THE HONG KONG POLYTECHNIC UNIVERSITY
DEPARTMENT OF COMPUTING

LIVE VIDEO IDENTIFICATION AND TRANSMISSION OVER WIRELESS NETWORK

Yin Yuan

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

June 2013

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____(Signed)

Yin Yuan (Name of Student)

Abstract

With the advancement in mobile computing technology, the capture devices and storage for video content become more and more mature and convenient. Live video identification and transmission are two prevalent and fashionable topics in mobile multimedia computing area. The growth of online video content raises new opportunities for the processing and delivery of the contents. Compared with traditional video applications, unprecedented challenges are raised on mobile real time video identification and transmission. Massive data are created on network every day, especially for the video uploading and downloading from mobile devices to the cloud. Moreover, limited resources in the mobile wireless network, such as bandwidth and computation capacity, demand more efficient and effective approaches of real time video identification and transmission. In the interests of achieving multimedia identification and transmission higher accuracy in real time under limited resources, we proposed corresponding solutions to solve the two fundamental problems. Simultaneously, based on the proposed methodology, a mobile based multimedia computing system is needed so as to embed complex multimedia computing process and realize it in real time. This thesis consists of three parts. The first part focuses on *video transmission scheduling*, which is to schedule and distributed the multimedia resources to all the users according to the multimedia unique characters. The second part focuses on *video-based human action recognition*, which is to exact the specific feature of the action and construct a model for action recognition. In the third part, a Mobile Cloud Computing (MCC) system is developed and supported by the above mentioned technologies, namely *Real-time mobile based Video Recognition* (RVR) system. The system demonstrates the live video transmission and identification can be processed in real time over wireless network with high accuracy under scarce resources.

Firstly, we study the problem of video transmission scheduling. We propose a newly video delivery scheme, Utility Coordination Function (UCF), over 802.11 networks. Given the limited wireless resources, supporting multi-user video streaming with good video Quality-of-Service (QoS) is very challenging. The key difficulties involve providing good playback quality while also satisfying the stringent video packet delay bounds, especially for transmitting large amount of data under the limited resources such as bandwidth. The allocation of wireless resources needs to be efficient and coordination of mobile video users should have a distributed fashion. In this thesis we present a distributed framework for multi-user video streaming over an ad-hoc 802.11 like wireless networks. The proposed algorithm is based on a utility-driven mechanism that adjusts the video users' sending rates according to Application layer video buffer status. We propose multiple schemes towards different levels of user requests, and deal the problem with scalable techniques. Simulation results demonstrate that the proposed scheme is quite efficient on radio resource will have better QoS than content blind Distributed Coordination Function (DCF) scheme. Besides, we also prove that our proposed solution is robust against the possible variations in the network.

Secondly, we study the problem of video-based human action recognition. We propose a *spline* approximation approach for video based action recognition to deal with large scale database. Video action recognition is another active research topic in computer vision and communication. Effective and fast processing approaches are highly demanded. Traditional pattern recognition and machine learning techniques can solve problems for text and image with satisfactory performance. However they become less helpful when processing large amount of video data. Besides, some statistic models designed for some special video processing applications, cannot handle the general video-based pattern recognition problem. In this thesis, we have tackled these problems from several aspects including simplifying the video representation and dimensionality reduction, improving spatio-temporal modeling, and speed up the online matching issue. The proposed approach focused on merging the current training trajectories into a much smaller but discriminative dataset to accelerate the processing for matching. An extension is also considered by introducing the idea of graph embedding; we polish the subspace learning with constructing an affinity matrix, to better evaluate the similarity within the same class during the training session. Experimental results demonstrate the proposed methods work effectively and efficiently.

Thirdly, we proposed a Real-time mobile based Video Recognition (RVR) System over Mobile Cloud Computing (MCC). The cloud computing and mobile computing technologies lead to the newly emerging MCC paradigm. Three major approaches have been proposed for mobile cloud applications: 1) extending the access to cloud services to mobile devices; 2) enabling mobile devices to work collaboratively as cloud resource providers; 3) augmenting the execution of mobile applications on portable devices using cloud resources. This part focuses on the third approach in supporting mobile data stream applications by employing the proposed transmission and recognition algorithms. More specifically, we apply the optimized partitioning algorithms to the RVR system, which separates the computation of a real time video application between the cloud and mobile devices and then achieves maximum speed in processing the streaming data under predefined recognition accuracy. We first involve a real time partition algorithm for MCC based live video recognition system. Both numerical evaluation and real world experiment have been performed, and the results show that the proposed system can achieve better performance in terms of throughput than without employing the proposed algorithms.

