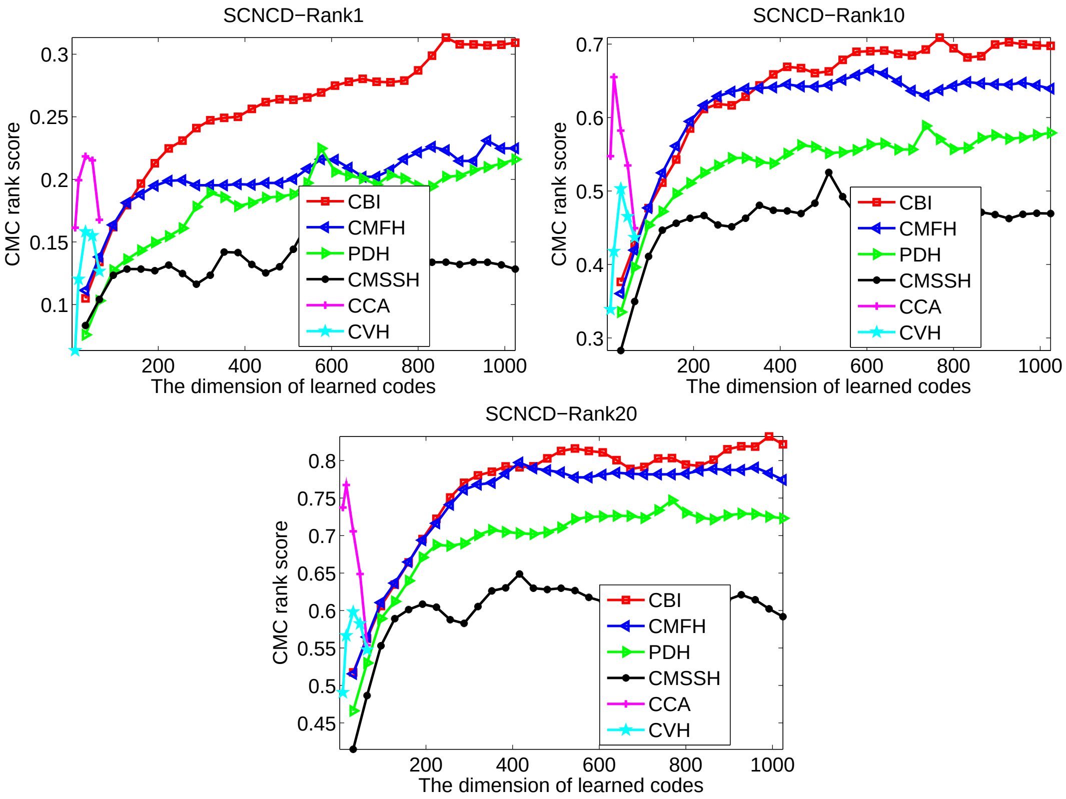


Figure 5.10: Comparison with four multi-modal hashing methods using SCNCD feature on the VIPeR dataset at ranks: 1 (left of first row), 10 (right of first row) and 20 (second row).

methods at all ranks, and from the perspective of overall AUC performance, CBI is also the best method. It is worth to point out that the advantages are more highlighted at lower ranks and this can be also reflected from Figs. 5.9, 5.10 and 5.11. For example, CBI achieves at least 4.4% higher result than other methods at rank 1, no matter what features are adopted.

CBI is also compared with the five methods on the CUHK01 dataset using two features ELF and LOMO, and the comparisons are shown in Tables 5.5 and 5.6. From the tables for the CUHK01 dataset, we can draw the same conclusions as on the VIPeR dataset that CBI performs the best among the six methods. For rank 1, CBI achieves at least 8.6% higher results than others.

Methods	CBI	CMFH	PDH	CMSSH	CCA	CVH
Rank 1	0.229	0.165	0.130	0.108	0.136	0.073
Rank 5	0.486	0.402	0.380	0.310	0.345	0.190
Rank 10	0.591	0.549	0.519	0.475	0.513	0.247
Rank 15	0.695	0.647	0.614	0.573	0.595	0.310
Rank 20	0.747	0.716	0.715	0.655	0.668	0.373
AUC	80.28	77.18	75.63	73.96	73.49	50.12

Table 5.2: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using ELF feature on VIPeR dataset.

Methods	CBI	CMFH	PDH	CMSSH	CCA	CVH
Rank 1	0.313	0.231	0.222	0.165	0.199	0.158
Rank 5	0.573	0.500	0.440	0.389	0.513	0.380
Rank 10	0.699	0.639	0.576	0.491	0.655	0.503
Rank 15	0.782	0.718	0.658	0.570	0.711	0.566
Rank 20	0.826	0.769	0.747	0.620	0.767	0.598
AUC	84.09	82.58	80.26	72.84	79.90	66.82

Table 5.3: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using SCNCD feature on VIPeR dataset.

Methods	CBI	CMFH	PDH	CMSSH	CCA	CVH
Rank 1	0.291	0.247	0.171	0.190	0.168	0.085
Rank 5	0.563	0.528	0.449	0.437	0.427	0.209
Rank 10	0.734	0.712	0.604	0.639	0.551	0.294
Rank 15	0.794	0.766	0.693	0.725	0.633	0.345
Rank 20	0.858	0.816	0.778	0.791	0.693	0.399
AUC	86.67	85.20	80.40	81.19	75.98	54.04

Table 5.4: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using LOMO feature on VIPeR dataset.

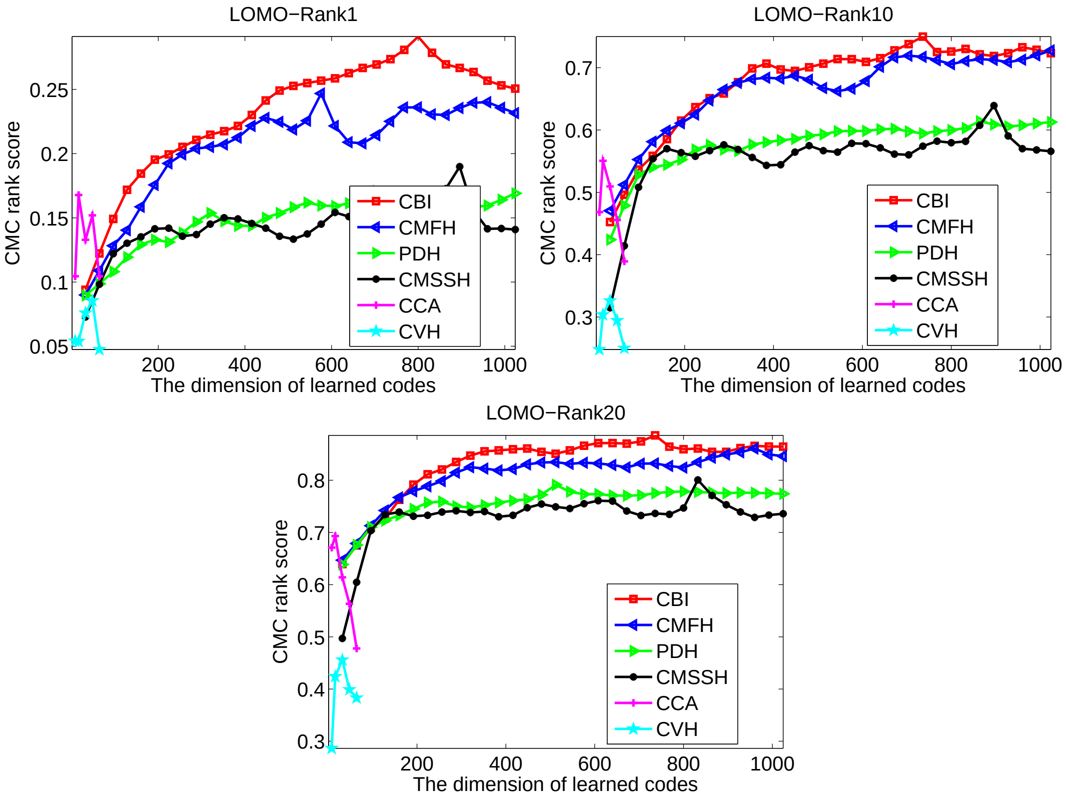


Figure 5.11: Comparison with four multi-modal hashing methods using LOMO feature on the VIPeR dataset at ranks: 1 (left of first row), 10 (right of first row) and 20 (second row).

5.4.4 Optimisation analysis

In this subsection, the procedure of optimisation is investigated. To demonstrate how to find the local optimal hashing functions, we run CBI on the VIPeR dataset using the LOMO feature and set the length of binary codes to 800. Normally, the algorithm will be terminated after 800 iterations, but to see more about the optimisation, the number of iterations is set to 850. Moreover, to reflect the ability of greedy searching, all the projections are initialised randomly and then are normalised. From the definition of Hamming distance, we understand that the largest Hamming distance between two binary codes is the length of codes, in which all the bits are different. A quantity which is an averaged ratio (Hamming distance ratio) between the Hamming distance and the length of code for all pairs of samples is defined to measure the similarity between two sets of binary codes. If this quantity equals 0, the two sets are completely same and 1 means they are

Method	CBI	CMFH	PDH	CMSSH	CCA	CVH
Rank 1	0.236	0.150	0.057	0.103	0.156	0.066
Rank 5	0.429	0.340	0.159	0.251	0.335	0.193
Rank 10	0.530	0.475	0.255	0.344	0.457	0.282
Rank 15	0.598	0.566	0.311	0.401	0.533	0.340
Rank 20	0.643	0.613	0.356	0.453	0.580	0.397
AUC	0.719	0.680	0.459	0.532	0.643	0.493

Table 5.5: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using ELF feature on CUHK01 dataset.

Method	CBI	CMFH	PDH	CMSSH	CCA	CVH
Rank 1	0.307	0.188	0.059	0.100	0.153	0.060
Rank 5	0.529	0.499	0.158	0.266	0.406	0.163
Rank 10	0.616	0.606	0.250	0.355	0.529	0.264
Rank 15	0.669	0.672	0.318	0.413	0.592	0.329
Rank 20	0.691	0.714	0.366	0.469	0.634	0.377
AUC	0.771	0.759	0.458	0.539	0.695	0.484

Table 5.6: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison, using LOMO feature on CUHK01 dataset.

totally different. In Fig. 5.12, three aspects including Hamming distance ratio, ranking accuracy (at ranks 1, 10 and 20) of training set and ranking accuracy of test set vs. the number of iterations are investigated. Firstly, by the increase of the iteration number, the Hamming distance ratio of the training set gradually decreases and the descend speed is stable. In this case, the optimal hash functions are obtained at the ratio of 0.221. Secondly, the ranking accuracies of both the training set and the test set increase generally but not constantly. Moreover, the accuracy at rank 1 of the test set increases stably but the ascend speeds of other ranks are faster at the front of optimisation than the later. Finally, we can see that the accuracies of the training set increase faster than the ones of the test set and most accuracies will converge after the number of iterations is around the length of binary codes.

5.5 Extension

In this chapter, we mainly focus on single-shot person re-identification across two views $n_v = 2$, but the proposed framework can be easily extended into $n_v(n_v > 2)$ and multi-shot situations. Although, in the CUHK01 dataset, each pedestrian has two images from each camera view, most methods [166, 225, 226] have not

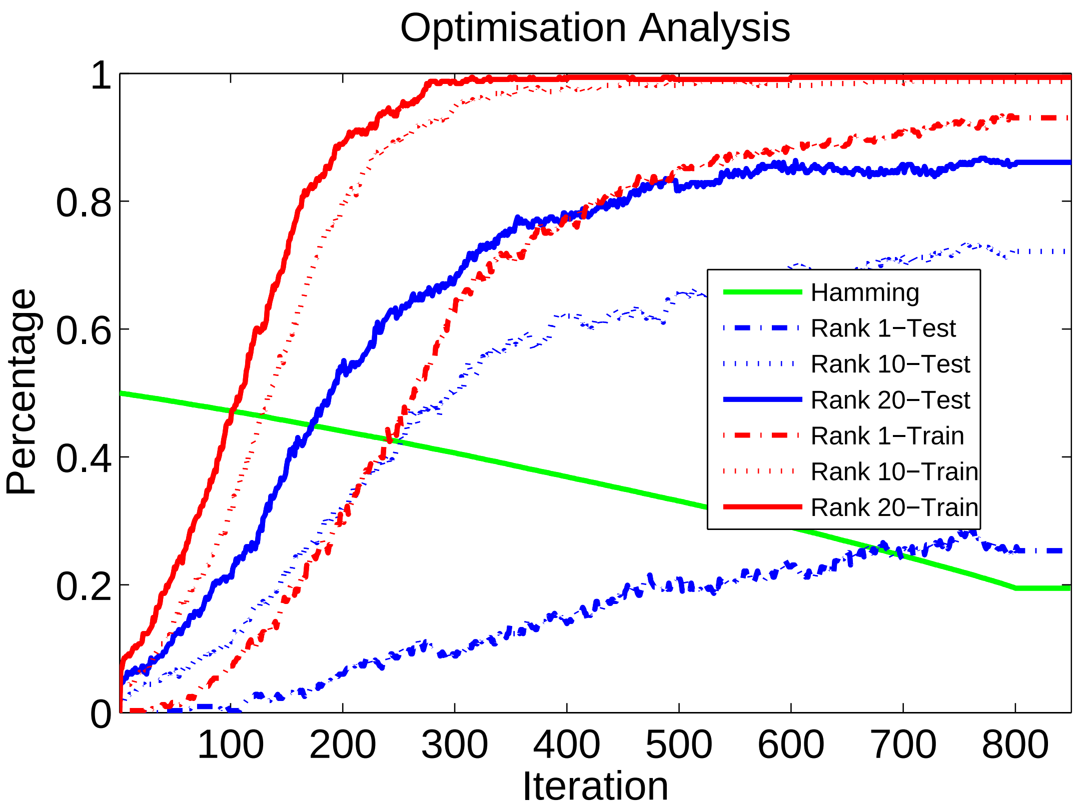


Figure 5.12: The optimisation analysis. In the legend, Hamming means "Hamming distance ratio", test refers to "Test set" and train refers to "Training set".

considered the relationships between any pair of images from both same view and different views.

In a real world scenario, even in a building or a shopping mall, much more than two cameras $n_v(n_v > 2)$ are installed to monitor the human activities. Therefore, learning the IDs of persons from more than two views is useful. In this multi-view case, we hope n_v projections $\mathcal{W} = \{W_1, \cdots, W_{n_v}\}$ will be learned considering that the Hamming distance between any two views v_1 and v_2 in the learned space should be minimised. Thus, the overall objective function in Eq. 5.6 will become:

$$\begin{aligned} \mathcal{L}(\mathcal{W}) = & \textstyle\sum_{v_1, v_2 \neq v_1} \sum_i ||W_{v_1}^T x_{v_1}^i - W_{v_2}^T x_{v_2}^i||_2^2 \\ & -(n_v - 1)n \textstyle\sum_{v=1}^{n_v} \mathcal{I}_v \\ = & -2tr(\textstyle\sum_{v_1, v_2 \neq v_1} W_{v_1}^T S_{v_1 v_2} W_{v_2}). \end{aligned} \tag{5.18}$$

Because CBI optimises the projections on each view v_1 separately with other projections fixed, when there is $n_v > 2$, it can be easily proved that the projections can be learned using the objective function in Eq. 5.10 by directly computing $s_{v_1} = \sum_{v_2 \neq v_1, v_2=1}^{n_v} \lambda_2 S_{v_1 v_2} w_{v_2}^k$.

Furthermore, CBI can also be generalised to multiple-shot cases by considering two aspects. In such cases, we have $X_v = \{X_v^1, X_v^2, \cdots, X_v^n\}$ where X_v^i is the sample set of person i in view v. On the one hand, different images of one person in a same view should have the same identity. Thus, we hope item $\sum_{i,j} ||W_v^T x_v^i - W_v^T x_v^j||_2^2 A_v^{i,j}$ can be minimised, where $A_{i,j} = 1$ if image samples x_v^i and x_v^j belong to a same person, otherwise $A_v^{i,j} = 0$. By defining the Laplacian matrix L_v using the affinity matrix A_v, this item can be rewritten as $tr(W_v^T X_v L_v X_v^T W_v)$. On the other hand, the Hamming distances between any image pair of one person for any two different views $\sum_{j_a, j_b} ||W_a^T x_a^{i,j_a} - W_b^T x_b^{i,j_b}||_2^2$ should be minimised, where $x_v^{i,j_v} \in X_v^i$ denotes the j_vth samples of person i in view v. Next, if we define $S_{v_1,v_2}^i = X_{v_1}^i (X_{v_2}^i)^T$, the item can be rewritten as $tr(\sum_{v \in \{a,b\}} W_v^T S_{vv}^i W_v - 2W_a^T S_{ab}^i W_b)$. In summary, for the multi-shot cases, the overall objective function can be defined as:

$$\begin{aligned} \mathcal{L}(\mathcal{W}) = \textstyle\sum_i tr(\sum_v W_v^T S_{vv}^i W_v - 2W_a^T S_{ab}^i W_b) \\ -n \textstyle\sum_v \mathcal{I}_v + \sum_v tr(W_v^T X_v L_v X_v^T W_v) \\ = -2tr(W_a^T S_{ab} W_b + \textstyle\sum_v W_v^T \mathcal{K}_v^i W_v), \end{aligned} \tag{5.19}$$

where the cross-variance matrix $S_{a,b} = \sum_i S_{ab}^i$ and $\mathcal{K}_v = X_v(L_v - I)X_v^T + \sum_i S_{vv}^i$. Therefore, same as in (5.10), an iterative optimisation method can be applied to obtain the hash functions.

5.6 Summary

In this chapter, a cross-view binary code learning method has been proposed for fast person re-identification. The main advantage of this method is that it hugely speeds up the procedure of the ranking or retrieval stage, when achieving equivalent performance to the state-of-the-art methods. Moreover, three more important points have also been observed. Firstly, just the heuristic hand-craft descriptors are used in this chapter and we think that utilising a stronger pixel-based descriptor which is learned using deep architecture will improve CBI a lot. Secondly, maximum margin has been used in learning binary codes by other methods. However, we firstly give an inside view of the intrinsic mechanism that the Hamming distance can be minimised by minimising the Euclidean distance when the learned linear hash functions satisfy the hinge loss constraint. In the future, it is meaningful to give a more compact boundary via the statistical perspective to enable a faster convergence of the algorithm. Finally, just dual modules have been used to learn the IDs of different persons. However, learning the IDs of persons from more than two views is useful. From this point of view, we can see that CBI is just a starting point in this area. Whether other information, such as the topology of the camera system and other biological features, would also benefit the learning of more robust IDs is still under investigation.