Keywords: Video action recognition, machine learning, distance metric co-learning, spatio-temporal modeling, video transmission, scheduling, mobile cloud computing, local indexing.

Publications

Journal Paper

1. **Yin Yuan**, Jiannong Cao,"*Survey of Feature Extraction and Modeling for Mobile Cloud Video Application*", to be submitted, 2013.

2. **Yin Yuan**, Jiannong Cao, Lei Yang, Yaguang Huangfu and Rui Liu, "*A novel live video recognition algorithm with Partitioning and Execution Framework Supporting in Mobile Cloud Computing*", to be submitted, 2013.

3. **Yin Yuan**, Jiannong Cao, and Zhu Li, "*Completed Content-aware Utility Coordination for Video Communication over Wireless Network*", to be submitted to IEEE Transaction on Multimedia(T-MM), 2013.

4. **Yin Yuan**, Jiannong Cao, Haomian Zheng and Zhu Li, "*Spatio-temporal indexing for Video-based Human Action Recognition*", submitted to IEEE Transactions on Circuit System and Video Technology (T-CSVT), 2013.

5. Lei Yang, Jiannong Cao, **Yin Yuan**, Tao Li, Andy Han and Alvin Chan. "*A Framework for Partitioning and Execution of Data Stream Applications in Mobile Cloud Computing*", ACM Sigmetrics Performance Evaluation Review (PER), vol.40, no.4, 2013.

Conference Paper

1. **Yin Yuan**, Haomian Zheng, Zhu Li, Jianwei Huang and Jiannong Cao. "*Utility-driven Distributed Transmission Coordination for Video Communications over Ad-hoc Wireless Networks*", in Proc. of IEEE International Conference on Multimedia and Expo (ICME), 2011.

2. **Yin Yuan**, Haomian Zheng, Zhu Li and David Zhang. "*Hand Gesture Recognition by Appearance Space Spline Approximation and spatio-temporal Graph Embedding*", in IEEE International conference on Acoustics, Speech, and Signal Processing (ICASSP), 2010.

Acknowledgements

Pursuing the PhD is my dream from childhood, now the dream is coming true. I wish to thank many people who have been helping me for the past few years of my PhD study.

In particular, I would like to express my most sincere gratitude to my supervisor Prof. Jiannong Cao for his devoted guidance, constant encouragement, and insightful and illuminative suggestions. Prof. Cao is not only an outstanding researcher with broad knowledge, sharp intuition and grand vision, but also a very nice and kind person who encouraged me to face life with a positive attitude. It is mostly due to his compassionate treatment; with his acuminous insight and guidance, here comes this dissertation and preparing me for my future career. He has taught me to always have high expectations and to demand more of myself. I learnt a lot from him, not only on the research but on way to be a trustful and valuable people.

Another very important person who always strives to make my research life enjoyable is my husband, Dr. Haomian Zheng. I want to express my sincere gratitude to him, who stayed up with me day by day to encourage me, and gave me valuable suggestions. Now it is the high time for me to formally acknowledge his supports and contribution.

I am largely indebted to my friends and colleagues like, Lei Yang, Xuefeng Liu, Peng Guo, Weiping Zhu, Rui Liu, Wanyu Lin, Zongjian He, Guanqing Liang, Yaguang Huangfu, Tao

Li, Jie Zhou, Miao Xiong, and all other members in Prof. Caos research group, for their insightful discussions and warm friendship.

Also I would like to thank the Boyan Hall warden Prof. Han and all my friends, who gave me lots help in accommodation and made my Hong Kong life colorful.

Special thanks to the entire staff in General Office and Research Office of HKPU for their support and assistance in all administrative matters. Last, but not the least, I would like to express my deepest gratitude to my big family, especially to my parents and parents in law for their love and unstinted encouragement that enable me to complete this work.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AFM: Action Feature Model

CCA: Canonical Correlation Analysis

CSMA/CA: Carrier Sense Multiple Access with Collision Avoidance

CRF: Conditional Random Fields

CTS: Clear to Send

CW: Contention Window

DCF: Distributed Coordination Function

DIFS: Distributed Inter Frame Space

DLFT: Differential Luminance Field Trajectory

DSSS: Direct Sequence Spread Spectrum

DTW: Dynamic Time Warping

EUCF: Enhanced Utility Coordination Function

GoP: Group of Pictures

HMM: Hidden Markov Model

KNN: K-Nearest Neighbors

LAPD: Luminance Aligned Projection Distance

LDA: Linear Discrimination Analysis

LPP: Linear Projection Preserving

MACH: Maximum Average Correlation Height

OFDM: Orthogonal Frequency Division Multiplexing

PCA: Principle Component Analysis

QoE: Quality of Experience

QoS: Quality of Service

R-D: Rate-Distortion

RTS: Request to Send

SIFS: Short Inter Frame Space

SIFT: Scale Invariant Feature Transform

SVC: Scalable Video Coding

SVM: Support Vector Machine

TCCA: Tensor Canonical Correlation Analysis

UCF: Utility Coordination Function

VQ: Vector Quantization

WLAN: Wireless Local Area Network

Chapter 1

Introduction

With the development of the multimedia technology, especially those in online video repositories like Youtube and those captured from millions of surveillance cameras, are proposing a very challenging to the real-world video understanding, analysis and delivery systems. It is highly demanded that effective and efficient video scheduling and analysis technology can be developed, such as in event detection, video action recognition and delivery. These techniques requires not only the accurate and robust analysis result, but also efficiency in computation and possibly real-time performance.

Video is composed by large amount of data due to the high resolution and thousands of frames against time. Meanwhile, the consumption in watching video is quite easy. Both of these result in a highly demand of technology in video transmission and delivery in various networks. Such technology can be applied in both fixed and mobile network, wired and wireless network.

Video content analysis is the capability of automatically analyzing video to detect and determine temporal events not based on a single image. This technical capability is used in a wide range of domains including entertainment, health-care, retail, automotive, transport,

home automation, safety and security [11][90]. The algorithms can be implemented as software on general purpose machines, or as hardware in specialized video processing units.

Video patterns are high-level semantic concepts that humans perceive when observing a video sequence. Video content and event understanding attempts to offer solutions to the problem of detecting the human perception of content with a computer perception. The major challenge for content analysis and event understanding is how to effectively translate low-level input into a semantically meaningful event description[51].

Real-time video-based pattern recognition is a kind of high level task in computer vision. It relies on sufficient solutions to many lower level tasks such as denoising, edge detection, optical flow estimation, object recognition and tracking. The maturity of many solutions to these low-level problems has spurred additional interest in utilizing them for higher level tasks such as video event understanding.

Another reason for the large amount of interest in video-based techniques is the promise of intelligent systems outfitted with inexpensive cameras enabling such applications as active intelligent surveillance, summarization and indexing of video data, and human computer interaction. There are various applications around such area such as healthy care and kinetic analysis.

The challenges in video delivery is in various aspects, with the scalable of video quality, it is difficult to serve many users at the same time with a satisfactory result, especially when there is only limited bandwidth.

The problem of video pattern recognition is also still challenging due to several reasons. The noise brought by different scale, rotation and illumination will confuse the computer, with uncertainty and large variance in the particular events. On the other hand, similarity

in the appearance of different events also degrade the performance. Therefore, it is a critical problem in video pattern recognition on how to effectively separate the discriminative information together with remove the noises which will result in misunderstanding. For simplicity, we define two basic questions for video pattern recognition, how to efficiently extract and preserve the discriminative features, and how to effectively classify the extracted features into the correct categories. We define the first class of question as video representation, and the second question as modeling.

In this thesis, we address the challenges of video delivery from scheduling and tackle video pattern recognition problem from both representation and modeling aspects. We propose a utility-driven scheduling method and allocate the video resource in a more efficient way in video delivery, and present trajectory representation in our pattern recognition solutions. Dataset representation is highly emphasized in this thesis and real-time pattern recognition can be achieved. Basically we are confronted with the following problems in the processing.

Firstly, we solve the video delivery problem by proposing the concept of utility. Instead of allocating the video resource averagely, we also consider the different request in video content. In the proposed method the resource is delivered according to the urgency of request, which is represented in the format of utility. The global QoS is optimized by this method and proved to be better than any other techniques, such as DCF in 802.11.

Secondly, we address the video pattern recognition problem. The video sequence is usually composed of hundreds of frames, each of them is composed by millions of pixels. The data amount is too large to achieve an efficient content understanding. On the other hand, it is well-known that lots of redundancy exist in the video sequence. Therefore in the first step we focused on is to reduce the redundancy in a single frame, in a single video clip, in a single action class and even in the whole video dataset. In this thesis we propose curve merging method, spline approximation techniques, graph embedding approaches to better

represent the dataset with discrimination.

Finally, we focus on improving the time consumption in the video pattern recognition problem. The processing speed is a critical section in most online applications, it is highly related with the user experience. In the thesis a detail of timing analysis is provided theoretically, and some application demos are also included as example for real-time application.