Chapter 6

Hetero-manifold Regularisation for Cross-modal Hashing

In the past three chapters, visual data association in both the signal camera setting and the cross-camera setting are discussed. In this chapter, we will turn to more general cases where the visual data could be associated to other types of data, such as text and voice, see Fig. 6.1. Compared to the fist two situations, the model of cross-modal retrieval is more complex, because the data between different modalities have diverse structures and different physical meanings. Moreover, the methods of cross-modal retrieval could be used to, in a higher level, connect the traces, which are detected both in a single view or across cameras, to other activities which need to be described using other types of data. For instance, by identifying a person in a view of camera which is set to public areas through some virtual activities on Internet, many potential security threats can be found before taking action. Furthermore, traces of a criminal which are detected by both the single camera tracking and the cross-camera tracking could be fully connected using some texts written by a witness. To this end, beside object tracking methods across cameras, it is necessary to develop some methods of cross-modality data association. In fact, cross-modal searching is the most popular strategy to address such kind of problems.

Searching is dramatically changed by the amount and the appearance of multimodal data. Multi-modal data are heterogeneous and large-scale because of the advancement of digital technologies and the Internet. Both of these fundamental characteristics of multi-modal data require measuring the cross-modal similarity when developing any searching algorithms by hashing.

To bridge the gap between modalities, various straightforward strategies have been developed to learn the cross-modal similarity. Some methods focus on the supervision information including correspondences [179], semantic correlation [180], pairwise sets [182] and semantic affinities [230] between heterogeneous data,

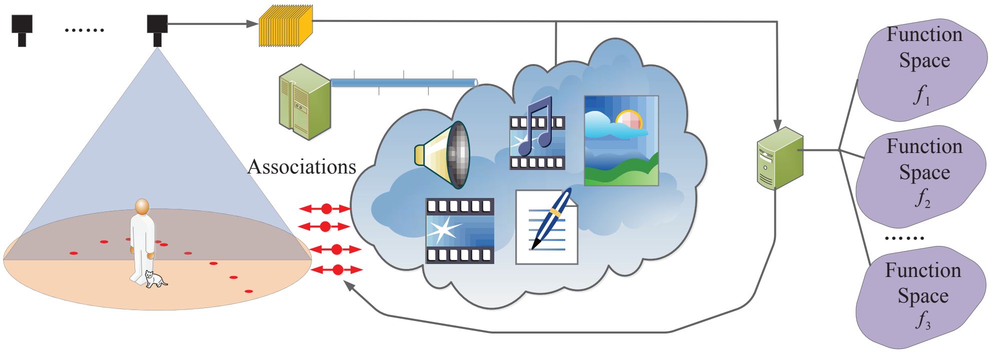


Figure 6.1: Visual data association in a cross-modal setting using a novel Hetero-manifold regularised hashing.

while others including composite multiple information sources [124], α-average technique [189, 231], Markov random field [232] and deep neural networks [181] emphasise the value of homogeneous manifold in the problem of multi-modal similarity learning in a common space. In this chapter, by integrating the supervision information and the local structure of heterogeneous data, a novel method termed hetero-manifold regularisation (HMR) is proposed to learn hash functions for efficient cross-modal search.

The rest of this chapter is organised as follows. The hypotheses and motivation of this work is introduced in Sec. 6.1. In Sec. 6.2, how to construct a Hetero-manifold for multi-modal data is detailed. Next, based on the introduced Hereo-manifold, how to learn a set of hash function for cross-modal retrieval will be presented in Sec. 6.3. Then, Sec. 6.4 presents how to solve the complicated objective function and convergence proof. Sec. 6.5 presents comprehensive experimental results for four datasets. Section 6.6 draws summary.

6.1 Preliminaries

6.1.1 Hypotheses

In this study, the principal aim is to connect and integrate all the information contained in different modalities into a uniform framework. Then, based on the integrated structure, a set of projections can be learned and the samples from different modalities can be embedded into a common space. The hypotheses behind this, which are used to support the solution in this thesis include:

- High dimensional data tend to lie in the local structure of a low dimensional manifold. This is a basic assumption for the classical manifold learning in a uni-modal setting, but it is still useful in cross-modal tasks. Based on this point, a sub-manifold can be modelled for each modality to capture the intrinsic structure of intra-manifold.

- The sub-manifolds in different modalities could be connected by some supervised information, or latent variables, which can be defined in diverse forms, considering the specificity of the problems. This is a basic assumption to support the connectivity and integrity of hetero-manifold in this work.
- The information could be propagated on an integrated framework in the cross-modal setting. Only when the information can be diffused in a certain pattern, can a global view be built based on these sub-manifolds so that it enables to cross-modal retrieval.

6.1.2 Motivation

However, despite the progress made by existing methods considering certain aspects of the problem, cross-modal search remains a very challenging task because of the integration complexity and heterogeneity of the multi-modal data. In fact, the nature of multi-modal data is a combination of heterogeneity and the homogeneity. Thus, in cross-modal search, the cross-modal and within-modal similarity information should be simultaneously considered. On the one hand, the methods developed based on supervision information mainly focus on the similarity information of heterogeneity without considering the homogeneous information, but it is obvious that the within-modal similarity benefits to capture the intrinsic geometric structure. On the other hand, the methods generated by emphasising within-modal similarity decompose multi-modal data into a set of uni-modal data, which means multi-modal similarity learning cannot be treated as a whole because more than one manifold are needed to represent both cross-modal and within-modal similarities. Therefore, it is necessary to **connect** and **integrate** all information from data in different modalities to describe the diversity of the world. To achieve this, the key of cross-modal search is to overcome the obstacle of multiple modalities by considering both the local geometric and global supervision information.

In this chapter, by integrating the supervision information and the local structure of heterogeneous data, a novel method termed hetero-manifold regularisation (HMR) is proposed to learn hash functions for efficient cross-modal search. Three significant advantages are illustrated in the schematic diagram of a hetero-manifold shown in Fig. 6.2. Firstly, a hetero-manifold well describes the local information by representing homogeneous data on the sub-manifolds. In Fig. 6.2, the data in three different modalities are represented by three sub-manifolds which well model the relationship between homogeneous data. Secondly, the hetero-manifold emphasises the global information of multi-modal data as well, by modelling the *information propagation* across modalities with three-order random walks. It is clear in Fig. 6.2 that any pair of points could be connected via two steps on homogeneous sub-manifolds and one step crossing two different sub-manifolds. Thus, the samples across modalities could be compared by integrating the information from all related homogeneous sub-manifolds. Lastly, the hetero-manifold is flexible and can be extended to model any number of modalities. As far as we know, existing cross-modal searching algorithms are limited to only two modalities.

Given a training set, the inherent similarity of multiple modalities on the hetero-manifold is represented by the hetero-Laplacian matrix. Thus, by minimising the regularisation item via the graph hetero-Laplacian, a set of cross-modal hash functions which are smooth on the hetero-graph can be learned to embed original data points into a Hamming space. In other words, the learned hash functions will preserve the geometrical structure and global supervision information of the hetero-manifold. Meanwhile, a novel weighted cumulative distance inequality on hetero-graph is introduced to cross the gap between Hamming distance and Euclidean distance. By using this novel distance inequality, the problem of learning hash functions is transformed into training a hetero-manifold regularised support vector machine.

6.2 Hetero-manifold of multi-modal data

Let $\mathcal{O} = \{O_1, O_2, \cdots, O_N\}$ be a set containing N objects. For the uth modality, $\mathcal{O}$ is recorded as a $d_u \times N$ matrix X^u where the ith column vector of X^u, x_i^u corresponds to O_i, $1 \leq u \leq M$, M is the number of modalities, and d_u is the dimension of x_i^u. Generally, the number of modalities is larger than 2, i.e., $M \geq 2$.

A hetero-manifold is an ensemble of uni- and cross-modal sub-manifolds. Uni-modal sub-manifolds are the manifolds whose elements corresponding to different objects share a common modality. For example, X^u is a dataset in which all samples are on the uth uni-modal sub-manifold. It is clear that uni-modal sub-manifolds are used to represent the intra-structure of uni-modal data. In contrast, cross-modal sub-manifolds serve as bridges to connect different uni-modal data. Ideally, any pair of data points on different uni-modal sub-manifolds could be connected via a path on the cross-modal manifolds and the distance of the path could be used to represent the similarity between the cross-modal data.

Given training samples, the hetero-manifold could be represented as a hetero-

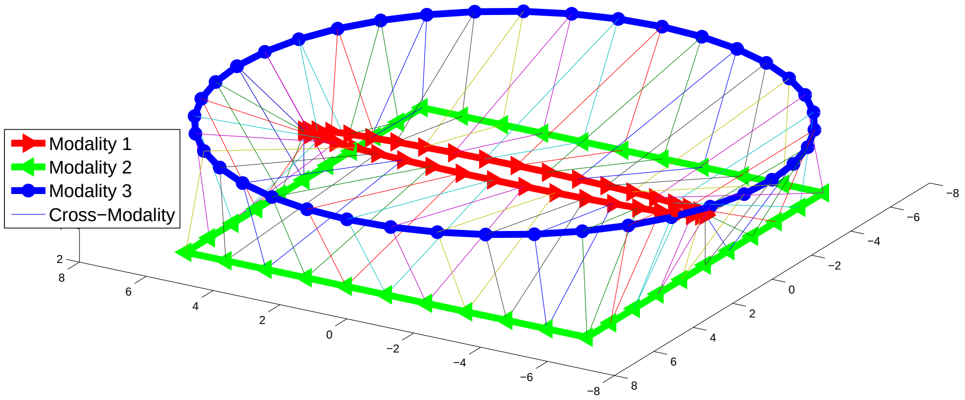


Figure 6.2: A hetero-manifold with three modalities: the blue, red and green closed curves represent three uni-modal data sub-manifolds; the lines used to connect two uni-modal data sub-manifolds constitute a cross-modal sub-manifold; all uni- and cross-modal sub-manifolds constitute a hetero-manifold; any change of a uni- or cross-modal sub-manifold will result in a change of the hetero-manifold.

graph $G = (V, S)$, where V is the set of vertices and S is the set of edges. In this chapter, V contains all feature matrices $X^1, X^2, \cdots, X^M$, and the edge between two vertices is defined as the similarity measurement between these two vertices. Following the idea of the hetero-manifold, a hetero-graph could be decomposed into a set of sub-graphs on the homogeneous sub-manifolds and a set of sub-graphs on the cross-modal sub-manifolds. Generally, both sub-graphes could be defined as follows:

Definition 1. ***Uni-modal sub-graph.*** *$G^{uu} = (V^{uu}, S^{uu})$ is a uni-modal sub-graph, if all vertices in this graph come from X^u.*

Definition 2. ***Cross-modal sub-graph.*** *$G^{uv} = (V^{uv}, S^{uv})$ is a cross-modal sub-graph, if, for each edge of this graph, one vertex comes from X^u and the other vertex comes from X^v.*

Definition 3. ***Hetero-graph.*** *$G = (V, S)$ is a hetero-graph, if, its vertices correspond to all multi-modal data $X^1, X^2, \cdots, X^M$, and the similarity matrix S satisfies*

$$S = \begin{pmatrix} S^{11} & S^{12} & \dots & S^{1M} \\ S^{21} & S^{22} & \dots & S^{2M} \\ \dots & \dots & \dots & \dots \\ S^{M1} & S^{M2} & \dots & S^{MM} \end{pmatrix}. \tag{6.1}$$

Three-order random walks on the hetero-graph is used to model the information diffusion among the vertices on the hetero-graph. For each pair of vertices x_i^u, x_j^v on the hetero-graph, the connection between them consists of three steps: from the end x_i^u to a possible neighbour of x_i^u, from the neighbour of x_i^u to the neighbour of x_j^v, and from the neighbour of x_j^v to the end x_j^v, just like the path shown in Fig. 6.3. On the one hand, for the first and third steps, the neighbours of the end must be represented in a common modality and the similarity between x_i^u and its neighbour $x_{i'}^u$ is generally measured by a Gaussian kernel, such as

$$s^u(i, i') = \exp\{-\frac{||x_i^u - x_{i'}^u||^2}{\sigma^2}\}, \tag{6.2}$$

where $\sigma \neq 0$ is a kernel parameter. Similarly, the similarity between x_j^v and its neighbour $x_{j'}^v$ is $s^v(j, j') = \exp\{-\frac{||x_j^v - x_{j'}^v||^2}{\sigma^2}\}$. On the other hand, for the second step, the similarity between $x_{i'}^u$ and $x_{j'}^v$ should be defined according to the different situations of their modalities. If $x_{i'}^u$ and $x_{j'}^v$ share a same modality, the similarity between them could be defined according to their neighborhood relationship, such as

$$\begin{aligned} & p(x_{i'}^u, x_{j'}^v) \\ = & \ p(x_{i'}^u, x_{j'}^u) \\ = & \begin{cases} 1, & s^u(i', j') \leq \delta, \\ 0, & s^u(i', j') > \delta, \end{cases} \end{aligned} \tag{6.3}$$

where $\delta \geq 1$ is a parameter for controlling the connection between two points on a uni-graph. $\delta \geq 1$ can be selected as the criterion in classical Laplacian Eigenmaps [68]. If $x_{i'}^u$ and $x_{j'}^v$ are represented in different modalities, the similarity between them should be defined according to the credible priori information. For example, the similarity between $x_{i'}^u$ and $x_{j'}^v$ could be set to be 1 if they correspond to a same object, and set to be 0 otherwise, that is

$$p(x_{i'}^u, x_{j'}^v) = \begin{cases} 1, & i' = j', \\ 0, & \text{otherwise.} \end{cases} \tag{6.4}$$

If the label information of multi-modal data is available, the similarity between $x_{i'}^u$ and $x_{j'}^v$ could be also defined according to the similarity of the labels of $x_{i'}^u$ and $x_{j'}^v$, such as

$$p(x_{i'}^u, x_{j'}^v) = \begin{cases} 1, & t_{i'} = t_{j'}, \\ 0, & \text{otherwise,} \end{cases} \tag{6.5}$$

where $t_{i'}, t_{j'}$ are respectively the labels of $x_{i'}^u, x_{j'}^v$. If the objects are described by multiple labels, the similarity between $x_{i'}^u$ and $x_{j'}^v$ could be also described as

$$p(x_{i'}^u, x_{j'}^v) = \frac{|t_{i'} \cap t_{j'}|_\#}{|t_{i'} \cup t_{j'}|_\#}, \tag{6.6}$$

where $t_{i'}, t_{j'}$ are the sets of multiple labels for describing objects $O_{i'}$ and $O_{j'}$, and $|\cdot|_\#$ is the size of a set. More meaningful priori depending on a particular task can be used here, such as semantic affinities and correlations.

Thus, all possible one-order similarities between the vertices on a uni- or cross-modal sub-graph could be respectively represented by three matrices $S^u = (s^u(i,i'))$, $P^{uv} = (p(x^u_{i'}, x^v_{j'}))$, and $S^v = (s^v(j,j'))$. According to these examples of $p(x^u_{i'}, x^v_{j'})$, for any u, v, we assume in this chapter that the priori matrix P^{uv} satisfies:

$$P^{uv} = (P^{vu})^T. \tag{6.7}$$

By combining these one-order similarities, the similarity information diffusion model could be defined by a three-order random walk as

$$S^{uv} = S^u P^{uv} S^v. \tag{6.8}$$

As a special case, the similarity matrix of a uni-modal sub-graph is $S^{uu} = S^u P^{uu} S^u$. The similarity matrix S^{uv} satisfies the following Lemmas.

Lemma 1. ***Non-negativity.*** *The elements of similarity matrix S^{uv} are non-negative.*

Lemma 2. ***Asymmetry.*** *In general, if two matrices S^{uu} and S^{vv} are unequal, S^{uv} is an asymmetric matrix.*

Lemma 3. ***Equivalence.*** *Any pair of similarity matrices S^{uv} and S^{vu} satisfies the relationship:*

$$S^{uv} = (S^{vu})^T. \tag{6.9}$$

Therefore, the similarity matrix S on the hetero-graph satisfies $S = S^T$. Lemma 1 is a result of the non-negativeness of Gaussian kernel (6.2) and the definition of $p(x^u_{i'}, x^v_{j'})$. Lemma 2 is the result of the definition of matrix multiplication. The proof of Lemma 3 can be found in Appendix 6.7.

Lemma 1 is the theoretical base of learning hash functions on a hetero-manifold. Lemma 2 unveils the intrinsic barrier of treating a multi-modal problem in a cross-modal view because of the asymmetry of both similarity matrices S^{uv} and S^{vu}. Lemma 3 hints the advantages of the global view to understanding multi-modal data as the hetero-manifold because of the symmetry of the similarity matrix on the hetero-manifold S. See Fig. 6.3 for more details.

6.3 Hash function learning on the hetero-manifold

A hetero-manifold integrates multi-modal data into a common manifold, however, a huge gap still exists for efficient cross-modal retrieval because of the difference

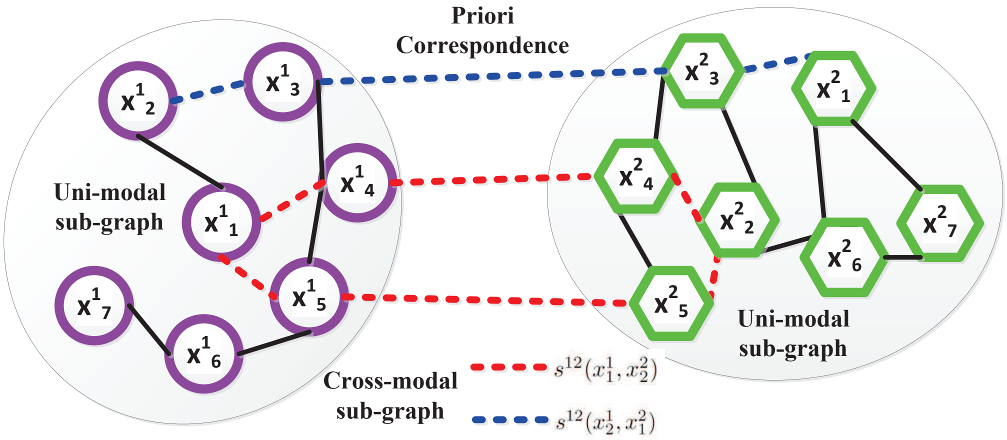


Figure 6.3: Cross-modal similarities between features of two objects O_i and O_j captured in two modalities. The lines represent the similarity between two points. The longer the lines, the less similar the two points are. The black lines represent the uni-modal similarity while the dashed lines represent the similarity defined by three-order random walks from one modality to another modality. Among them, we can see that the features x_1^1 and x_2^2 are connected by two red dashed lines whilst the two features x_2^1 and x_1^2 are connected by only one dashed blue line. This point reflects the asymmetry of S^{uv} in Lemma 2.

between different modalities. To this end, a framework of hetero-manifold regularised hash function learning is introduced to embed multi-modal data into a common Hamming space and simultaneously preserve the cross-modal and within-modal similarities on the hetero-manifold.

For the u-th uni-modal data X^u, a set of functions $\mathcal{F}^u = \{f_k^u, 1 \leq k \leq K\}$ is used to generate the hash codes of X^u, where K is the length of codes. Using these functions $\mathcal{F}^u$, for each sample x_i^u, a vector of real values[1] $F(x_i^u) = (f_1^u(x_i^u), f_2^u(x_i^u), \cdots, f_K^u(x_i^u))^T \in R^K$ can be obtained. Then, a binary code vector y_i^u of x_i^u can be learned by using $y_i^u = (F(x_i^u))_+$, where $(\cdot)_+$ is an operator which sets all positive numbers to 1 and other numbers to 0. Specifically, we have the k-th element of y_i^u:

$$y_i^u(k) = (f_k^u(x_i^u))_+. \tag{6.10}$$

[1] For simplicity, $F(x_i^u) = F^u(x_i^u)$ without confusion.

6.3.1 Distance inequality on a graph

In general, learning to hash tries to minimise a cumulative Hamming distance with some constraints. If the distance is defined on a manifold, then a weighted cumulative Hamming distance $\mathcal{L}_c^h(G)$ should be minimised.

$$\mathcal{L}_c^h(G) = \sum_{u,v=1}^{M} \sum_{i,j=1}^{N} s^{uv}(x_i^u, x_j^v) \mathcal{D}_h(y_i^u, y_j^v), \tag{6.11}$$

where $\mathcal{D}_h(y_i^u, y_j^v)$ is the Hamming distance between y_i^u and y_j^v. Actually, the weights between the samples embody the intrinsic structures and useful information including local neighbourhood, prior semantic cues and affinities. By considering these weights, the original structure and information can be preserved in a new learned space. In this paper, the weights reflect the information contained in the hetero-manifold. Meanwhile, besides the Hamming distance, for any pair of points x_i^u and x_j^v on graph G in the learned space, an accompanied Euclidean distance[1] can be defined as $\mathcal{D}_e(F(x_i^u), F(x_j^v)) = ||F(x_i^u) - F(x_j^v)||_2^2$. Same as Hamming distance, a weighted cumulative Euclidean distance on graph (G, S) is given as:

$$\begin{aligned} &\mathcal{L}_c^e(G) \\ &= \sum_{u,v=1}^{M} \sum_{i,j=1}^{N} s^{uv}(x_i^u, x_j^v) \mathcal{D}_e(F(x_i^u), F(x_j^v)). \end{aligned} \tag{6.12}$$

Normally, during the matching stage, the Hamming distance is far less computationally expensive than the Euclidean distance. However, despite the simplicity in Eq. 6.11, minimisation of the Hamming distance is generally intractable, because it is a concrete quantity. Thus, we seek to minimise an alternative item, which guarantees that the Hamming distance will be minimised simultaneously.

First, a constraint will be given as follows:

Definition 4. ***Hinge loss constraint.*** *For a function f_k^u in the uth modality, if any point x_i^u captured in this modality and its corresponding hash code defined in Eq. 6.10 satisfies*

$$y_i^u(k) f_k^u(x_i^u) \geq 1 - \xi_{ik}^u, \tag{6.13}$$

where ξ_{ik}^u is a minimal non-negative value, thus f_k^u is the hinge loss constraint-satisfied function in the uth modality.

Next, under the above constraint, a distance inequality in the following can be obtained:

[1] For simplicity, $F(x_i^u) = F^u(x_i^u)$ without confusion.

Lemma 4. ***Distance inequality.*** *If two sets of functions $\mathcal{F}^u$ and $\mathcal{F}^v$ are the hinge loss constraint-satisfied functions in modalities u and v respectively, for any two samples x_i^u and x_j^v, the two types of distance in the learned Hamming space and the Euclidean space have the following relationship, when satisfying $\forall k, \xi_{ik}^u + \xi_{jk}^v \leq 1$:*

$$\mathcal{D}_h(y_i^u, y_j^v) \leq \mathcal{D}_e(F(x_i^u), F(x_j^v)), \tag{6.14}$$

$\mathcal{D}_h$ and $\mathcal{D}_e$ are defined in Eq. 6.11 and 6.12, respectively.

It is worth to point out that f_k^u is a hinge loss constraint-satisfied function only when all the samples in modality u satisfy condition 6.13. And Eq. 6.14 can be proved, when a condition $\forall k, \xi_{ik}^u + \xi_{jk}^v \leq 1$ is given. We can see that ξ_{ik}^u and ξ_{jk}^v are two minimal non-negative values in the definition of the hinge loss constraint. If the two modalities are the same ($u = v$), the same inequality can be established for any two samples captured in the same modality.

Then, based on the condition 6.13, we can extend the inequality 6.14 to a weighted cumulative distance inequality on a graph.

Corollary 1. ***Weighted distance inequality.*** *For a graph $G = (V, S)$, if two sets of functions $\mathcal{F}^u$ and $\mathcal{F}^v$ satisfy the condition in Eq. 6.13, thus the following weighted cumulative distance inequality can be established, when S is a similarity matrix with non-negative members:*

$$\mathcal{L}_c^h(G) \leq \mathcal{L}_c^e(G). \tag{6.15}$$

Consequently, with the help of the inequality in the Corollary 1, a relaxed optimisation problem which will be introduced in the following section can be generated. In this chapter, we will consider to learn two sets of linear hash functions W^u and W^v via minimising the upper bound $\mathcal{L}_c^e(G)$ of the cumulative Hamming distance $\mathcal{L}_c^h(G)$. Corollary 1 is a direct result of Lemma 4. More proof details of Lemma 4 are provided in the Appendix 6.7.

6.3.2 Objective function

Specifically, the binary codes of x_i^u are defined by linear functions as $y_i^u = (((w_1^u)^T x_i^u)_+, ((w_2^u)^T x_i^u)_+, \cdots, ((w_K^u)^T x_i^u)_+)^T = ((W^u)^T x_i^u)_+$, where W^u is a matrix whose k-th column vector is w_k^u. Then, for the u-th uni-modal dataset X^u, the corresponding binary code set is $Y^u = ((W^u)^T X^u)_+$, in which the i-th column y_i^u is the binary code vector of x_i^u.

Furthermore, denote projection matrix

$$\mathbf{W}^T = ((W^1)^T, (W^2)^T, \cdots, (W^M)^T), \tag{6.16}$$

and multi-modal data matrix

$$\mathbf{X} = \begin{pmatrix} X^1 & 0 & \cdots & 0 \\ 0 & X^2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & X^M \end{pmatrix}. \tag{6.17}$$

Thus, the binary codes can be obtained:

$$\mathbf{Y} = (\mathbf{W}^T\mathbf{X})_+. \tag{6.18}$$

Using $Y^u = ((W^u)^T X^u)_+$, it is easy to prove that $\mathbf{Y} = (Y^1, \cdots, Y^u, \cdots, Y^M)$. Meanwhile, using Eq. 6.16 and 6.17, the cumulative Euclidean distance $\mathcal{L}_c^e(\mathbf{G})$ can be rewritten as

$$\mathcal{L}_c^e(G) = 2\mathbf{tr}(\mathbf{W}^T\mathbf{XLX}^T\mathbf{W}), \tag{6.19}$$

where Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{S}$, $\mathbf{D} = \mathbf{diag}(d_{11}, d_{12}, \cdots, d_{ui}, \cdots, d_{MN})$ and $d_{ui} = \sum_{v,j} \mathbf{S}(x_i^u, x_j^v)$. In this paper, **diag** is an operator to generate a diagonal matrix. The detailed proof of Eq. 6.19 is given in Appendix 6.7.

With the hinge loss constraint, the problem of hash function learning on hetero-manifold (6.11) could be approximated by minimising its upper bound (6.19) with some constraint conditions:

$$\begin{aligned} \mathbf{W}^* \quad & = \arg\min_{\mathbf{W}} \frac{1}{2}\mathbf{tr}(\mathbf{W}^T\mathbf{XLX}^T\mathbf{W}) \\ s.t. \quad & \forall u, i, k \\ & (i)\ y_i^u(k)(w_k^u)^T x_i^u \geq 1 - \xi_{ik}^u,\ \xi_{ik}^u \geq 0, \\ & (ii)\ \xi_{ik}^u + \xi_{jk}^u \leq 1, \\ & (iii)\ \mathbf{W}^T\mathbf{W} = I, \end{aligned} \tag{6.20}$$

where ξ_{ik}^u is a slack variable, y_i^u is the hash coding vector corresponding to the u-th modal data of the i-th object, and $y_i^u(k)$ is the k-th element of y_i^u. The first and second constraint conditions which are from Lemma 4 ensure Euclidean distance based loss $\mathcal{L}_c^e(G)$ be the upper bound of the Hamming distance based loss $\mathcal{L}_c^h(G)$. The third constraint condition corresponds to the requirement of orthogonality between two hash functions.

To further simplify the optimisation problem (6.20), the last two constraint conditions are slightly relaxed and transferred into the objective function by using the Lagrangian principle. As for constraint condition (ii), the total number of pairs ξ_{ik}^u, ξ_{jk}^u is $\frac{M^2N^2K}{2}$ because of the structure of the hetero-graph, and each ξ_{ik}^u

exists in MN constraint conditions. Thus all of these constraint conditions can be summed up and the conditions will be relaxed as

$$\sum_{u=1}^{M}\sum_{i=1}^{N}\sum_{k=1}^{K}\xi_{ik}^{u} \leq \frac{MNK}{2}. \tag{6.21}$$

Therefore, the original optimisation problem (6.20) is transformed by replacing the constraint conditions (ii) with the relaxed constraint conditions (6.21) and using the Lagrangian principle into

$$\begin{aligned} \mathbf{W}^{*} \quad &= \arg\min_{\mathbf{W}} \frac{1}{2}\mathbf{tr}(\mathbf{W}^{T}\mathbf{XLX}^{T}\mathbf{W}) \\ &+C_1 \sum_{u=1}^{M}\sum_{i=1}^{N}\sum_{k=1}^{K}\xi_{ik}^{u} \\ s.t. \quad &\forall u, i, k \\ &(i)\ y_i^u(k)(w_k^u)^T x_i^u \geq 1 - \xi_{ik}^{u},\ \xi_{ik}^{u} \geq 0 \\ &(ii)\ \mathbf{W}^T\mathbf{W} = I, \end{aligned} \tag{6.22}$$

where $C_1 > 0$ is the regularisation parameter.

It should be noticed that the Laplacian matrix $\mathbf{L}$ depends on all uni- and cross-modal similarity matrices because any sole sub-matrix used to define the similarity matrix S, for example S^{uv}, is not enough for defining the counterpart sub-matrix of $\mathbf{L}$. It implies that the Laplacian matrix contains the global information of the hetero-manifold. Therefore, the optimisation problem (6.22) is a hetero-manifold regularised hash function learning problem.

6.4 Sequential optimisation

In order to solve the problem in Eq. 6.22, we first divide it into sub-problems, in each of which only one projection for the k-th code is considered. Thus, in Eq. 6.18, the k-th row vector $\mathbf{y}_k$ of $\mathbf{Y}$ is a binary vector which corresponds the k-th bits of all samples in all modalities while the corresponding k-th column vector of $\mathbf{W}$ is denoted as $\mathbf{w}_k$. Then, we have

$$\mathbf{y}_k = (\mathbf{w}_k^T\mathbf{X})_+, \tag{6.23}$$

where the vector $\mathbf{w}_k^T = ((w_k^1)^T, (w_k^2)^T, \cdots, (w_k^M)^T)$.

Although these sub-problems are not independent with each other, they are convex when all the other variables are fixed. The convexity will be reflected by the standard quadratic programming problems in the following Eq. 6.25 and

6.27. Hence, the optimisation problem (6.22) could be resolved bit by bit in a sequential way. A similar work of sequential learning could be found in [233], when the sub-problems can be solved by a direct eigen-decomposition. In this paper, more specifically, the local optimal solution $\mathbf{W}^*$ is learned by sequentially optimising each of its column vectors $\mathbf{w}_k^*, k = 1, 2, \cdots, K$. For distinguishing the iterations of optimisation, the τ-th $\mathbf{W}^*$ and $\mathbf{w}_k^*$ are denoted as $\mathbf{W}^{(\tau)}$ and $\mathbf{w}_k^{(\tau)}$, respectively. In round τ, before solving the sub-problem, the binary codes $\mathbf{y}_k^{(\tau-1)}$ should be initiated using codes in the last round or generated randomly.

6.4.1 The first hash function learning

Firstly, the first set of linear projections $\mathbf{W}^*$ is learned by sequentially optimising each of its column vectors $\mathbf{w}_k^*, k = 1, 2, \cdots, K$. For distinguishing the iterations of optimisation, the τ-th $\mathbf{W}^*$ is denoted as $\mathbf{W}^{(\tau)}$, and the corresponding column vectors of $\mathbf{W}^{(\tau)}$ are denoted as $\mathbf{w}_k^{(\tau)}, k = 1, 2, \cdots, K$. Therefore, we have

$$\mathbf{Y}_k^{(\tau)} = (\mathbf{w}_k^{(\tau)})^T \mathbf{X}. \tag{6.24}$$

To train the hash functions, the hash codes $Y^{(0)}$ will be randomly initialised in the first round when $\tau = 1$. Then, $\mathbf{w}_1^{(1)}$ could be learned from the optimisation problem

$$\begin{aligned} \mathbf{w}_1^{(1)} \quad &= \arg\min_{\mathbf{w}_1} \frac{1}{2} \mathbf{w}_1^T \mathbf{X} \mathbf{L} \mathbf{X}^T \mathbf{w}_1 \\ &+ C_1 \sum_{u=1}^{M} \sum_{i=1}^{N} \xi_{i1}^u \\ s.t. \quad & \forall u, i, \\ & (i)\ y_i^u(1)(w_1^u)^T x_i^u \geq 1 - \xi_{i1}^u, \\ & (ii)\ \xi_{i1}^u \geq 0. \end{aligned} \tag{6.25}$$

The optimisation problem (6.25) is derived from the problem (6.22) where the orthogonal constraint condition becomes zero because it is assumed that $\mathbf{w}_1^{(1)}$ is orthogonal with the other projection directions $\mathbf{w}_k^{(1)}, k = 2, 3, \cdots, K$ without any information about $\mathbf{w}_k^{(1)}, k = 2, 3, \cdots, K$.

It is clear that the optimisation problem (6.25) is convex. Meanwhile, the Lagrange dual of the optimisation problem (6.25) is a problem of quadratic programming. Therefore, the optimal $\mathbf{w}_1^{(1)}$ could be defined as

$$\mathbf{w}_1^{(1)} = (\mathbf{X} \mathbf{L} \mathbf{X}^T)^{-1} \mathbf{X}_Y^{(0)} \alpha_1^{(1)}, \tag{6.26}$$

where $\mathbf{X}_{Y_1}^{(0)} = \mathbf{diag}(X_{Y_1}^1, \cdots, X_{Y_1}^M)$[1], the matrix $X_{Y_1}^u = (y_1^u(1)x_1^u, \cdots, y_N^u(1)x_N^u)$,

[1] Without confusion, the subscript $\mathbf{X}_{\mathbf{Y}_1^{(0)}}$ will be simplified as $\mathbf{X}_{Y_1}^{(0)}$.

and $\alpha_1^{(1)}$ is the result of the Lagrange dual problem of (6.25). $y_i^u(1)$ is the initial bit which is from $\mathbf{Y}_1^{(0)}$ for object O_i in the uth modality.

6.4.2 The following hash function learning

Given $\mathbf{w}_1^{(1)}, \mathbf{w}_2^{(1)}, \cdots, \mathbf{w}_{k-1}^{(1)}$, the next optimal projection $\mathbf{w}_k^{(1)}$ could be defined via the following optimisation problem

$$\begin{aligned} \mathbf{w}_k^{(1)} &= \arg\min_{\mathbf{w}_k} \frac{1}{2}\mathbf{w}_k^T \mathbf{XLX}^T \mathbf{w}_k \\ &+ \frac{C_2}{2}\mathbf{w}_k^T \mathbf{Q}_k^{(1)} \mathbf{w}_k + C_1 \sum_{u=1}^{M}\sum_{i=1}^{N} \xi_{i,k}^u \\ s.t. \quad & \forall u, i, \\ & (i) \;\; y_i^u(k)(w_k^u)^T x_i^u \geq 1 - \xi_{i,k}^u, \\ & (ii) \; \xi_{i,k}^u \geq 0, \end{aligned} \tag{6.27}$$

where $C_2 > 0$ is a regularisation parameter, and $\mathbf{Q}_k^{(1)} = \sum_{l=1}^{k-1} \mathbf{w}_l \mathbf{w}_l^T$ which is used to measure the orthogonality between $\mathbf{w}_k$ and the other learned $\mathbf{w}_l$, $l = 1, 2, \cdots, k-1$. It is clear that

$$\mathbf{w}_k^T \mathbf{Q}_k^{(1)} \mathbf{w}_k = \sum_{l=1}^{k-1} (\mathbf{w}_k^T \mathbf{w}_l)^2, \tag{6.28}$$

where $\mathbf{w}_k^T \mathbf{w}_l$ defines the linear correlation between $\mathbf{w}_k$ and $\mathbf{w}_l$. By minimising the term $\mathbf{w}_k^T \mathbf{Q}_k^{(1)} \mathbf{w}_k$, the learned projection direction $\mathbf{w}_k^{(1)}$ will be approximatively orthogonal to all of the other learned projection directions. Similar to formula (6.26), the optimisation problem (6.27) could also be resolved by using the Lagrange dual method

$$\mathbf{w}_k^{(1)} = (\mathbf{XLX}^T + C_2 \mathbf{Q}_k^{(1)})^{-1} \mathbf{X}_{Y_k}^{(0)} \alpha_k^{(1)}, \tag{6.29}$$

where $\alpha_k^{(1)}$ is the result of Lagrange dual of optimisation problem (6.27) and $\mathbf{X}_{Y_k}^{(0)}$ will be updated according to the binary vector $\mathbf{Y}_k^{(0)}$.

When $\mathbf{W}^{(1)}$ is learned according to the formulas (6.25) and (6.29), the following $\mathbf{W}^{(\tau)}, \tau = 2, 3, \cdots, t$ could be learned by using a similar objective function. The differences to problem (6.25) are the definition of the orthogonal item:

$$\mathbf{Q}_k^{(\tau)} = \sum_{l \neq k} \mathbf{w}_l^{(\tau-1)} (\mathbf{w}_l^{(\tau-1)})^T - \mathbf{w}_k^{(\tau-1)} (\mathbf{w}_k^{(\tau-1)})^T,$$

and, according to the bits learned in the last round $\mathbf{Y}_k^{(\tau-1)}$, the quantity $\mathbf{X}_{Y_k}^{(\tau-1)}$ should be also updated. Similarly, the optimal result $\mathbf{w}_k^{(\tau)}$ could be represented as

$$\mathbf{w}_k^{(\tau)} = (\mathbf{XLX}^T + C_2\mathbf{Q}_k^{(\tau)})^{-1}\mathbf{X}_{Y_k}^{(\tau-1)}\alpha_k^{(\tau)}. \tag{6.30}$$

The objective functions in Eq. 6.25 and 6.27 can be considered as a general dual problem[1], when we define $H = \mathbf{XLX}^T + C_2\mathbf{Q}^{(1)}$. Thus, the optimal solution can be obtained by a Representation Theory in Appendix 6.7. Therefore, all of these steps of optimising the original optimisation problem (6.22) can be summarised in Algorithm 5.