The rest of this chapter is organized as follows, section 1.5 presents the unified research framework. Section 1.6 summarizes the contributions of this thesis. Finally we give an outline of the thesis in Section 1.7.

Subsequently, we present some related techniques in wireless video transmission and video pattern recognition literature. For video transmission, we only discuss about the 802.11 protocol, while for the video pattern recognition, we present the related work in representation skills, spatio-temporal modeling and real-time processing approaches respectively. We will briefly review the related works in section 1.1.

Representation refers to translating video sequences into intermediate units understandable by spatio-temporal models. In section 1.2 we provide some representation approaches.

Spatio-temporal modeling is a critical section in the video pattern recognition problem. Given input from the representation layer, the model should categorize the video sequence into several pre-defined classes. Spatio-temporal modeling has already received a lot of attention in the computer vision research community, and we will briefly introduce these methods with our comprehensive understanding and analysis in section 1.3.

Real-time application is popular nowadays because of the development of mobile devices and applications. Timing Analysis is also a critical section in most of the video pattern

recognition approaches. We will present some popular techniques for fast processing in section 1.4.

1.1 Video Delivery on Wireless Networks

IEEE 802.11 is a set of standards for implementing wireless local area network (WLAN) computer communication in the 2.4, 3.6 and 5 GHz frequency bands. They are created and maintained by the IEEE LAN/MAN Standards Committee (IEEE 802). The base version of the standard was released in 1997 and has had subsequent amendments. These standards provide the basis for wireless network products using the Wi-Fi brand.

802.11 divides each of the above-described bands into channels, analogous to the way radio and TV broadcast bands are sub-divided. For example the 2.4000 - 2.4835 GHz band is divided into 13 channels spaced 5 MHz apart, with channel 1 centered on 2.412 GHz and channel 13 on 2.472 GHz. 802.11b was based on DSSS with a total channel width of 22 MHz and did not have steep skirts. Consequently only three channels do not overlap. Even now, many devices are shipped with channels 1, 6 and 11 as preset options even though with the newer 802.11g standard there are four non-overlapping channels - 1, 5, 9 and 13. There are now four because the OFDM modulated 802.11g channels are 20 MHz wide.

Availability of channels is regulated by country, constrained in part by how each country allocates radio spectrum to various services. At one extreme, Japan permits the use of all 14 channels for 802.11b, while other countries such as Spain initially allowed only channels 10 and 11, and France only allowed 10, 11, 12 and 13. They now allow channels 1 through 13.North America and some Central and South American countries allow only 1 through 11.

In addition to specifying the channel centre frequency, 802.11 also specifies (in Clause 17) a spectral mask defining the permitted power distribution across each channel. The mask requires the signal be attenuated a minimum of 20 dB from its peak amplitude at 11 MHz from the centre frequency, the point at which a channel is effectively 22 MHz wide.

1.2 Video Representation

Representation is the organization of low-level inputs into various primitives representing the abstract content of the video data. It is motivated by providing an intermediate summarization of the video content. Compared with modeling issues, representation is not highlighted in the literature. However, every research work should consider how to present the low-level features in an efficient way. This decision is the output of representation phase and is an integral part of the video pattern recognition processing.

Researchers were interested in pixel-based representation in the past a few years. Pixel-based representations utilizes abstraction schemes that rely on single or group of pixel features such as texture and color moment. Motion history image [112] and gradient histogram [9] are examples of pixel-based representation.

Intuitively video content can also be composed by a group of different objects. Therefore object-based representation is becoming an alternative solution. Low-level input is abstracted and object properties, such as speed, position and trajectory are used as representation [25] [34] [71] [89]. Silhouettes are another popular object-based representation, which is widely used for action recognition [7] [81].

Another group of representation can be categorized into concept-based. The idea is that the daily video content is not composed by pixels, and can hardly described by a group

of object. Instead, it should be described by some semantic concept. Scale Invariant Feature Transform (SIFT) is firstly proposed in [59] and used as a definition of "word" as a representation. It is widely applied in image processing and Spatio-temporal Interest Point (STIP) is developed in [48] for video representation.

1.3 Spatio-temporal Modeling

As presented in the previous section, spatio-temporal modeling is the complementary problem to representation. The modeling phase targets on seeking formal ways to describe and recognize specific video content in a particular domain given the choice of a representation scheme. A particular spatio-temporal model is chosen based on both the capacity for representation in a particular domain and the capacity for recognition of these content as they appear in the video sequence input.

Spatio-temporal modeling methods can be categorized into many different ways. Most of the research works propose novel modeling schemes to improve the performance. The model can be either deterministic or probabilistic, either generative or discriminative. Depend on different applications, the models also vary a lot.

However, such kind of division did not fully capture the diversity of event modeling approaches in the video pattern recognition literature. Therefore we further categories the models into "state modal" and "semantic models". Noted that such kind of category is not meaning that ever model should be exclusively include into one class.

We defined the first class of these models as "state models" for the reason that they concentrate on specifying the state space of the model. Often, this state space is reduced or factorized using semantic knowledge. This class of approaches includes finite-state machines

(FSMs) and the set of probabilistic graphical model approaches. The existence (under some structural assumptions) of efficient algorithms for the learning of parameters from training as well as recognition motivates the choice of these models to model video patterns.

Higher level semantics include ordering information (including partial ordering), and complex temporal, spatial, and logical relations. These properties become important when the event domain includes high-level events, which are best expressed in qualitative terms and natural language. To this end, a group of modeling formalisms that we defined as "semantic models" have been proposed, which enable explicit specification of these complex semantic properties. Among these are Petri nets (PNs) and grammar models as well as constraint satisfaction and logic-based approaches. These models are usually fully specified using domain knowledge and are not usually learned from training data.

An effective model will be definitely helpful to the pattern recognition. Besides such models, machine learning methods will also affect the performance. We will present a review of such methods in 4.1.

1.4 Real-time Online Applications

Basically, an approach for video pattern recognition in recognition accuracy is not necessarily a practical method for an online application. Another issue we need to consider is the timing complexity. Real-time performance is nowadays on highly demand, for various online media processing, including both in video scheduling and pattern recognition.

In the previous section we discuss the video representation, spatio-temporal modeling and machine learning method. In this section we will review some popular approaches and present their timing analysis. The main advantage of the classifiers in this category is that

they are well understood.

There are many examples of pattern-recognition methods for event recognition in the literature. Nearest neighbor based classifiers are widely used in [7] [9] [82] [112]. Support vector machine (SVM) is applied in [16] [24] [75] [103] [73]. Boosting based method is tested in [15] [49] [63] [68] [86] [76]. Corresponding timing analysis report is also given in the literature.

1.5 The Unified Research Framework

In this thesis we target on solving the video delivery and pattern recognition problem. The solution is composed by three different aspects: utility-driven resource allocation for video delivery, video-based pattern recognition performance and timing analysis.

For the video delivery problem, we investigated the current problem and find out the bottleneck for video transmission. To replace the DCF mechanism on resource allocation, we proposed a utility based scheduling method. We present our proposed method in Chapter 2, and demonstrate the performance by comparing to DCF in 802.11.

For video action recognition problem, we propose a novel approach to improve the recognition accuracy. The proposed approach is based on the statistical information in the video clips. A few mathematical tools, including high-dimensional curve merging, spline approximation and graph embedding are applied to provide a representative modeling of the training clips. The proposed approach is proved in multiple dataset that the recognition accuracy outperforms some other typical approaches in the literature.

For timing complexity, we extend our work by giving a detail time analysis of our proposed

approach. We also give a demo for application to prove the correctness of our proposed approach, together with the efficiency.

1.6 Contributions

The contributions of this thesis can be divided into three different aspects on solving the video transmission problem, pattern recognition problem, and real-time solution. The highlights are summarized as follows.

- We propose a utility-driven scheduling method to solve video transmission and delivery problem. It is a general algorithm which can be applied on text, image, video and other kind of data, with the consideration of request content of media. Motivated by the benefits brought request urgencies, we present an algorithm based on the dynamic utility of different users. The proposed content-aware solution is demonstrated to provide better result than DCF mechanism in 802.11.

- We propose a novel approach for video pattern recognition problem. For video representation, we propose a trajectory base representation scheme. The video similarity is then converted to an evaluation equivalent to trajectory distance. The processing is simple and fast, which can achieve a balance between performance and processing time. Spatio-temporal modeling for video pattern recognition is the main contribution of this section. We investigate both global and local modeling scheme in this work. Multiple mathematical tools are utilized to solve the problem step by step. A curve merging method is applied to reduce the size of dataset, and then a spline approximation is used to smooth the merged curve. Graph embedding is utilized to evaluate the relationship between and within action classes. We demonstrate the effectiveness of our proposed method by testing different action dataset and outperforming the classic

approaches in the literature.

- Timing complexity is another issue we focused on during the video pattern recognition problem. We carefully evaluate the time consumption in our proposed processing method and find out the bottleneck. Then a few replacement method is proposed to achieve a better offline processing. For online section, we simplify the matching and finally achieve a real-time performance. The timing complexity is demonstrated by demos.