Algorithm 5 Hetero-manifold Regularised Hashing (HMR)

```
Input: Dataset $\{X^1, \cdots, X^M\}$, parameters $C_1, C_2$, the number of iterations $t$ and the length
  of hash coding vector $K$.
Output: $\mathbf{W}^t$.
Initialisation
(0) Construct matrix $S$ according to Eqs. (6.2), (6.8), and (6.1).
(1) Construct Laplacian graph $\mathbf{L}$ according to Eq. (6.19).
(2) Randomly initiate the binary codes $\mathbf{Y}^{(0)}$ and calculate $\mathbf{X}_{Y_1}^{(0)}$.
(3) Generate the first projection $\mathbf{w}_1^{(1)}$ according to Eq. (6.26).
For $k = 2, \cdots, K$
  (4) Randomly initiate the binary codes $\mathbf{y}_k^{(0)}$.
  (5) Calculate $\mathbf{Q}_k^{(1)}$ and $\mathbf{X}_{Y_k}^{(0)}$.
  (6) Generate $\mathbf{w}_k^{(1)}$ according to Eq. (6.29).
  (7) Update $\mathbf{Y}_k^{(1)}$ using Eq. 6.24.
End
For $\tau = 2, \cdots, t$
  For $k = 1, \cdots, K$
    (8) Calculate $\mathbf{Q}_k^{(\tau)}$ and $\mathbf{X}_{Y_k}^{(\tau-1)}$.
    (9) Generate the $k$-th projection $\mathbf{w}_k^{(\tau)}$ according to Eq. (6.30).
    (10) Update $\mathbf{W}^{(\tau)}$ and $\mathbf{Y}_k^{\tau}$ using Eq. 6.24.
  End
End
Return
```

6.5 Experiments

The proposed HMR is validated on four recent public datasets: the VIPeR [23] and CUHK01 [166] datasets for cross-camera person re-identification, the Wiki

[1] In the case of Eq. 6.25, the parameter can be set to $C_2 = 0$.

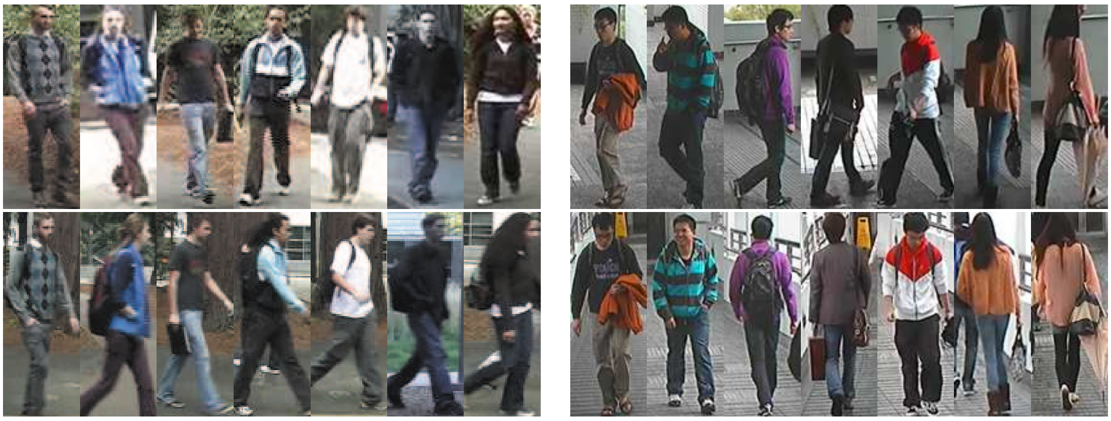


Figure 6.4: Some image examples of the two person re-identification datasets: VIPeR (left) and CUHK01 (right).

dataset [12] for cross-modal retrieval and the FG-NET ageing dataset [234] for cross-age face image retrieval where the number of modalities is 6. Four state-of-the-art cross-modal binary code learning methods, including PDH [24], CVH [125], CMSSH [126] and CMFH [128], are mainly compared with and some other area-specific methods are also used for comparative analysis in our experiments.

Evaluation Metrics: On the one hand, for identification systems, the Cumulated Matching Characteristics (CMC) [224] are commonly used for performance evaluation and measuring how well an identification system ranks the identities in the gallery with respect to a probe sample. Moreover, the Area Under Curve (AUC) corresponding to the CMC curves is also reported to show the overall performance at ranks from 1 to a fixed maximum. A larger AUC score means the corresponding method is more robust. On the other hand, for the ranking cases of multiple feedbacks, the precision and recall are normally calculated:

$$precision = \frac{|\Im \cap \Re|_{\#}}{|\Re|_{\#}}, \quad recall = \frac{|\Im \cap \Re|_{\#}}{|\Im|_{\#}},$$

where $\Re$ is a set of retrieved samples, $\Im$ is a set of relevant samples and $|\cdot|_{\#}$ denotes the size of the set. Precision-Recall (PR) curves [235] which are often used in information retrieval are used to measure performance in cross-modal retrieval. By varying the similarity measurement between the pair of retrieved samples (Hamming distance in this chapter) and evaluating the precision, recall and the number of retrieved points accordingly, PR curves can be obtained. Furthermore, Mean Average Precision (MAP) [128], which is the average precision at the ranks where recall changes, is generally used to evaluate a ranking system.

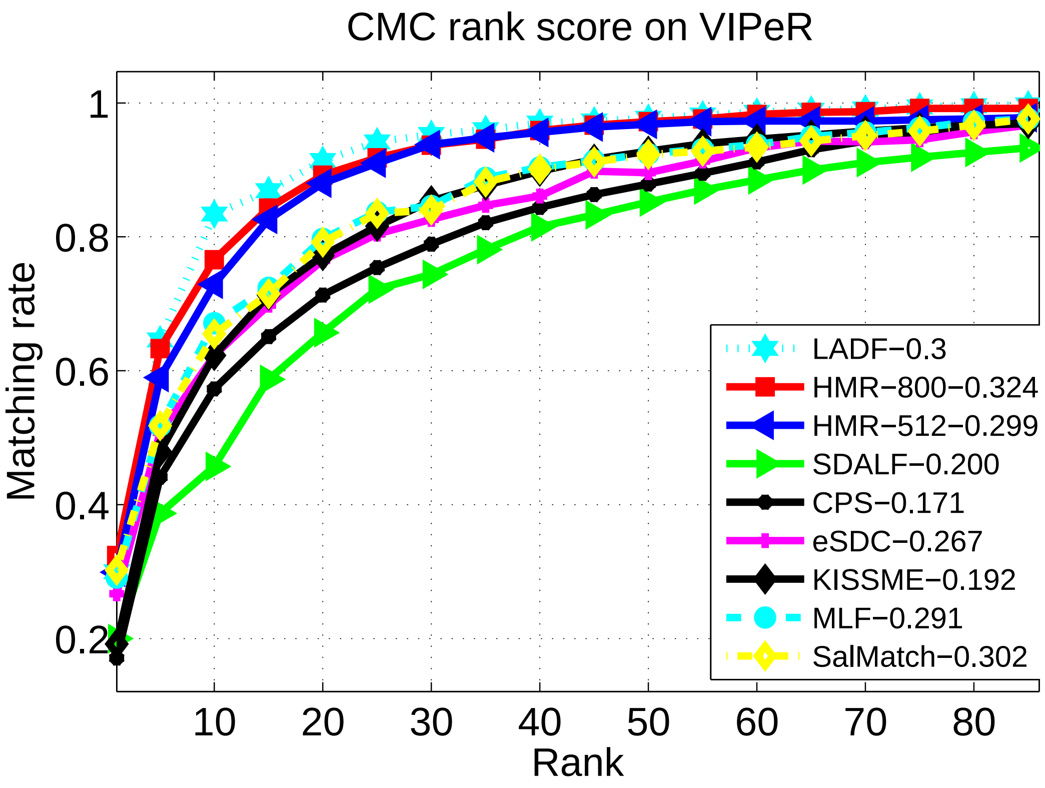


Figure 6.5: The CMC rankings of the compared methods on the VIPeR dataset with #316 test persons. Numbers in legend are the Rank-1 accuracies and HMR-512 means the length of learned codes of HMR is 512.

6.5.1 Cross-camera re-identification

Cross-camera person re-identification is a very challenging task because of the variation of camera views and the environment. Given a probe image containing a person, the most popular method of recognising the person is to rank the similarities between the probe image and the images in the gallery (captured by other cameras). In this experiment, the similarity is calculated in the learned Hamming space across the cameras and the maximum rank of AUC is 85.

VIPeR: This dataset contains 632 pedestrian image pairs in an outdoor environment. Each pair contains two images of the same individual taken from two different camera views. Changes of viewpoint, illumination and pose are the most significant causes of appearance change. Each image has been scaled to be 128 × 48 pixels. Some example images in VIPeR are shown in Fig. 6.4 (Left). The experimental setting is the same as [170]. Half of the dataset including 316 images for each view is used for training the algorithms and the remaining (316

Method	R1	R5	R10	R15	R20	AUC
HMR	**0.299**	**0.590**	**0.729**	**0.826**	**0.880**	**0.897**
CMFH	0.247	0.528	0.712	0.766	0.816	0.871
PDH	0.171	0.449	0.604	0.693	0.778	0.822
CMSSH	0.190	0.437	0.639	0.725	0.791	0.831
CCA	0.168	0.427	0.551	0.633	0.693	0.776
CVH	0.085	0.209	0.294	0.345	0.399	0.551

Table 6.1: Ranking accuracy comparison at ranks 1, 5, 10, 15 and 20 and overall AUC performance comparison when 512 dimensional binary codes are learned. R1 denotes Rank 1.

pedestrian) is used for testing.

CUHK01: Two cameras setting in different places of a campus environment are used to collect the samples. Camera A captures the frontal view or back view of pedestrians, while camera B captures the side view. This dataset contains 971 persons, each of which has two images. Some example images in CUHK01 are shown in Fig. 6.4 (Right). All the images are normalised to 160×60 for evaluations. The experimental setting is the same as [225] where 486 persons are chosen for testing and the remaining persons for training.

In this experiment, the Local Maximal Occurrence Feature (LOMO) which was proposed in [160] is used. The original dimension of the LOMO feature is 26960 and then is reduced to 70 as suggested by [160]. In this experiment, the parameters C_1 and C_2 of Algorithm 5 are set to 20 and 2, respectively. All the results are reported by averaging 10 runs.

To compare the performance with the state-of-the-art person re-identification methods, we evaluate the proposed HMR and the recently published algorithms on the VIPeR dataset including: SDALF [227], CPS [228], KISSME [171], eSDC [164], SalMatch [226] MLF [225] and LADF [236]. For the proposed HMR, two lengths of binary codes 512 and 800 have been learned and the experimental results corresponded to both code lengths are denoted as HMR-512 and HMR-800, respectively. The comparison results are shown in Fig. 6.5. Firstly, we can see that, except for LADF, HMR (HMR-512 and -800) significantly outperforms other methods and the advantages are more obvious especially at higher ranks (from 5 to 60). It is worth to point out that HMR is the only hashing-based method among the compared ones and still achieves comparative results to a non-hashing metric learning method LADF. In fact, due to quantisation loss, the performance of hashing methods is normally lower than that of non-hashing methods in many applications. Secondly, HMR-512 achieves similar results as HMR-800 and this demonstrates that the performance keeps stable when the code length is above a certain threshold. Finally, we also compare with other hashing methods on the VIPeR dataset when the binary code length is fixed at 512[1] and the comparison results are illustrated in Table 6.1. We can see that, both from the perspectives of ranks 1, 5, 10, 15 and 20 and the overall performance AUC, HMR achieves much better results than state-of-the-art hashing methods.

To further compare with other hashing methods, binary codes of shorter lengths (32, 64 and 128) are learned on the CUHK01 dataset. The results are shown in Fig. 6.6 and Table 6.2. We can observe that, as the code length increases, the performance of eigenvalue decomposition based methods such as CVH decreases since the first few projection directions occupy most of variances. However, it is reasonable that our HMR can achieve better when the code length increases. More information can be kept because HMR considers both the orthogonality and the cross-modal intrinsic structure. We can see that HMR achieves best results at all code lengths. Specifically, the advantages of HMR are more obvious, when the length of learned codes increases. The rank 1 scores of the five methods are also shown in the legend of Fig. 6.6 and HMR obtains at least 0.024 higher scores than other methods.

Method	CVH	CMFH	PDH	CMSSH	HMR
32 bits	45.66	65.39	58.96	54.21	**67.99**
64 bits	38.47	65.37	66.59	54.78	**69.36**
128 bits	30.13	67.03	69.29	55.15	**72.14**

Table 6.2: AUC Comparison on CUHK01 corresponding to the curves in Fig. 6.6.

6.5.2 Cross-modal retrieval

Images and texts are the two popular modalities for testing cross-modal retrieval methods. There are several datasets available but Wiki is the most popular one. Thus, in this experiment, the Wiki [12] dataset is used for our evaluations.

Wiki: It is generated from the "Wikipedia featured articles" and consists of 2866 image-text pairs in 10 most populated categories. The texts are represented by 10 dimensional latent Dirichlet allocation model and each image has a 128 dimensional SIFT histogram feature. We follow the data partition adopted in [12] to split the dataset into a training set of 2173 pairs and a test set of 693 pairs. In our setting, both gallery and query samples are from the test set which is different to the setting in [128]. In [128], the gallery samples are from the training set and thus their retrieval results are better than ours. If the query comes from the test set, then the samples in the text test set will be considered

[1] Because of the limitation of covariance, CVH and CCA cannot learn functions with a number exceeding the rank of the matrix. Thus, best results are reported at a certain length.

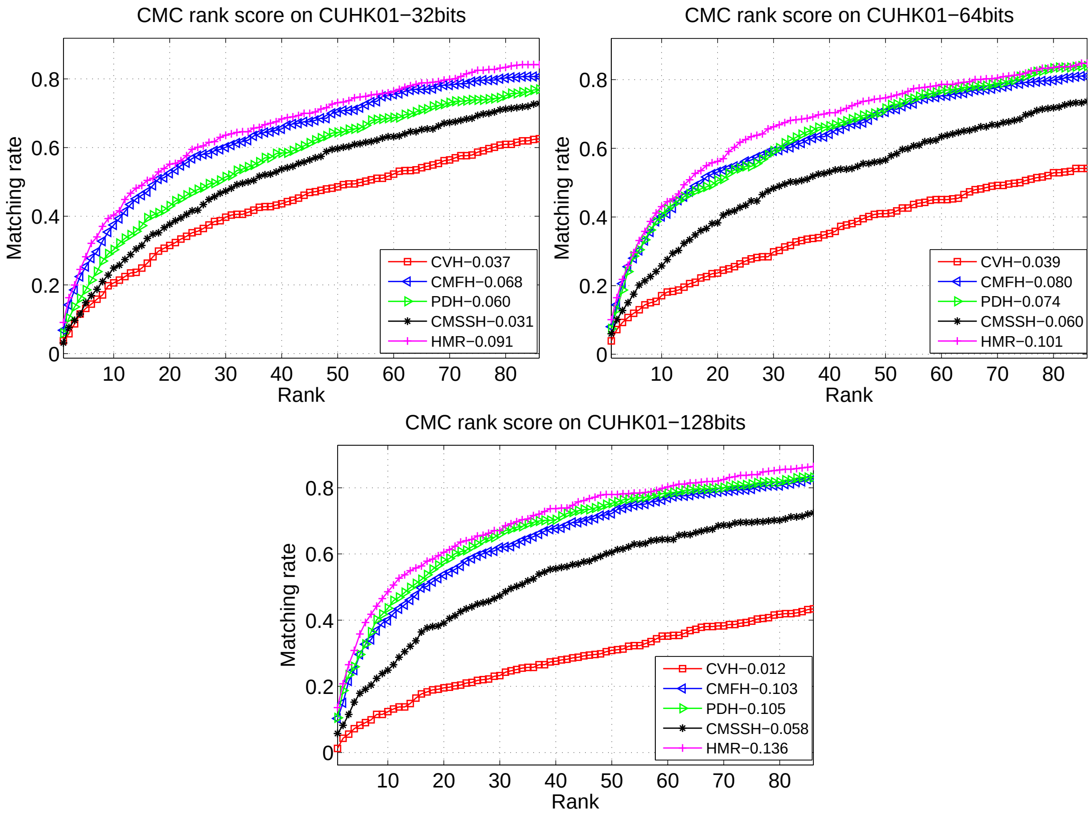


Figure 6.6: The CMC rankings of five methods on the CUHK01 dataset at code lengths 32, 64 and 128 with 486 test persons.

as the database and vice versa. In this experiment, the parameters C_1 and C_2 of Algorithm 5 are set as 30 and 1.2, respectively. The number of retrieved instances is set to $|\Re|_{\#} = 50$.

The MAP results on the test set are shown in Table 6.3. The same phenomenon of performance reduction as the code length increases for the eigenvalue decomposition based methods can be also observed on Wiki. From Table 6.3, we can see that HMR outperforms the state-of-the-art methods at code lengths 32 and 64, and achieves very close scores to the best method at code length 16. Moreover, the Precision-Recall (PR) curves on the Wiki dataset, which are obtained by varying the Hamming distance between the query points and the retrieved points, are reported in Fig. 6.7. HMR can obtain higher scores for almost all the Hamming radii from 1 to the maximum at code lengths 32 and 64 and get a similar PR curve to the best one at code length 16. Finally, MAP performance on each category is shown in Fig. 6.8. The retrieval difficulties of the 10 cate-

Task	Method	16 bits	32 bits	64 bits
Image Query	CVH	0.2021	0.1668	0.1723
	CMSSH	0.2276	0.1940	0.1982
	PDH	0.1885	0.1796	0.2086
	CMFH	**0.2583**	0.2567	0.2691
	HMR	0.2503	**0.2621**	**0.2833**
Text Query	CVH	0.2560	0.1902	0.2019
	CMSSH	0.2483	0.2431	0.2505
	PDH	0.2309	0.2278	0.2279
	CMFH	**0.3192**	0.3347	0.3351
	HMR	0.3151	**0.3408**	**0.3511**

Table 6.3: MAP Comparison on Wiki.

gories to the five methods are similar and three of them, i.e., Biology, Geography and Warfare, seem to be more easily classified. From Fig. 6.8, we can see that HMR is more robust on different categories over other methods. Very recently, deep neural networks were also exploited for multi-modal hashing [237] or cross-modal hashing [238] and achieved more advanced results than some other types of methods. However, the complexity of code generation in deep neural networks is generally much higher than that in linear functions. Take the model of layers $100 - 256 - 128 - 64 - 32 - 32$ in [237] for example, the number of multiplications is 68608 times of that in the corresponding linear function.

6.5.3 Cross-age face retrieval

In this section, we validate the proposed HMR on a more challenging task: cross-age face retrieval. Given a probe face image, we need to search for the face images of the same person but captured in different age stages. This task is derived from age estimation [13] but it is more difficult and novel because: 1) The principal characteristics of the face appearance of a same person vary hugely along with the variation of his or her age. 2) The capturing conditions of images are quite diverse in different places and years. 3) As far as we know, the cross-age face retrieval is the first multi-modal experiment, in which 6 modalities are considered. Intuitively, the ages of faces can be considered as modalities in our setting, in which faces of different persons with the same age range share similar characteristics including smoothness, wrinkles and hair.

FG-NET: Some examples of an ageing dataset [234], which contains 82 people with age ranges from 0 to 69, are shown in Fig. 6.9. The images of a same person distribute unevenly and most of the images are captured in the early ages. Thus, we divide the ages into 6 stages including $0-4$, $5-9$, $10-14$, $15-19$, $20-30$ and $31-69$ which correspond to 6 modalities in our method. In this experiment, the

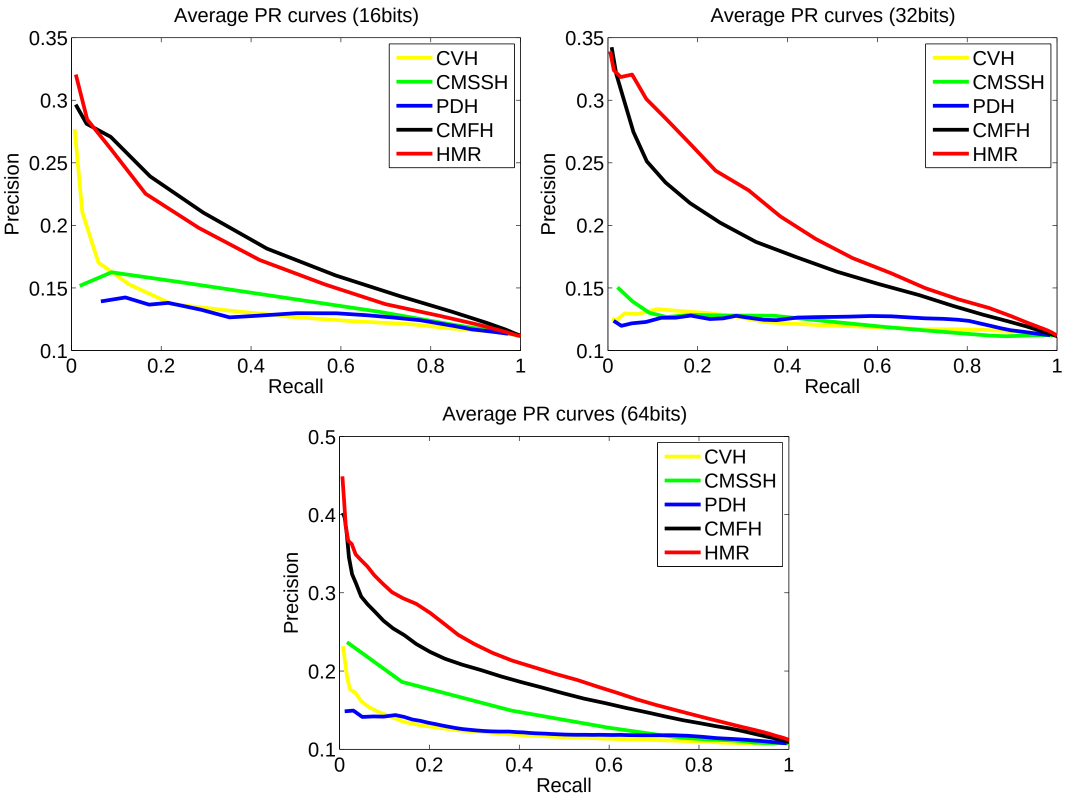


Figure 6.7: Precision recall curves on Wiki by varying the Hamming distance.

parameters C_1 and C_2 of Algorithm 5 are set to 10 and 0.1, respectively. 10-fold cross validation is used and, in each fold, 90% persons will be chosen as training and the remaining as for testing. In this experiment, the maximum value for AUC is set to 50.

Firstly, same as most age estimation works, features are directly extracted based on the 64 landmarks offered by the FG-NET dataset. For each landmark, a simple descriptor GIST [239] is used for representing a fixed rectangle (19×19) around it and then a feature for a face image can be constructed by concatenating the features of all landmarks. Principal Component Analysis (PCA) is adopted to reduce the feature into a space with 255 dimensions. Secondly, it is worth to point out that the number of images of a same person differs significantly for different age stages. Thus, compared to person re-identification and cross-modal retrieval, the task becomes more difficult because the correspondence matrix between two modalities is not diagonal. For some methods such as CMFH, the optimisation is not even technically correct. By duplicating the samples of a same person, a diagonal correspondence matrix can be obtained. Moreover, except for our HMR,

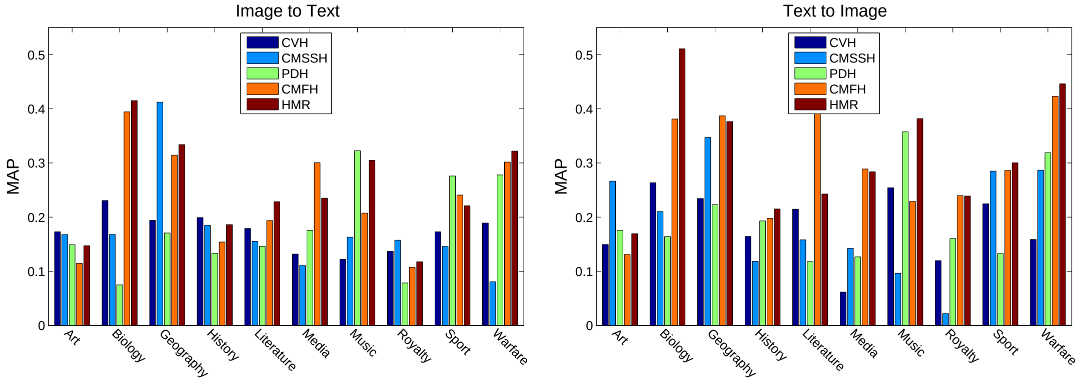


Figure 6.8: MAP performance for each category at 32 bits.

no existing methods can directly tackle multiple modalities with an inconsistent number of samples or features. To compare with these methods, any two modalities will be considered as the input of the two-modality methods. Taking the PDH, CMFH, CCA, CMSSH and CVH as examples, these methods will be trained 15 times for cross-age face retrieval and, for each modality, 5 different groups of projections will be obtained. This demonstrates that our proposed HMR is very powerful and flexible to deal with different tasks without particular limitations and the hash functions for different modalities can be obtained simultaneously by one optimisation.

Modalities	0-4	5-9	10-14	15-19	20-30	31-69
0-4	–	0.108	0.102	0.059	0.043	0.000
5-9	0.248	–	0.216	0.179	0.050	0.050
10-14	0.220	0.265	–	0.134	0.125	0.163
15-19	0.096	0.220	0.162	–	0.149	0.304
20-30	0.120	0.102	0.055	0.141	–	0.322
31-69	0.033	0.113	0.132	0.125	0.103	–

Table 6.4: Rank 1 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities.

The overall performance comparison of cross-age face retrieval is given in Fig. 6.10 and the different methods are ranked according to the Area Under Curve (AUC). From this figure, we can see that the proposed method consistently outperforms other methods at all ranks. Moreover, we can conclude that non-hashing method CCA achieves better results than other hashing-based methods. Furthermore, compared to the above experiments of two modalities, the advantages of the proposed HMR are more obvious in this experiment. The substantial reason

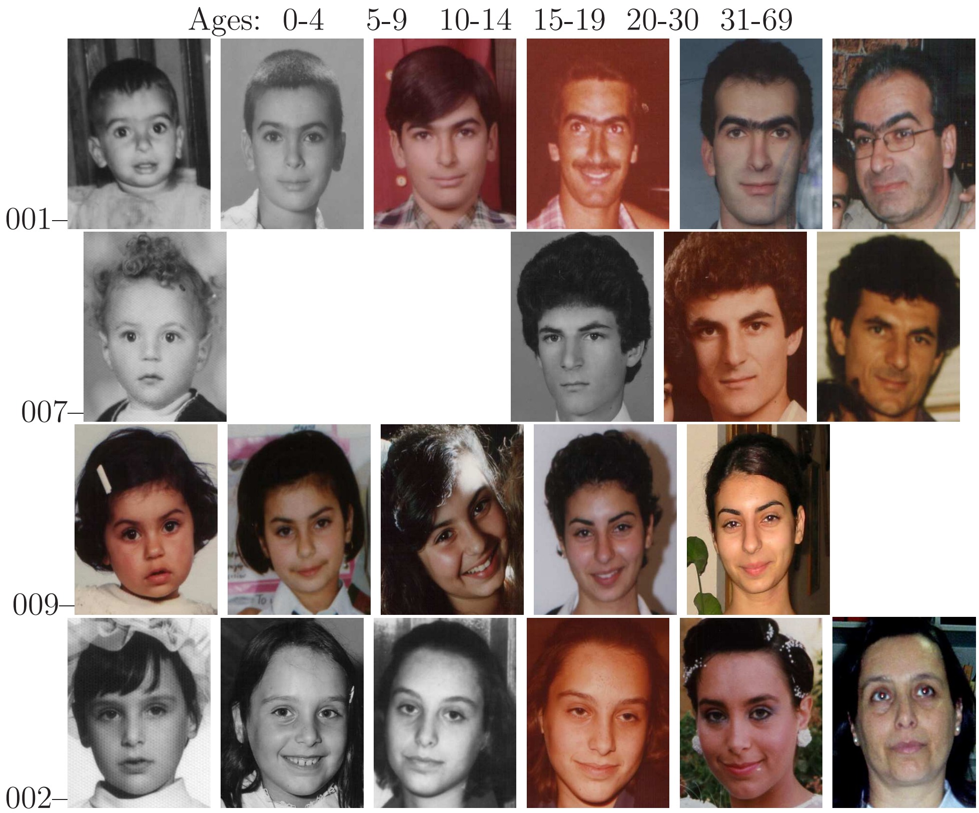


Figure 6.9: Some image examples of the FG-NET dataset. For person 007, the dataset contains no image samples with age range 5-14.

is that the information can be propagated on the proposed Hetero-manifold and then supervises the learning of hash functions. However, most state-of-the-art methods are specially designed for two modalities and, in the multi-modal cases ($M > 2$), to some extent, the global information is ignored.

To investigate the details of cross-age retrieval, the performance at ranks 1, 10 and 20 between any modalities is shown in Tables 6.4, 6.5 and 6.6, respectively. On the one hand, we can see that, in general, the performance of cross-age retrieval between two adjacent modalities is higher than that of non-adjacent modalities. In essence, the appearance changes between adjacent modalities will be smaller than those between large age gaps. On the other hand, it is interesting that the retrieval performance when the probe image comes from older age stages and the gallery consists of images from earlier ages normally will be better than the opposite conditions. We think this is because the appearance variation trend in the later age stages becomes smaller and some important identification characteristics remain as age increases.

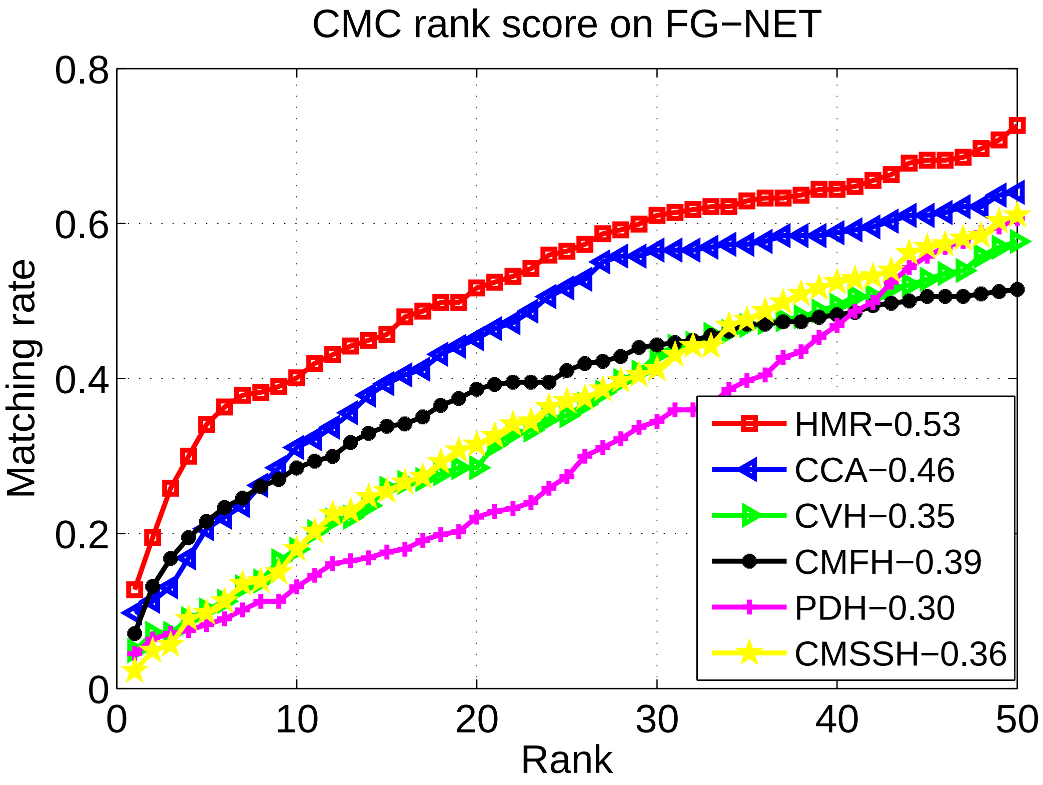


Figure 6.10: Overall performance comparison between the proposed HMR, CCA and other state-of-the-art methods. The number in the legend is the Area Under Curve (AUC) and the possible largest AUC can be up to 1.

Two probe samples with first 3 matches are shown in Fig. 6.11. The two persons have images from the $0-4$ modality to the $15-19$ modality. The left probe comes from the $5-9$ modality while the right one comes from the $0-4$ modality. We can see that several images with a same person have been successfully matched in different age stages by cross-age retrieval.

6.6 Summary

In this chapter, the concept of hetero-manifold was introduced for integrating the uni- and cross-modal similarities of multi-modal data in a global view. Both types of similarity are represented in the Laplacian matrix $\mathbf{L}$ corresponding to the hetero-manifold. The Laplacian matrix $\mathbf{L}$ appears smoothly when the Hamming distance in Eq. (6.11) is replaced by the Euclidean distance in Eq. (6.20), which hints that no hash functions could be learned without all uni- and cross-modal

Modalities	0-4	5-9	10-14	15-19	20-30	31-69
0-4	–	0.284	0.216	0.151	0.085	0.111
5-9	0.537	–	0.437	0.358	0.400	0.250
10-14	0.515	0.565	–	0.387	0.328	0.490
15-19	0.346	0.488	0.414	–	0.460	0.536
20-30	0.337	0.367	0.233	0.424	–	0.589
31-69	0.333	0.340	0.374	0.347	0.370	–

Table 6.5: Rank 10 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities.

Modalities	0-4	5-9	10-14	15-19	20-30	31-69
0-4	–	0.319	0.282	0.168	0.106	0.148
5-9	0.578	–	0.477	0.421	0.475	0.350
10-14	0.556	0.604	–	0.465	0.391	0.571
15-19	0.394	0.549	0.485	–	0.506	0.565
20-30	0.361	0.408	0.301	0.515	–	0.633
31-69	0.400	0.453	0.396	0.403	0.495	–

Table 6.6: Rank 20 performance of cross-age retrieval on the FG-NET face dataset with 6 modalities.

similarities being defined on the hetero-manifold. Therefore, the proposed framework of hetero-manifold regularised hash function learning (Eq. (6.20)) could benefit from the view of treating multi-modal data as a whole. The experimental results demonstrate that the proposed HMR outperforms the state-of-the-art methods on four popular datasets.

The hetero-manifold also offers some interesting problems in the field of cross-modal hashing. Firstly, it is interesting to consider a kernel extension of the proposed HMR. It is clear that the proposed hetero-manifold regularised framework (Eq. (6.20)) can be rewritten in Reproducing Kernel Hilbert Space (RKHS). By using RKHS, nonlinear hash functions could be learned, which may improve the performance of HMR. However, to achieve this, an induced problem needs to be considered for multi-modalities. For a common reproduced space or several individually reproduced spaces, which case is more reasonable? Moreover, what is the relationship between the reproduced spaces and the kernels? Secondly, it would be interesting to consider the proposed framework (Eq. (6.20)) in semi-supervised settings.

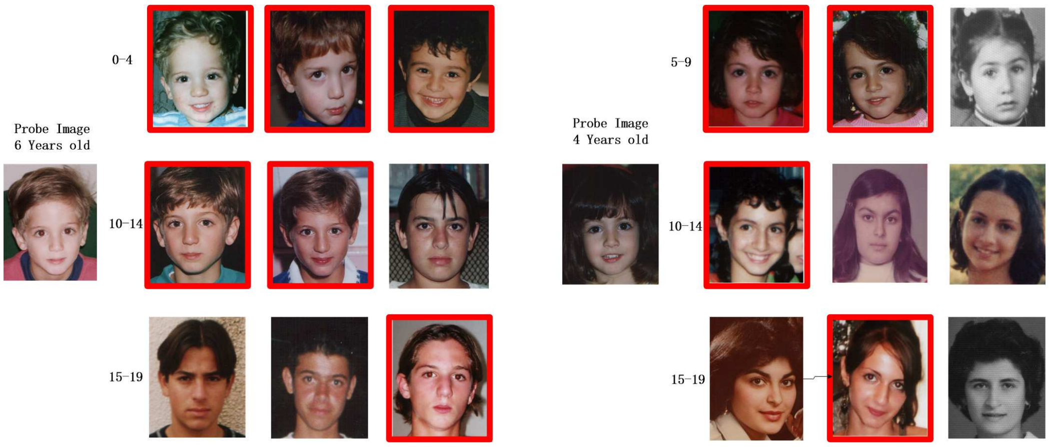


Figure 6.11: The cross-modal first three matching results of two probe images. The red rectangles demonstrate the correctly matched images in the gallery of a same person.

6.7 Appendices

Proof of Lemma 3

Proof. We have $S^{uv} = S^{uu}P^{uv}S^{vv}$ and $S^{vu} = S^{vv}P^{vu}S^{uu}$. The transposition of S^{vu} is:

$$\begin{aligned}(S^{vu})^T &= (S^{vv}P^{vu}S^{uu})^T \\ &= (S^{uu})^T(P^{vu})^T(S^{vv})^T \\ &= S^{uu}P^{uv}S^{vv} \\ &= S^{uv}.\end{aligned}$$

The third equation holds because matrices S^{uu}, S^{vv} and $P^{uv} = (P^{vu})^T$ are symmetric.