1.7 Thesis Organization

The rest of this thesis is organized as follows.

- In Chapter 2, we solve the video transmission and delivery problem problem by proposing a utility-driven scheduling method. Request content is considered during the resource allocation and a balance between different users are found. The proposed solution is demonstrated by experiment to be effective and performs better than DCF mechanisms in 802.11.

- In Chapter 3, we propose a trajectory-based video representation scheme, and followed by spatio-temporal modeling for video action recognition. We present two approaches focused on temporal feature and spatial feature respectively. Maximum Likelihood is used for decision. The numerical results from different dataset are competitive with the ones in the literature, so that the robustness of the proposed method is also guaranteed.

- In Chapter 4, we present a timing analysis which is extended from our proposed method in Chapter 3. We target on implementing real-time performance for online

applications. Demo is provided to prove the robustness and efficiency of our proposed solution.

- In Chapter 5, we present conclusions and propose several potential future directions of research arising from this work.

Chapter 2

Literature Review

In section 2.1 we will briefly review the techniques video streaming and scheduling approaches which are widely applied in the literature. In section 2.2, we briefly review some of the popular techniques in the video-based action recognition literature.

2.1 Existing Works about Video Transmission Scheduling

2.1.1 Wireless Video Communications

Multimedia transmission over wireless is becoming a key research field in video coding and networking. For example, people have proposed various joint source and coding schemes based on information theory to address the challenges in wireless communications. These efforts have strongly influenced the H.264 video coding standard [38]. On the other hand, there does not yet exist a unified framework of addressing the QoS problem of multimedia communications in wireless networks [26].

Recent year results show that it is feasible to support high data-rates and satisfy low delay constraints of multimedia over wireless network [62] [94]. For ad-hoc communications,

however, research on video streaming is still at the beginning period, especially a cross-layer optimization design. Recently people have stated to look at a cross-layer design approach of video streaming over wireless ad-hoc networks.

Most recent research focused only on joint optimization. It is proposed in [62] to jointly look at path diversity and video coding. In another cross-layer proposal [94], the source, channel coding, and MAC layer retransmissions are jointly and optimally designed. Power and flow are allocated through convex optimization in [100]. MAC layer scheduling combine the above proposed in [98]. Much research still to be proposed along these directions to identify and exploit optimization and cross-layer interactions in real-time video streaming over ad-hoc wireless networks.

In the wireless video delivery system, a media server contains multiple video sequences with multiple quality level provided by scalable video coding. In this work we assume that each sequence is packetized into multiple packets, and each packet is independently decodable. In some popular video encoding standards, such packet should represent some content. The content can be either as large as an entire video frame or Group of Pictures (GoP), or as small as a group of macroblocks, such as in H. 264. A header acts as a synchronization marker for each packet to guarantee every packet is independently decodable. In such a system, a packet scheduling problem emerges without a coordinator.

The server has only one channel to deliver in each delivery cycle. In other words, the server can only serve one of those multiple users each cycle. In this work we assume that the wireless network is lossless and of enough bandwidth, in which there will not be any retransmission. Besides, for simplicity, we assume each of the users being served are video users. Users are requesting video with different content and watching is going on together with delivery. Therefore, there is one packet delivery and only one user is served in each delivery cycle, meanwhile every user consume their content which is delivery in previous cycles. In this paper we propose a scheduling scheme which can maximize the global quality

of video requested and strictly guarantee the fairness among users.

2.1.2 System Model and 802.11 DCF Algorithm Descriptions

The DCF is a distributed random access scheme based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol [5]. There are some important concepts in the protocol, and we present them as follow. The basic mode in DCF is well-known as the two-way handshaking. A transmission station first senses the channel for at least a Distributed Inter Frame Space (DIFS) time. If there is no other node transmitting at this time, the station will transmit the Request to Send (RTS) and wait for a Short Inter Frame Space (SIFS) time. If the corresponding receiver station successfully receives a packet, it will send out a Clear to Send (CTS) to the transmission station. This scheme is applicable to elastic data transmission, including video content delivery.