According to the definition of similarity matrix S, the symmetry of S could be proved by using the fact of $(S^{vu})^T = S^{uv}$. □

Proof of Lemma 4

Proof. The Hamming distance between two binary codes y_i^u and y_j^v is defined by:

$$\begin{aligned}\mathcal{D}_h(y_i^u, y_j^v) &= \sum_k y_i^u(k) \oplus y_j^v(k) \\ &= \sum_k \mathbf{1}((f_k^u(x_i^u))_+ \neq (f_k^v(x_j^v))_+),\end{aligned}$$

where $\mathbf{1}(\cdot)$ is an indicator function. Thus, for any k, we consider two conditions: (1) If $(f_k^u(x_i^u))_+ = (f_k^v(x_j^v))_+$, it is obvious that

$$y_i^u(k) \oplus y_j^v(k) = 0 \leq |f_k^u(x_i^u) - f_k^v(x_j^v)|.$$

(2) If $(f_k^u(x_i^u))_+ \neq (f_k^v(x_j^v))_+$, we assume that $(f_k^u(x_i^u))_+ = 1$ (Otherwise, same conclusion can be also obtained). There must be $(f_k^v(x_j^v))_+ = -1$. Since the two linear projections are both hinge loss constraint-satisfied functions, we have:

$$\begin{aligned} f_k^u(x_i^u) &\geq 1 - \xi_{ik}^u, \\ f_k^v(x_j^v) &\leq -1 + \xi_{jk}^v. \end{aligned}$$

So, there is $2 - \xi_{ik}^u - \xi_{jk}^v \leq |f_k^u(x_i^u) - f_k^v(x_j^v)|$. Provided that $\xi_{ik}^u + \xi_{jk}^v \leq 1$, the following inequality is true:

$$y_i^u(k) \oplus y_j^v(k) = 1 \leq 2 - \xi_{ik}^u - \xi_{jk}^v \leq |f_k^u(x_i^u) - f_k^v(x_j^v)|.$$

In total, we obtain the following conclusion by satisfying $\forall k, \xi_{ik}^u + \xi_{jk}^v \leq 1$:

$$\begin{aligned} \mathcal{D}_h(y_i^u, y_j^v) &= \sum_k y_i^u(k) \oplus y_j^v(k) \\ &= \sum_k \mathbf{1}((f_k^u(x_i^u))_+ \neq (f_k^v(x_j^v))_+), \\ &= \sum_k \mathbf{1}^2((f_k^u(x_i^u))_+ \neq (f_k^v(x_j^v))_+), \\ &\leq \sum_k (f_k^u(x_i^u) - f_k^v(x_j^v))^2. \end{aligned}$$

The third equation holds due to that $0^2 = 0$ and $1^2 = 1$. Therefore, we have:

$$\begin{aligned} \mathcal{D}_h(y_i^u, y_j^v) &\leq ||F(x_i^u) - F(x_j^v)||_2^2 \\ &= \mathcal{D}_e(F(x_i^u), F(x_j^v)). \end{aligned}$$

□

Proof of Equation (6.19)

Proof. According to the definition of $\mathbf{W}^T$ (6.16) and the definition of $\mathbf{X}$ (6.17), it is clear that

$$\begin{aligned} & \mathbf{W}^T\mathbf{X} \\ = \; & ((W^1)^T, \cdots, (W^M)^T) \begin{pmatrix} X^1 & 0 & \cdots & 0 \\ 0 & X^2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & X^M \end{pmatrix} \\ = \; & ((W^1)^T X^1, (W^2)^T X^2, \cdots, (W^M)^T X^M). \end{aligned} \tag{6.31}$$

Then

$$\begin{aligned}
& \mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W}) \\
= \ & \mathbf{tr}\Big(\big((W^u)^T X^u\big)_{u=1}^{M}\big(S^{uv}\big)_{u,v=1}^{M}\big((X^v)^T W^v\big)_{v=1}^{M}\Big) \\
= \ & \sum_{u,v} \mathbf{tr}((W^u)^T X^u S^{uv} (X^v)^T W^v) \qquad (6.32)
\end{aligned}$$

Notice the definition of $F(x_i^u)$ and $X^u = (x_1^u, \cdots, x_N^u)$, we have

$$\begin{aligned}
& \mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W}) \\
= \ & \sum_{u,v} \mathbf{tr}((F(x_i^u))_{i=1}^N S^{uv}((F(x_j^v))_{j=1}^N)^T) \\
= \ & \sum_{u,v}\sum_{i,j} S^{uv}(x_i^u, x_j^v)\langle F(x_i^u), F(x_j^v)\rangle_2. \qquad (6.33)
\end{aligned}$$

Meanwhile, we have the following equations

$$\begin{aligned}
& \mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{W}) = \sum_{u,i} d_{ui}||F(x_i^u)||_2^2 \\
= \ & \sum_{u,i} ||F(x_i^u)||_2^2 \sum_{v,j} \mathbf{S}(x_i^u, x_j^v) \\
= \ & \sum_{u,v}\sum_{i,j} \mathbf{S}(x_i^u, x_j^v)||F(x_i^u)||_2^2 \qquad (6.34)
\end{aligned}$$

where $\mathbf{D} = \mathbf{diag}(d_{11}, d_{12}, \cdots, d_{ui}, \cdots, d_{MN})$ and $d_{ui} = \sum_{v,j} \mathbf{S}(x_i^u, x_j^v)$. Similarly, the following equation is true.

$$\mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{W}) = \sum_{u,v}\sum_{i,j} \mathbf{S}(x_i^u, x_j^v)||F(x_j^v)||_2^2 \qquad (6.35)$$

Combining the equations (6.33), (6.34) and (6.35) and considering $\mathbf{S}(x_i^u, x_j^v) = S^{uv}(x_i^u, x_j^v)$, we have

$$\begin{aligned}
& 2\mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}) \\
= \ & 2\mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{W}) - 2\mathbf{tr}(\mathbf{W}^T\mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W}) \\
= \ & \sum_{u,v}\sum_{i,j} S^{uv}(x_i^u, x_j^v)\|F(x_i^u) - F(x_j^v)\|_2^2 \\
= \ & \mathcal{L}_c^e(\mathbf{G}) \qquad (6.36)
\end{aligned}$$

□

Proof of the formula (6.29 and 6.26)

Proof. For simplicity, we delete the index of projections and, then the objective function in 6.27 become similar to the function in 6.25. The only difference between them is Eq. 6.27 has a orthogonal item. Thus, if further define $H = \mathbf{XLX}^T + C_2\mathbf{Q}$, we obtain:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T H\mathbf{w} + C_1 \sum_{u=1}^{M}\sum_{i=1}^{N} \xi_i^u \tag{6.37}$$
$$s.t. \forall u, i,\ y_i^u (w^u)^T x_i^u \geq 1 - \xi_i^u,\ \xi_i^u \geq 0,$$

where the element y_i^u of $\mathbf{y}$ is the bit of initial or learned in the last round. In case of solving the problem in 6.25, the parameter C_2 can be directly set to 0. The Lagrange function of the problem 6.37 is

$$\begin{aligned} & L(\mathbf{w}, \xi, \alpha, \gamma) \\ = & \frac{1}{2}\mathbf{w}^T H\mathbf{w} + C_1 e^T \xi \\ & -\mathbf{w}^T \mathbf{X_y}\alpha + e^T\alpha - \alpha^T\xi - \gamma^T\xi, \end{aligned} \tag{6.38}$$

where $\alpha = (\alpha_1^1, \cdots, \alpha_i^u, \cdots, \alpha_N^M)^T$ and $\mathbf{X_y} = \mathbf{diag}(X_\mathbf{y}^1, \cdots, X_\mathbf{y}^u, \cdots, X_\mathbf{y}^M)$, the matrix $X_\mathbf{y}^u = (y_1^u x_1^u, \cdots, y_N^u x_N^u)$. The gradients with respect to the parameters are:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}} &= H\mathbf{w} - \mathbf{X_y}\alpha; \\ \frac{\partial L}{\partial \xi} &= C_1 e - \alpha - \gamma. \end{aligned}$$

Thus, the optimal values should satisfy the following conditions:

$$\begin{aligned} \mathbf{w}^* &= H^{-1}\mathbf{X_y}\alpha; \\ \gamma &= C_1 e - \alpha. \end{aligned}$$

Substituting the above equations into the original Lagrange function (6.38), we obtain the dual problem:

$$\begin{aligned} \alpha^* &= \arg\min_{\alpha} -e^T\alpha + \frac{1}{2}\alpha^T \mathbf{X_y}^T H^{-1}\mathbf{X_y}\alpha \\ s.t. &\quad 0 \leq \alpha_i \leq C_1. \end{aligned} \tag{6.39}$$

The problem (6.39) is a standard quadratic programming problem. Therefore, if α^* is the solution of (6.39), the optimal projection direction can be obtained as:

$$\mathbf{w}^* = (\mathbf{XLX}^T + C_2\mathbf{Q})^{-1}\mathbf{X_y}\alpha^*.$$

□

Relationship to KKT conditions

In this chapter, a non-convex constraint optimisation problem 6.22 is considered. We first divide the overall problem into finite sub-problems. Then, to solve this complex objective, we use Lagrange multipliers to transfer constraint problem to a non-constraint problem and thus a optimal solution for the primal sub-problem could be obtained by a dual problem. Here, we show that the results could be related to KKT (Karush-Kuhn-Tucker) conditions of a general optimisation problem:

$$
\begin{aligned}
& x^* = \arg\min_x \quad f_0(x) \\
s.t. \quad & f_i(x) \leq 0, \quad i = 1, \cdots, m, \\
& h_j(x) = 0, \quad j = 1, \cdots, p.
\end{aligned}
$$

where m and p are the numbers of inequality and equality constraints. We can define the Lagrangian function associated with the above problem:

$$
L(x, \mu, \nu) = f_0(x) + \sum_i \mu_i f_i(x) + \sum_j \nu_j h_j(x),
$$

where μ and ν are the dual variables. Therefore, the dual problem could be obtained:

$$
\begin{aligned}
(\mu^*, \nu^*) \quad &= \quad \arg\max_{\mu,\nu} \quad g(\mu, \nu), \\
s.t. \quad \mu_i \quad &\geq \quad 0, \quad i = 1, \cdots, m,
\end{aligned}
$$

where $g(\mu, \nu) = \inf_x(L(x, \mu, \nu))$. The dual problem $g(\mu, \nu)$ is always concave no matter whether the primal problem f_0 is convex or not. Moreover, if a strong duality holds, x is primal optimal, and both μ and ν are the dual optimal, hence the following four conditions called KKT conditions hold:

- Primal feasibility: $f_i(x) \leq 0, h_j(x) = 0 \quad i = 1, \cdots, m, \quad j = 1, \cdots, p.$
- Dual feasibility: $\mu_i \geq 0 \quad i = 1, \cdots, m.$
- Complementary slackness: $\mu_i f_i(x) = 0, \quad i = 1, \cdots, m.$
- Stationarity: $\partial L(x, \mu, \nu) = 0.$

The mathematical symbols introduced above are only used to illustrate the general framework of a constraint optimisation. Now, we turn back to our optimisation problems for cross-modal hashing. Firstly, it is obvious that the objective and constraint functions in Eq.6.25 and 6.27 are differentiable and thus

$\partial L(\mathbf{w}, \xi, \alpha, \gamma)$ could be computed. Secondly, it is easy to prove that the primal problem is convex because that the Laplacian matrix in quadratic term (highest-degree) is positive semi-definite. Therefore, the strong duality holds where the minimum of primal objective equals to the maximum of dual objective. Then, the optimal solution must satisfy the KKT conditions. Thirdly, according to the KKT Stationarity condition $\partial L(\mathbf{w}, \xi, \alpha, \gamma) = 0$, we can obtain our representation theory to build a relationship between the original optimal and the dual optimal in Eq. 6.26 and 6.29. Moreover, the dual problem (infimum of Lagrangian function) in Eq. 6.39 can be obtained when $\partial L(\mathbf{w}, \xi, \alpha, \gamma) = 0$. This is because all the objective and constraint functions are convex and thus a single optimal function could be obtained when $\partial L(\mathbf{w}, \xi, \alpha, \gamma) = 0$. Finally, the complementary slackness property provides a sparsity explanation of α in Eq. 6.26 and 6.29. That is to say, the inequality conditions in Eq. 6.37 and the optimal solution of the dual problem α^* have the following relationships:

$$\alpha_i^u > 0 \quad \Longrightarrow \quad y_i^u (w^u)^T x_i^u = 1 - \xi_i^u,$$

or equivalently,

$$y_i^u (w^u)^T x_i^u \;\; > \;\; 1 - \xi_i^u \quad \Longrightarrow \quad \alpha_i^u = 0.$$

Chapter 7

Conclusions

This chapter summarises the findings of this thesis, presents the limitations, which could not be more deeply discussed in this thesis due to the limited space, and points out some directions for future work.

7.1 Discussion

To realise the goal provided in Chapter 1, this thesis has presented four novel algorithms to solve the diverse problems of visual data association in three levels: signal camera object tracking, cross-camera person re-identification and cross-modal retrieval. In particular, the essential problem of object tracking has been carefully investigated. Based on the discoveries, the improvement of the diversity of the tracking system has also been intensively discussed from two different conditions, in which the trackers are sampled from the same function space or several different hypothesis spaces. It is more valuable to mention that the proposed tracking methods are real time and of low computational cost and one of them has been successfully executed on an intelligent Mobile with an Android platform. In addition, how to efficiently associate samples collected in two different cameras for fast person re-identification has also been studied. In addition, a framework of hetero-manifold regularisation for cross-modal hashing has been proposed to extend the research into more general cases. In total, in this thesis, efficiency is always considered to be one of the most significant factors to achieve the ends of the visual data association.

Specially,

- Chapter 3 proposed a Learn++ based tracker for visual tracking. By means of automatically adjusting the members of classifiers, a democracy mechanism has been adopted by the LPP tracker to solve numerous challenges appearing in the scenarios, simultaneously. Moreover, the LPP tracker has achieved an optimal balance between the flexibility and stability of the classifiers and between the efficiency and performance of the model as well.

- Chapter 4 explained a winner-take-all framework for object tracking by incorporating the strengths of trackers for different challenges, to further improve the performance and efficiency. It proves that different trackers have different characteristics and the combination of them is valuable. The proposed WTA framework has been tested on a large benchmark dataset and extensive experimental results have illustrated that WTA can significantly improve both the performance and the efficiency.

- Chapter 5 detailed a cross-view binary code learning method for fast person re-identification. The main advantage of this method is that it greatly speeds up the procedure of the ranking or retrieval stage, when achieving equivalent performance to the state-of-the-art methods. Moreover, three more important points have also been observed. Firstly, just as the heuristic hand-craft descriptors are used in Chapter 5 and we think that utilising a stronger pixel-based descriptor, which is learned using deep architecture, will greatly improve CBI. Secondly, maximum margin has been used in learning binary codes by other methods. However, we firstly give an inside view of the intrinsic mechanism that the Hamming distance can be minimised by minimising the Euclidean distance when the learned linear hash functions satisfy the hinge loss constraint.

- Chapter 6 introduced a novel concept of hetero-manifold for integrating the uni- and cross-modal similarities of multi-modal data in a global view. Both types of similarity are represented in the Laplacian matrix corresponding to the hetero-manifold. The Laplacian matrix appears smoothly when the Hamming distance is replaced by the Euclidean distance, which hints that no hash functions could be learned without all uni- and cross-modal similarities being defined on the hetero-manifold. Therefore, the proposed framework of hetero-manifold regularised hash function learning could benefit from the view of treating multi-modal data as a whole. The experimental results demonstrate that the proposed HMR outperforms the state-of-the-art methods on four popular datasets.

We conclude that, by fully exploiting the algorithms for solving the problems in the three situations, an integrated trace for an object moving anywhere can be definitely discovered. By using the detected traces, we have opportunities to investigate the intrinsic structures of the data, bridge the gaps between different modalities or sensors and associate the objects in different environments and platforms, then to create exciting and fascinating applications.

7.2 Future Work

Undoubtedly, visual data association is a novel, interesting but challenging area in both the theoretical research and the system development of real-world applications. Obviously, this thesis would serve as a modest spur to induce someone to come forward with more valuable contributions on visual data association in the future. To improve and extend the discoveries in this thesis, some potential research directions for future work from two perspectives will be summarised as follows to end this thesis.

On the one hand, from the perspective of theoretical research, we have following valuable directions:

- In the first level of research for single camera object association, the proposed methods have concentrated on single object association in different locations and times only. In fact, multiple objects in the same view, in which some of them maybe similar or dissimilar to each other, occur frequently in the real world. The existing multi-object tracking (MOT) methods mainly focus on instant or long historical trajectories of these targets. However, this task of MOT could be considered as a visual data association problem in which the appearance identities could be modelled as well by using multiple classifiers (agents).

- Based on the research of learning cross-view identities, the CBI can be extended for person re-identification in the setting of multiple cameras (more than two) or multi-shot setting. In fact, in a real world scenario, even in a building or a shopping mall, many more than two cameras are installed to monitor the human activities. Therefore, learning the identities of persons from more than two views is useful. Moreover, in most cases, more than one image, or even a video, could be captured and utilised for one subject in one camera. Hence, since more valuable information has been used, both the performance and robustness of the system could be further improved.

- Cross-modal online hashing would be very a promising direction to solve the problems rapidly in visual data association, or some other related tasks, when samples are obtained sequentially. Recently, due to the urgent necessity of research for large-scale data, hashing has become a very hot topic in the areas of machine learning and the data mining communities. However, there are rare online methods of hashing which have been proposed and, in the literature of hash learning, only two initial works [50, 51] considered such types of problem and there are still many challenges. Furthermore, as far as we know, the cross-modal online hashing has never been investigated in any area. In fact, in most real cases, no matter whether in a signal modality or multiple modalities setting, it is likely that the samples are generated in sequential way and, generally, collecting a sufficient dataset before building a model is a human-labour expensive task. Thus, to efficiently associate samples, it is very meaningful to propose a cross-modal online hashing method in the future.

- Similar to the development of SVM, HMR can be naturally extended to the semi-supervised setting or non-linear function learning in Reproduced Kernel Hilbert Space (RKHS). Briefly, HMR can be considered as the improvement of manifold regularisation in a setting of multiple information sources or representations. In classical methods, generally, the manifold structure could be reflected by the unlabelled samples, which are easily collected and used to penalise the smoothness of learned models. Hence, in the multi-modality setting, the unlabelled samples could be exploited to guide the learning as well. In addition, HMR can also be considered as the initial extension of the Support Vector Machine (SVM) for cross-modal tasks. The kernel version of SVM could be used to classify the samples, which are originally linear non-separable, in a high-dimensional feature space induced by a Kernel function. There are many advantageous properties in such a type of RKHS. Thus, HMR also can be extended to improve the performance by exploiting these properties. As a result, it is predictable that the role of cross-modal support vectors could be investigated in some more ways.

On the other hand, from the perspective of applications, we have the following interesting and promising developments:

- Cross-modal retrieval methods proposed in this thesis can be used in a new application: cross-age image retrieval, which was firstly discussed in the HMR paper. Given an image of a person, the task is to search the images of the same person in other certain age stages. This task is far more difficult than the two classical tasks: age estimation or cross-age face recognition. Firstly, the learned function set should be sensitive to the age variant as the system needs to search the image in a specified age. Secondly, similar to the age period discriminative characteristic, the function set is required to be individual sensitive, which means, for any age stage, the similarity between the same individual is supposed to be larger than for different individuals. Lastly, in fact, cross-age face retrieval is a task of multiple-task learning, in which both age and individual information must be sampled in a common manifold.