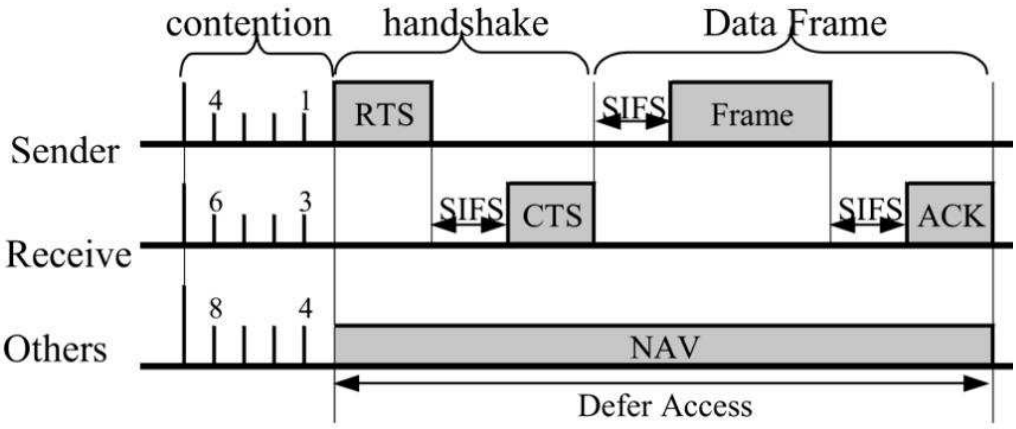

Fig. 2.1: Data Transmission in 802.11 DCF

The delivery based on DCF is completed in the following way. In every delivery cycle, each user needs to wait a random selected back-off time after DIFS, before attempting to transmit. Each station competes to access to the medium by selecting a random number from their own contention window (CW). The resource will be allocated to the user whose random number is smallest among all the users, which is a totally random distribution.

Noted that each user have a same contention window size, so in this situation, all the users are treated with equally weight, and the allocation is completed randomly. In the long run the delivery is in an average style, which will cause some problem in wireless video delivery due to the different content requested.

Most recent research considers only some joint optimization. It is proposed to jointly look at path diversity and video coding. Path diversity combined with video coding is proposed in [62]. In another cross-layer proposal [94], the source, channel coding, and MAC layer retransmissions are jointly and optimally designed. Many research works were proposed along these directions to identify and exploit optimization and cross-layer interactions in real-time video streaming over ad-hoc wireless networks.

Since the solutions above do not take the different content in video request into account, every user will obtain same amount in the long-term resource allocation. Therefore, those users with higher consumption rate, i.e., the requested video is a high bit-rate video, will consume his/her content quickly. Then this user is very likely to freeze after some time. After freezing, the DCF based delivery cannot support the request anymore and a video sequence with lower quality level will be served in a scalable video coding system, which will degrade the Quality-of-Experience. One way to resolve this issue is to allocate the resource to the users based on their remaining playback time. This motives our utility driven algorithm as follows.

2.2 Existing Works about Video-based Human Action Recognition

As an important area in computer vision, the human action and activity recognition have received much attention in recent years. A comprehensive review of this research topic has been presented in a number of survey papers, e.g., [1] [20] [90]. In this section, we mainly

focus on discussing the most critical processing in this special problem.

Pixel values can be directly obtained from an image or a video clip. So the optical flow representation, which is based on the moving pixels, has been widely used as a simple representation of the video by a lot of researchers [23] [40]. In this approach, the idea is to directly use the optical flow to derive a video representation which can be used for recognition. So motion detection and analysis from video compression work have also been combined into this technique. For example, Motion Energy Image (MEI) [8] and Motion History Image (MHI) [9] have been proposed to describe the motion information.

In general, a class of approaches for human action recognition analysis are based on the modeling of the extracted features from the video sequences. The modeling and learning of the extracted features is the critical part of the problem, in improving the accuracy of the recognition. Some popular techniques include optical motion detection, 3-D volume representation, temporal modeling, Hidden Markov Model (HMM) training, Dynamic Time Warping (DTW) and multi-view subspace learning. We offer a brief review of these techniques in the following several paragraphs.

The appearance based feature representation is not robust with respect to background changes such as scaling and rotation. Also, the failure on handling occlusions and cloth changing limited the application on these methods. Space-time interest points and their trajectories for action and activity analysis are quite popular in the recent literature [50] [70]. The main strength of this representation is the robustness to occlusions, since there is no need to detect or track the human body or hand. A dictionary can be constructed by a bag-of-words approach and therefore the image or video can be represented as statistical information of words.

Temporal properties have been proved to contribute a lot towards action classification. Compared with traditional 3-D modeling, a 4-D (x, y, z, t) action feature model (AFM) was

proposed for representation and recognition of action from arbitrary views [106]. Temporal features are also highly emphasized in [93] for creating intelligent robot systems. By utilizing Conditional Random Fields (CRF) and applying discrimination training, the algorithm is proved to be effective.