- Kinship verification or retrieval is also a very interesting and challenging problem. The proposed HMR also can be applied to learn kin links for four representative relations: Father-Son, Father-Daughter, Mother-Son and Mother-Daughter. Compared to cross-age face retrieval, this task of learning kin links is more challenging due to the inherited properties that are controlled by both gene and some occasional mutations, which cannot be predicted. In addition, the appearance of a face is also influenced by many other factors, such as local climate and lifestyle. However, kinship verification has many potential applications, including family album organization, image annotation, social media analysis, and missing children/parents search, etc. With exploring the common manifold, which is used to describe the cross-kin structures and local similarities between any pair of persons, the four types of relations can be integrally investigated.

- Visual data association can be exploited to solve some basic problems in Human-Computer Interaction (HCI). At present, based on the results in this thesis, several colour information based systems have been developed for controlling the computer, TV and mobile by using hand gestures or motions of the human body. Firstly, in the future, similar systems can be implemented using the depth image or other types of sensors. In fact, to develop some low-cost system for simple applications is a very promising direction for HCI. Secondly, based on a real-time system on an Android platform to play games on mobiles, it is possible to develop a system to control mobiles using any object around the user. Thirdly, the discoveries of visual data association in this thesis could be used for a natural controller in virtual reality. If users can interact with the virtual object using hands directly, rather than devices, then the experience of users will be hugely improved. Finally, the cues of visual data association can also be exploited for intelligent advertising. If people can interact with the advertisement, then people are more likely to watch it and remember the content of the adverts.

Bibliography

[1] Z. Kalal, J. Matas, and K. Mikolajczyk, “P-n learning: Bootstrapping binary classifiers by structural constraints,” in *Proc. CVPR*, 2010. xii, 37, 54, 70, 72, 79, 84

[2] F. Zheng, L. Shao, J. Brownjohn, and V. Racic, “Learn++ for robust object tracking,” in *Proc. BMVC*, 2014. xii, 84

[3] A. Li, M. Lin, Y. Wu, M.-H. Yang, and S. Yan, “Nus-pro: A new visual tracking challenge,” *IEEE Transactions on PAMI*, 2015. xii, 9, 36, 70, 84

[4] Z. Jia, A. Balasuriya, and S. Challa, “Autonomous vehicles navigation with visual target tracking: Technical approaches,” *Algorithms*, vol. 1, no. 2, 2008. 3

[5] B. Kisacanin, V. Pavlovic, and T. S. Huang, *Real-Time Vision for Human-Computer Interaction*. Springer US, 2005. 4

[6] A. Poole and L. J. Ball, “Eye tracking in human-computer interaction and usability research: Current status and future prospects,” *Encyclopedia of Human Computer Interaction*, 2005. 4

[7] Y. Fu, R. Li, T. S. Huang, and M. Danielsen, “Real-time multimodal humanavatar interaction,” *IEEE Transactions on CSVT*, vol. 18, no. 4, 2008. 4

[8] Z. Song, H. Yang, Y. Zhao, and F. Zheng, “Hand detection and gesture recognition exploit motion times image in complicate scenarios,” in *Proc. ISVC*, 2010. 4

[9] V. A. Prisacariu and I. Reid, “3d hand tracking for human computer interaction,” *Image and Vision Computing*, vol. 30, no. 3, pp. 236–250, 2011. 4

[10] O. Javed and M. Shah, “Tracking and object classification for automated surveillance,” in *Proc. ECCV*, 2002. 4

[11] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *Proc. CVPR*, 1999. 4

[12] J. C. Pereira, E. Coviello, G. Doyle, N. R. G. R. Lanckriet, R. Levy, and N. Vasconcelos, "On the role of correlation and abstraction in cross-modal multimedia retrieval," *IEEE Transactions on PAMI*, 2014. 4, 40, 41, 129, 132

[13] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1178–1188, 2008. 4, 134

[14] H. Han, C. Otto, and A. K. Jain, "Age estimation from face images: Human vs. machine performance," in *The 6th IAPR International Conference on Biometrics*, 2013. 4

[15] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou., "Neighborhood repulsed metric learning for kinship verification," *IEEE Transactions on PAMI*, vol. 36, no. 2, pp. 331–345, 2014. 4

[16] F. Zheng, L. Shao, V. Racic, and J. Brownjohn, "Measuring human-induced vibrations of civil engineering structures via vision-based motion tracking," *Measurement*, vol. 83, pp. 44–56, 2016. 4

[17] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. IJCAI*, 1981. 9, 36, 37, 53, 72

[18] M. Isard and A. Blake, "Icondensation: Unifying low-level and high-level tracking in a stochastic framework," in *Proc. ECCV*, 1998. 9, 37

[19] S. Avidan, "Support vector tracking," *IEEE Transactions on PAMI*, 2004. 9, 36

[20] ——, "Ensemble tracking," *IEEE Transactions on PAMI*, vol. 29, no. 2, pp. 261–271, 2007. 9, 31, 36, 43, 70

[21] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. CVPR*, 2013. 9, 49, 50, 60, 65, 70, 78, 79, 80

[22] S. Gong, M. Cristani, and S. Yan, *Person Re-Identification*, ser. Advances in Computer Vision and Pattern Recognition, C. C. Loy, Ed. Springer, 2013. 12, 86, 88

[23] D. Gray and H. Tao, “Viewpoint invariant pedestrian recognition with an ensemble of localized features,” in *Proc. ECCV*, 2008. 18, 38, 86, 98, 101, 128

[24] M. Rastegari, J. Choi, S. Fakhraei, H. D. III, and L. S. Davis, “Predictable dual-view hashing,” in *Proc. ICML*, 2013. 18, 36, 40, 93, 96, 104, 129

[25] Y. Freund and R. E. Schapire, “Decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997. 27

[26] B. Efron, “Bootstrap methods: another look at the jackknife,” *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979. 27

[27] M. I. Jordan and R. A. Jacobs, “Hierarchical mixtures of experts and the em algorithm,” in *Proc. IJCNN*, 1993. 27

[28] R. Polikar, “Ensemble based systems in decision making,” *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21–45, 2006. 28

[29] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proc. CVPR*, 2001. 28

[30] R. E. Schapire, “The strength of weak learnability,” *Machine Learning*, vol. 5, pp. 197–227, 1990. 28

[31] L. Valiant, “A theory of the learnable,” *Communications of the ACM*, 1984. 28

[32] A. Krogh and J. Vedelsby, “Neural network ensembles, cross validation, and active learning,” in *Proc. NIPS*, 1995. 29

[33] R. H. X. Y. Gavin Brown, Jeremy Wyatt, “Diversity creation methods: A survey and categorisation,” *Journal of Information Fusion*, vol. 6, no. 1, 2005. 29

[34] R. Elwell and R. Polikar, “Incremental learning of concept drift in non-stationary environments,” *IEEE Transactions on NN*, vol. 22, no. 10, pp. 1517–1531, 2011. 29

[35] M. Oster, R. Douglas, and S.-C. Liu, “Computation with spikes in a winner-take-all network,” *Neural Computation*, 2009. 29

[36] D. K. Lee, L. Itti, C. Koch, and J. Braun, “Attention activates winner-take-all competition among visual filters,” *Nature neuroscience*, vol. 2, no. 4, pp. 375–381, apr 1999. 29

[37] M. Riesenhuber and T. Poggio, “Hierarchical models of object recognition in cortex,” *Nature Neuroscience*, vol. 2, pp. 1019–1025, 1999. 29

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. NIPS*, 2012. 29, 49

[39] D. Dai, H. Riemenschneider, and L. V. Gool, “The synthesizability of texture examples,” in *Proc. CVPR*, 2014. 29

[40] J. Yagnik, D. Strelow, D. A. Ross, and R. sung Lin, “The power of comparative reasoning,” in *Proc. ICCV*, 2011. 29, 34

[41] T. Dean, M. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, and J. Yagnik, “Fast, accurate detection of 100,000 object classes on a single machine,” in *Proc. CVPR*, 2013. 29

[42] I. Guyon, A. Saffari, G. Dror, and G. Cawley, “Model selection: Beyond the bayesian/frequentist divide,” *Journal of Machine Learning Research*, vol. 11, pp. 61–87, 2010. 29

[43] J. Feldman and D. Ballard, “Connectionist models and their properties,” *Cognitive Science*, vol. 6, no. 3, pp. 205–254, 1982. 29

[44] A. Rakhlin and K. Sridharan, *Statistical Learning and Sequential Prediction.* Draft, 2014. 30

[45] F. Rosenblatt, “The perceptron–a perceiving and recognizing automaton,” Cornell Aeronautical Laboratory, Tech. Rep. 85-460-1, 1957. 30

[46] A. Novikof, “On convergence proofs on perceptrons,” in *Symposium on the Mathematical Theory of Automata*, vol. XII, 1962, pp. 615–622. 30

[47] J. C. Platt, “Fast training of support vector machines using sequential minimal optimization,” *Advances in kernel methods*, pp. 185–208, 1999. 30

[48] L. Bottou and Y. L. Cun, “Large scale online learning,” in *Proc. NIPS*, 2004. 30

[49] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, “Large scale online learning of image similarity through ranking,” *Journal of Machine Learning Research*, vol. 11, pp. 1109–1135, 2010. 30

[50] C. Leng, J. Wu, J. Cheng, X. Bai, and H. Lu, “Online sketching hashing,” in *Proc. CVPR*, 2015. 30, 148

[51] L.-K. Huang, Q. Yang, and W.-S. Zheng, "Online hashing," in *Proc. IJCAI*, 2013. 30, 148

[52] D. Sculley and G. M. Wachman, "Relaxed online svms for spam filtering," in *Proc. ACM SIGIR*, 2007. 31

[53] G. V. Cormack and A. Bratko, "Batch and online spam filter comparison," in *Proc. CEAS*, 2006. 31

[54] S. Shalev-Shwartz, "Online learning: Theory, algorithms, and applications," Ph.D. dissertation, Hebrew University, 2007. 31

[55] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on PAMI*, vol. 27, no. 10, pp. 1631–1643, 2005. 31, 37, 43

[56] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 3, pp. 125–141, 2008. 31, 36, 43

[57] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. CVPR*, 2006. 31, 60

[58] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on PAMI*, vol. 33, no. 8, pp. 1619–1632, 2011. 31, 37, 43, 60

[59] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. ICCV*, 2011. 31, 37, 59, 70, 79

[60] E. J. Ruiz, V. Hristidis, C. Castillo, A. Gionis, and A. Jaimes, "Correlating financial time series with micro-blogging activity," in *WSDM*, 2012. 31

[61] J. Bollen, H. Mao, and X.-J. Zeng, "Twitter mood predicts the stock market," in *Proc. CoRR*, 2011. 31

[62] E. Soulas and D. Shasha, "Online machine learning algorithms for currency exchange prediction," Computer Science Department in New York University, Tech. Rep., 2013. 31

[63] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer New York, 2002, no. 0172-7397. 31, 32

[64] J. McClurkin, L. Optican, B. Richmond, and T. Gawne, "Concurrent processing and complexity of temporally encoded neuronal messages in visual perception," *Science*, vol. 253, no. 5020, pp. 675–677, 1991. 31, 32

[65] R. D. Cook, “Fisher lecture: Dimension reduction in regression,” *Statistical Science*, vol. 22, no. 1, pp. 40–43, 2007. 31, 32

[66] Z. Zhang and H. Zha, “Principal manifolds and nonlinear dimensionality reduction via tangent space alignment,” *SIAM Journal on Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2005. 31, 32

[67] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000. 31, 32

[68] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003. 31, 32, 119

[69] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000. 31, 32

[70] R. R. Coifman and S. Lafon, “Diffusion maps,” *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 5–30, 2006, diffusion Maps and Wavelets. 31, 32

[71] K. Fukumizu, F. R. Bach, and M. I. Jordan, “Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces,” *Journal of Machine Learning Research*, vol. 5, pp. 73–99, 2004. 32

[72] I. Rish, G. Grabarnik, G. Cecchi, F. Pereira, and G. J. Gordon, “Closed-form supervised dimensionality reduction with generalized linear models,” in *Proc. ICML*. New York, NY, USA: ACM, 2008, pp. 832–839. 32

[73] X. Geng, D.-C. Zhan, and Z.-H. Zhou, “Supervised nonlinear dimensionality reduction for visualization and classification,” *IEEE Transactions on SMC, Part B: Cybernetics*, vol. 35, no. 6, pp. 1098 –1107, dec. 2005. 32

[74] O. Kouropteva, O. Okun, and M. Pietikainen, “Supervised locally linear embedding algorithm for pattern recognition,” *Pattern Recognition and Image Analysis*, vol. 2652, pp. 386–394, 2003. 32

[75] H. Li, L. Teng, W. Chen, and I.-F. Shen, “Supervised learning on local tangent space,” *Advances in Neural Networks*, vol. 3496, pp. 546 – 551, 2005. 32

[76] Sajama and A. Orlitsky, “Supervised dimensionality reduction using mixture models,” in *Proc. ICML*. New York, NY, USA: ACM, 2005, pp. 768–775. 33

[77] Z. Liang and Y. Li, "A regularization framework for robust dimensionality reduction with applications to image reconstruction and feature extraction," *Pattern Recognition*, vol. 43, no. 4, pp. 1269–1281, 2010. 33

[78] J. Yang, S. Yan, and T. S. Huang, "Ubiquitously supervised subspace learning," *IEEE Transactions on Image Processing*, vol. 18, no. 2, pp. 241–249, 2009. 33

[79] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006. 33

[80] D. Zhang, Z. Zhou, and S. Chen, "Semi-supervised dimensionality reduction," in *Proc. SIAM ICDM*, 2007, pp. 629–634. 33

[81] H. Zhang, W. Deng, J. Guo, and J. Yang, "Locality preserving and global discriminant projection with prior information," *Machine Vision and Applications*, pp. 1432–1769, 2009. 33

[82] M. Sugiyama, T. Ide, S. Nakajima, and J. Sese, "Semi-supervised local fisher discriminant analysis for dimensionality reduction," *Machine Learning*, vol. 78, no. 1, pp. 35–61, Jan. 2010. 33

[83] Y. Song, F. Nie, C. Zhang, and S. Xiang, "A unified framework for semi-supervised dimensionality reduction," *Pattern Recognition*, vol. 41, no. 9, pp. 2789 – 2799, 2008. 33

[84] X. He and P. Niyogi, "Locality preserving projections," in *Proc. NIPS*, S. Thrun, L. Saul, and B. Schölkopf, Eds., vol. 16. Cambridge, MA: MIT Press, 2004. 33

[85] D. Xu and S. Yan, "Semi-supervised bilinear subspace learning," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1671–1676, 2009. 33

[86] X. Yang, H. Fu, H. Zha, and J. Barlow, "Semi-supervised nonlinear dimensionality reduction," in *Proc. ICML*. New York, NY, USA: ACM, 2006, pp. 1065–1072. 33

[87] H. Gong, C. Pan, Q. Yang, H. Lu, and S. Ma, "A semi-supervised framework for mapping data to the intrinsic manifold," in *Proc. ICCV*, vol. 1, oct. 2005, pp. 98–105. 33

[88] G. Lin, C. Shen, D. Suter, and A. van den Hengel, "A general two-step approach to learning-based hashing," in *Proc. ICCV*, 2013. 34, 35

[89] A. Gionis, P. Indyky, and R. Motwani, “Similarity search in high dimensions via hashing,” in *Proc. Int. Conf. Very Large Datadases*, 1999. 34

[90] B. Kulis, P. Jain, and K. Grauman, “Fast similarity search for learned metrics,” *IEEE Transactions on PAMI*, 2009. 34

[91] Y. Weiss, A. Torralba, and R. Fergus, “Spectral hashing,” in *Proc. NIPS*, 2008. 34, 36

[92] J. Wang, S. Kumar, and S.-F. Chang, “Semi-supervised hashing for large scale search,” *IEEE Transactions on PAMI*, 2012. 34

[93] A. Shrivastava and P. Li., “Densifying one permutation hashing via rotation for fast near neighbor search,” in *ICML*, 2014. 34

[94] F. Yu, S. Kumar, Y. Gong, and S.-F. Chang, “Circulant binary embedding,” in *ICML*, 2014. 34

[95] X. Yi, C. Caramanis, and E. Price, “Binary embedding: Fundamental limits and fast algorithm,” in *ICML*, 2015. 34

[96] A. Shrivastava, “Simple and efficient weighted minwise hashing,” in *NIPS*, 2016. 34

[97] A. Choromanska, K. Choromanski, M. Bojarski, T. Jebara, S. Kumar, and Y. LeCun, “Binary embeddings with structured hashed projections,” in *ICML*, 2016. 34

[98] J. He, R. Radhakrishnan, S.-F. Chang, and C. Bauer, “Compact hashing with joint optimization of search accuracy and time,” in *Proc. CVPR*, 2011. 34

[99] J.-P. Heo, Y. Lee, J. He, S.-F. Chang, and S.-E. Yoon, “Spherical hashing,” in *Proc. CVPR*, 2012. 34

[100] Y. Mu, X. Chen, X. Liu, T.-S. Chua, and S. Yan, “Multimedia semantics-aware query-adaptive hashing with bits reconfigurability,” *International Journal Multimedia Information Retrieval*, vol. 1, pp. 59–70, 2012. 34

[101] M. A. Carreira-Perpinan and R. Raziperchikolaei, “An ensemble diversity approach to supervised binary hashing,” in *NIPS*, 2016. 34

[102] F. Zheng, Y. Tang, and L. Shao, “Hetero-manifold regularisation for cross-modal hashing,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 99, 2016. 34, 35

[103] C. Leng, J. Wu, J. Cheng, X. Zhang, and H. Lu, "Hashing for distributed data," in *ICML*, 2015. 34

[104] J. Wang, S. Kumar, and S.-F. Chang, "Sequential projection learning for hashing with compact codes," in *Proc. ICML*, 2010. 34

[105] Z. Jin, Y. Hu, Y. Lin, D. Zhang, S. Lin, D. Cai, and Xuelong, "Complementary projection hashing," in *Proc. ICCV*, 2013. 34

[106] M. Norouzi and D. J. Fleet, "Minimal loss hashing for compact binary codes," in *ICML*, 2011. 34

[107] X. Li, G. Lin, C. Shen, A. van den Hengel, and A. Dick, "Learning hash functions using column generation," in *ICML*, 2013. 35

[108] W. Liu, C. Mu, S. Kumar, and S.-F. Chang, "Discrete graph hashing," in *NIPS*, 2014. 35

[109] J. Wang, W. Liu, A. X. Sun, and Y.-G. Jiang, "Learning hash codes with listwise supervision," in *ICCV*, 2013. 35

[110] D. Song, W. Liu, R. Ji, D. A. Meyer, and J. R. Smith, "Top rank supervised binary coding for visual search," in *ICCV*, 2015. 35

[111] F. Zhao, Y. Huang, L. Wang, and T. Tan, "Deep semantic ranking based hashing for multi-label image retrieval," in *CVPR*, 2015. 35

[112] J. Wang, T. Zhang, J. Song, N. Sebe, and H. T. Shen, "A survey on learning to hash," *arXiv:1606.00185*, 2016. 35

[113] D. Zhang, J. Wang, and D. C. J. Lu, "Self-taught hashing for fast similarity search," in *SIGIR*, 2010. 35

[114] F. Shen, C. Shen, W. Liu, and H. T. Shen, "Supervised discrete hashing," in *CVPR*, 2015. 35

[115] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. CVPR*, 2011. 35, 90, 104

[116] R. Raziperchikolaei and M. A. Carreira-Perpinan, "Optimizing affinity-based binary hashing using auxiliary coordinates," in *NIPS*, 2016. 35

[117] W. Kong and W.-J. Li, "Isotropic hashing," in *NIPS*, 2012. 35

[118] G. Irie, Z. Li, X.-M. Wu, and S.-F. Chang, "Locally linear hashing for extracting non-linear manifolds," in *CVPR*, 2014. 35

[119] K. Zhao, H. Lu, and J. Mei, “Locality preserving hashing,” in *AAAI*, 2014. 35

[120] Y. Gong, S. Kumar, H. A. Rowley, and S. Lazebnik, “Learning binary codes for high-dimensional data using bilinear projections,” in *CVPR*, 2013. 35

[121] Y. Mu, J. Shen, and S. Yan, “Weakly-supervised hashing in kernel space,” in *CVPR*, 2010. 35

[122] M. Rastegari, J. Choi, S. Fakhraei, H. D. III, and L. S. Davis, “Predictable dual-view hashing,” in *ICML*, 2013. 35

[123] Y. Gong, S. Kumar, V. Verma, and S. Lazebnik, “Angular quantization-based binary codes for fast similarity search,” in *NIPS*, 2012. 35

[124] D. Zhang, F. Wang, and L. Si, “Composite hashing with multiple information sources,” in *Proc. SIGIR*, 2011. 36, 40, 41, 115

[125] S. Kumar and R. Udupa, “Learning hash functions for cross-view similarity search,” in *Proc. IJCAI*, 2011. 36, 40, 41, 104, 129

[126] M. M. Bronstein and A. M. Bronstein, “Data fusion through cross-modality metric learning using similarity-sensitive hashing,” in *Proc. CVPR*, 2010. 36, 40, 104, 129

[127] K. Liu, Z. Zhao, X. Guo, and A. Cai, “Anchor-supported multi-modality hashing embedding for person re-identification,” in *Proc. VCIP*, 2013. 36

[128] G. Ding, Y. Guo, and J. Zhou, “Collective matrix factorization hashing for multimodal data,” in *Proc. CVPR*, 2014. 36, 40, 104, 129, 132

[129] D. Zhai, H. Chang, Y. Zhen, X. Liu, X. Chen, and W. Gao, “Parametric local multimodal hashing for cross-view similarity search,” in *Proc. IJCAI*, 2013. 36

[130] M. Long, J. Wang, , and P. S. Yu, “Quantized correlation hashing for fast cross-modal search,” in *Proc. IJCAI*, 2015. 36

[131] A. Adam, E. Rivlin, and I. Shimshoni, “Robust fragments-based tracking using the integral histogram,” in *Proc. CVPR*, 2006. 36, 60

[132] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in *Proc. CVPR*, 2001. 36

[133] J. Kwon and K. M. Lee, “Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling,” in *Proc. CVPR*, 2009. 36

[134] Y. Wu, J. Lim, and M.-H. Yang, “Object tracking benchmark,” *IEEE Transactions on PAMI*, 2015. 36, 70, 74, 77

[135] X. Jia, H. Lu, and M.-H. Yang, “Visual tracking via adaptive structural local sparse appearance model,” in *Proc. CVPR*, 2012. 36, 60, 79

[136] W. Zhong, H. Lu, and M.-H. Yang, “Robust object tracking via sparsity-based collaborative model,” in *Proc. CVPR*, 2012. 36, 59

[137] S. Zhang, H. Yao, X. Sun, and X. Lu, “Sparse coding based visual tracking: Review and experimental comparison,” *Pattern Recognition*, 2013. 36, 70, 77

[138] K. Zhang, L. Zhang, and M.-H. Yang, “Real-time compressive tracking,” in *Proc. ECCV*, 2012. 37, 38, 60, 70, 79

[139] F. Tang, S. Brennan, Q. Zhao, and H. Tao, “Co-tracking using semi-supervised support vector machines,” in *Proc. CVPR*, 2007. 37

[140] H. Grabner, C. Leistner, and H. Bischof, “Semi-supervised on-line boosting for robust tracking,” in *Proc. ECCV*, 2008. 37

[141] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. van den Hengel, “Robust tracking with weighted online structured learning,” in *Proc. ECCV*, 2012. 37

[142] G. Li, L. Qin, Q. Huang, J. Pang, and S. Jiang, “Treat samples differently: Object tracking with semi-supervised online covboost,” in *Proc. ICCV*, 2011. 37

[143] J. Gao, H. Ling, W. Hu, and J. Xing, “Tgpr: Transfer learning based visual tracking with gaussian process regression,” in *Proc. ECCV*, 2014. 37

[144] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, “Visual tracking: An experimental survey,” *IEEE Transactions on PAMI*, 2014. 37, 38, 65, 70, 74, 77, 78

[145] B. Stenger, T. Woodley, and R. Cipolla, “Learning to track with multiple observers,” in *Proc. CVPR*, 2009. 37

[146] J. H. Yoon, M.-H. Yang, and K.-J. Yoon, “Interacting multiview tracker,” *IEEE Transactions on PAMI*, 2015. 37

[147] J. Xiao, R. Stolkin, and A. Leonardis, “Single target tracking using adaptive clustered decision trees and dynamic multi-level appearance models,” in *Proc. CVPR*, 2015. 37

[148] J. Kwon and K. M. Lee, “Tracking by sampling trackers,” in *Proc. ICCV*, 2011. 37, 59

[149] Q. Bai, Z. Wu, S. Sclaroff, M. Betke, and C. Monnier, “Randomized ensemble tracking,” in *Proc. ICCV*, 2013. 37, 70

[150] D.-Y. Leey, J.-Y. Sim, and C.-S. Kim, “Multihypothesis trajectory analysis for robust visual tracking,” in *Proc. CVPR*, 2015. 37

[151] J. Zhang, S. Ma, and S. Sclaroff, “Meem: Robust tracking via multiple experts using entropy minimization,” in *Proc. ECCV*, 2014. 37, 83

[152] J. Santner, C. Leistner, A. S. T. Pock, and H. Bischof, “Prost: Parallel robust online simple tracking,” in *Proc. CVPR*, 2010. 37, 43, 72

[153] X. Yan, X. Wu, I. A. Kakadiaris, and S. K. Shah, “To track or to detect? an ensemble framework for optimal selection,” in *Proc. ECCV*, 2012. 37

[154] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, “Multistore tracker (muster): a cognitive psychology inspired approach to object tracking,” in *Proc. CVPR*, 2015. 38, 83

[155] J. Xing, J. Gao, B. Li, W. Hu, and S. Yan, “Robust object tracking with online multi-lifespan dictionary learning,” in *Proc. CVPR*, 2013. 38

[156] S. He, Q. Yang, R. W. Lau, J. Wang, and M.-H. Yang, “Visual tracking via locality sensitive histograms,” in *Proc. CVPR*, 2013. 38

[157] Y. Wu, M. Minoh, and M. Mukunoki, “Collaboratively regularized nearest points for set based recognition,” in *Proc. BMVC*, 2013. 38

[158] W. Li, Y. Wu, M. Mukunoki, and M. Minoh, “Bi-level relative information analysis for multiple-shot person re-identification,” *IEICE Transactions on Information and Systems*, 2013. 38

[159] T. Wang, S. Gong, X. Zhu, and S. Wang, “Person re-identification by video ranking,” in *Proc. ECCV*, 2014. 38

[160] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, “Person re-identification by local maximal occurrence representation and metric learning,” in *Proc. CVPR*, 2015. 38, 98, 101, 103, 131

[161] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, “Salient color names for person re-identification,” in *Proc. ECCV*, 2014. 38, 98, 101

[162] W. Li and X. Wang, “Locally aligned feature transforms across views,” in *Proc. CVPR*, 2013. 38, 86

[163] C. Liu, S. Gong, C. C. Loy, and X. Lin, “Person re-identification: What features are important?” in *Proc. ECCV*, 2012. 38, 86, 98

[164] R. Zhao, W. Ouyang, and X. Wang, “Unsupervised salience learning for person re-identification,” in *Proc. CVPR*, 2013. 38, 101, 102, 103, 131

[165] Y. Xu, L. Lin, W.-S. Zheng, and X. Liu, “Human re-identification by matching compositional template with cluster sampling,” in *Proc. ICCV*, 2013. 39

[166] W. Li, R. Zhao, T. Xiao, and X. Wang, “Deepreid: Deep filter pairing neural network for person re-identification,” in *Proc. CVPR*, 2014. 39, 86, 98, 99, 102, 110, 128

[167] E. Ahmed, M. Jones, and T. K. Marks, “An improved deep learning architecture for person re-identification,” in *Proc. CVPR*, 2015. 39, 103

[168] S. Ding, L. Lin, G. Wang, and H. Chao, “Deep feature learning with relative distance comparison for person re-identification,” *Pattern Recognition*, vol. 48, no. 10, pp. 2993–3003, 2015. 39

[169] B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, “Person re-identification by support vector ranking,” in *Proc. BMVC*, 2010. 39, 87, 98, 101

[170] W.-S. Zheng, S. Gong, and T. Xiang, “Re-identification by relative distance comparison,” *IEEE Transactions on PAMI*, 2012. 39, 87, 98, 99, 101, 130

[171] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, “Large scale metric learning from equivalence constraints,” in *Proc. CVPR*, 2012. 39, 101, 131

[172] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, “Local fisher discriminant analysis for pedestrian re-identification,” in *Proc. CVPR*, 2013. 39, 87

[173] D. Chen, Z. Yuan, G. Hua, N. Zheng, and J. Wang, “Similarity learning on an explicit polynomial kernel feature map for person re-identification,” in *Proc. CVPR*, 2015. 39, 103

[174] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, “Learning to rank in person re-identification with metric ensembles,” in *Proc. CVPR*, 2015. 39, 103

[175] X. Liu, W. Liu, H. Ma, and H. Fu, “Large-scale vehicle re-identification in urban surveillance videos,” in *Proc. ICME*, 2016. 39

[176] L. Yang, P. Luo, C. C. Loy, and X. Tang, “A large-scale car dataset for fine-grained categorization and verification,” in *Proc. CVPR*, 2015. 39

[177] D. Zapletal and A. Herout, “Vehicle re-identification for automatic video traffic surveillance,” in *Proc. CVPR*, 2016. 39

[178] X. Liu, W. Liu, T. Mei, and H. Ma, “A deep learning-based approach to progressive vehicle re-identification for urban surveillance,” in *Proc. ECCV*, 2016. 39

[179] V. Mahadevan, C. WahWong, J. C. Pereira, T. T. Liu, N. Vasconcelos, and L. K. Saul, “Maximum covariance unfolding: Manifold learning for bimodal data,” in *Proc. NIPS*, 2011. 40, 114

[180] X. Mao, B. Lin, D. Cai, X. He, and J. Pei, “Parallel field alignment for cross media retrieval,” in *Proc. MM*, 2013. 40, 41, 114

[181] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, “Multimodal deep learning,” in *Proc. ICML*, 2011. 40, 115

[182] C. Kang, S. Liao, Y. He, J. Wang, W. Niu, and S. Xiang, “Cross-modal similarity learning : A low rank bilinear formulation,” in *Proc. CIKM*, 2015. 40, 114

[183] F. Zhu, L. Shao, and M. Yu, “Cross-modality submodular dictionary learning for information retrieval,” in *Proc. CIKM*, 2014. 40, 41

[184] Y. Zhuang, Y. Yang, and F. Wu, “Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval,” *IEEE Transactions on Multimedia*, 2008. 40, 41

[185] Y. Yang, Y. Zhuang, F. Wu, and Y.-H. Pan, “Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval,” *IEEE Transactions on Multimedia*, 2008. 40

[186] K. Wang, R. He, W. Wang, L. Wang, and T. Tan, “Learning coupled feature spaces for cross-modal matching,” in *Proc. ICCV*, 2013. 40

[187] Y. Yang, D. Xu, F. Nie, J. Luo, and Y. Zhuang, "Ranking with local regression and global alignment for cross media retrieval," in *Proc. ACM MM*, 2009. 40

[188] Y. Zhuang, Z. Yu, W. Wang, F. Wu, S. Tang, and J. Shao, "Cross-media hashing with neural networks," in *Proc. ACM MM*, 2014. 40

[189] S. Kim, Y. Kang, and S. Choi, "Sequential spectral learning to hash with multiple representations," in *Proc. ECCV*, 2012. 40, 115

[190] L. Gao, J. Song, F. Nie, Y. Yan, N. Sebe, and H. T. Shen, "Optimal graph learning with partial tags and multiple features for image and video annotation," in *Proc. CVPR*, 2015, pp. 4371–4379. 40

[191] J. Masci, M. M. Bronstein, A. M. Bronstein, and J. Schmidhuber, "Multimodal similarity-preserving hashing," *IEEE Transactions on PAMI*, vol. 36, no. 4, pp. 824–830, 2014. 40, 41

[192] O. Zoidi, N. Nikolaidis, A. Tefas, and I. Pitas, "Multi-modal label propagation based on a higher order similarity matrix," in *Workshop on MLSP*, 2015. 41

[193] S. H. Amiri and M. Jamzad, "Efficient multi-modal fusion on supergraph for scalable image annotation," *Pattern Recognition*, vol. 48, pp. 2241–2253, 2015. 41

[194] J. Song, L. Gao, F. Zou, Y. Yan, N. Sebe, and J. Wang, "Deep and fast: Deep learning hashing with semi-supervised graph construction," *Journal of Image and Vision Computing*, 2016. 41

[195] H. Lai, Y. Pan, Y. Liu, and S. Yan, "Simultaneous feature learning and hash coding with deep neural networks," in *Proc. CVPR*, 2015, pp. 3270–3278. 41

[196] J. Wang, H. T. Shen, J. Song, and J. Ji, "Hashing for similarity search: A survey," *CoRR*, vol. abs/1408.2927, 2014. 41

[197] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011. 43

[198] R. Polikar, L. Udpa, S. S. Udpa, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Transactions on SMC, Part C: Applications and Reviews*, vol. 31, no. 4, pp. 497–508, 2001. 46

[199] D. Achlioptas, “Database-friendly random projections: Johnson-lindenstrauss with binary coins,” *Journal of Computer and System Sciences*, vol. 66, pp. 671–687, 2003. 48

[200] M. Ozuysal, P. Fua, and V. Lepetit, “Fast keypoint recognition in ten lines of code,” in *Proc. CVPR*, 2007. 48, 49

[201] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. CVPR*, 2005. 52

[202] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, ISBN: 0-38731073-8, 2007. 58

[203] J. Kwon and K. M. Lee, “Visual tracking decomposition,” in *Proc. CVPR*, 2010. 59

[204] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the circulant structure of tracking-by-detection with kernels,” in *Proc. ECCV*, 2012. 60, 79

[205] L. Sevilla-Lara and E. Learned-Miller, “Distribution fields for tracking,” in *Proc. CVPR*, 2012. 60

[206] C. Bao, Y. Wu, H. Ling, and H. Ji, “Real time robust l1 tracker using accelerated proximal gradient approach,” in *Proc. CVPR*, 2012. 60

[207] B. Liu, J. Huang, L. Yang, and C. Kulikowsk, “Robust tracking using local sparse appearance model and k-selection,” in *Proc. CVPR*, 2011. 60

[208] T. Liu, G. Wang, and Q. Yang, “Real-time part-based visual tracking via adaptive correlation filters,” in *Proc. CVPR*, 2015. 70

[209] H. Wang, A. Klaser, C. Schmid, and L. Cheng-Lin, “Action recognition by dense trajectories,” in *Proc. CVPR*, 2010. 72

[210] J. C. van Gemert, M. Jain, E. Gati, and C. G. M. Snoek, “Apt: Action localization proposals from dense trajectories,” in *Proc. BMVC*, 2015. 72

[211] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, “Support vector machine learning for interdependent and structured output spaces,” in *Proc. ICML*, 2004. 74

[212] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, “Large margin methods for structured and interdependent output variables,” *Journal of Machine Learning Research*, vol. 6, pp. 1453–1484, 2005. 75

[213] J. Weston, B. Scholkopf, O. Bousquet, T. Mann, and W. S. Noble, "Joint kernel maps," in *Proc. ICCV*, 2004. 75

[214] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on PAMI*, vol. 25, no. 5, pp. 564–577, 2003. 79

[215] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on PAMI*, 2015. 83

[216] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with gaussian processes regression," in *Proc. ECCV*, 2014. 83

[217] Y. Li, J. Zhu, and S. C. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *Proc. CVPR*, 2015. 83

[218] N. Wang and D.-Y. Yeung, "Ensemble-based tracking: Aggregating crowdsourced structured time series data," in *Proc. ICML*, 2014. 83

[219] M. Hirzer, P. M. Roth, M. Kostinger, and H. Bischof, "Relaxed pairwise learned metric for person re-identification," in *Proc. ECCV*, 2012. 87

[220] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995. 92

[221] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, pp. 321–377, 1936. 93, 104

[222] C. H. Lampert and O. Kromer, "Weakly-paired maximum covariance analysis for multimodal dimensionality reduction and transfer learning," in *Proc. ECCV*, 2010. 93

[223] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. NIPS*, 2006. 94

[224] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu, "Shape and appearance context modeling," in *Proc. CVPR*, 2007. 98, 129

[225] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. CVPR*, 2014. 99, 101, 102, 103, 110, 131

[226] ——, "Person re-identification by salience matching," in *Proc. ICCV*, 2013. 101, 103, 110, 131

[227] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. CVPR*, 2010. 101, 102, 103, 131

[228] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, "Custom pictorial structures for re-identification," in *Proc. BMVC*, 2011. 101, 131

[229] C. C. Loy, C. Liu, and S. Gong, "Person re-identification by manifold ranking," in *Proc. ICIP*, 2013. 101

[230] Z. Lin, G. Ding, M. Hu, and J. Wang, "Semantics-preserving hashing for cross-view retrieval," in *Proc. CVPR*, 2015. 114

[231] S. ichi Amari, "Integration of stochastic models by minimizing ? -divergence," *Neural Computation*, vol. 19, p. 27802796, 2007. 115

[232] Y. Jia, M. Salzmann, and T. Darrell, "Learning cross-modality similarity for multinomial data," in *Proc. ICCV*, 2011. 115

[233] D. Zhang and W.-J. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *Proc. AAAI*, 2014. 126

[234] Face and G. R. W. group., "Fg-net aging database," in *http://www-prima.inrialpes.fr/FGnet/*, 2000. 129, 134

[235] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proc. ICML*, 2006. 129

[236] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *In Proc. CVPR*, 2013. 131

[237] D. Wang, P. Cui, M. Ou, and W. Zhu, "Deep multimodal hashing with orthogonal regularization," in *Proc. IJCAI*, 2015. 134

[238] Q.-Y. Jiang and W.-J. Li, "Deep cross-modal hashing," in *arXiv:1602.02255v2*, 2016. 134

[239] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001. 135