Researchers also applied Hidden Markov Models and their variants for better analysis of their temporal behavior [87]. The general methodology was to learn the appearance model of the human body or hand and match it explicitly to images in a target video sequence for action and gesture recognition [105]. This approach is highly dependent on the features extracted from the video. Different representations also has different models. In [29], actions in video clip are treated as 3-D shapes induced by silhouettes in the space-time volume and properties of the solution for Poisson equation was utilized to extract the features such as action dynamics, shape structure and orientation. The method is proved to be fast and robust to partial occlusions and can be applied to low-quality videos. Similarly, in [96] an exemplar-based Hidden Markov model (HMM) was proposed and this model took advantage of dependencies between three dimensional exemplars. Furthermore, a template-based method, named the Maximum Average Correlation Height (MACH), was proposed in [77]. By capturing the intra-class variability, the single action class is simply and carefully modeled after analyzing the response of the MACH filter.

Instead of building models for only one set of the features, there are some approaches that focused on both temporal and spatial domains. A more comprehensive understanding can be obtained during such a process. In [45], a spatio-temporal volume modeling based solution is investigated and proved to be insensitive to image formation variations. In [99], a new approach, which is composed of a 2-layer statistical field model, was proposed and demonstrated to be robust to occlusions. Besides robustness, the structure was also more flexible to image observations, which made the method robust to clutter as well. In [44] Canonical Correlation Analysis (CCA) is used to measure the similarity of any two image sets for robust object recognition. Correlation information is also considered to be helpful

for recognition. After this, in [43], a method is also applied for hand gesture recognition by combining feature selection and the Tensor Canonical Correlation Analysis (TCCA) learning process. Tensor work has also been applied for gait recognition in [88], combined with Gabor features contained in the gait sequence.

Chapter 3

Utility-Driven Coordination Function for Video Delivery over Wireless Network

3.1 Overview

With the development of computationally powerful portable devices and wireless transmitters, the demand of real-time multimedia communications and delivery is greatly increasing. However, there are still many open challenging problems on providing satisfactory Quality-of-Experience (QoE) for wireless users. One of these problems is multi-user video delivery is more complicated over wireless channels, where the demand for better video quality and real-time services is more difficult. It is demonstrated the Scalable Video Coding (SVC) [79] is providing differentiated video services against wireless time-varying channel for single user. Now a problem is emerging on how to balance the resource allocation under multi-user case, as we discuss in this thesis.

There are several existing approaches for solving similar problems. Some cellular systems are designed to support voice and video data [17], however it cannot support most video applications with high bit rate. The high rate video sources in current delivery approaches are usually adapted through a variety of schemes, such as scalable video stream extraction [72], transcoding [95] and summarization [53] before they can be accommodated by the wireless channel.

The difference in video content segments results in different rate-distortion characteristics. The application requests different video content to watch and consume different amount of resource in the network. Furthermore, the video playback rate varies against time, due to the different video content. Both of such differences should be taken into account for optimizing the network resource.

The resource measurement in video delivery is discrete. In most applications, the resource is consumed by frame of GoP, rather than bits. Many previous resource allocation schemes for elastic data fail to work under such measurement. The delivery requirements is difficult to be satisfied and a new scheduling solution, which is specially designed for video delivery, is required. Motivated by the challenges above, several cross-layer scheduling schemes and resource allocation methods are proposed in the literature. In [60], researchers targeted on maximizing the throughput of the network while maintaining fairness across multiple users [58].

The source content and channel model are jointly considered in determining the optimal delivery in many works. A thorough review of existing approaches can be found in [42]. In this work, we focus on downlink video delivery where the media server is located far away

from the wireless base station. Due to the infeasibility of adaptive video encoding in the channel, we assume the video is pre-encoded and packetized at the server side. Then the delivery problem is coming to a packet scheduling problem for the streaming of pre-encoded video. Researchers focused on the rate-distortion characteristics of video and provided many solutions to optimize the transmission of a pre-encoded sequence of video packets.

in this thesis, we start by formulating the optimization problem that is dependent on video user request for video delivery. We target on improving the QoE of user by providing video with higher quality, under the constraints of the network. A content-aware scheduling scheme is proposed for packet-based video transmission over wireless Ad-hoc networks. The delivery scheduling scheme is performed at each transmission time slot based only on the current playback status. We consider each of the user requests as different consumption rate, and the urgency of each request is also evaluated by the rate and the amount of data which can be played. We focused on the gradient-based scheduling scheme proposed in [2] and introduce a content-aware utility function to describe the urgency of each request, which will determine the delivery result. We apply our proposed method on different network conditions and compare the proposed solution with DCF scheme, which is widely used.

In this chapter we propose the Utility Coordination Function (UCF) scheme to achieve a better delivery performance, and in this work we further extend the UCF delivery method into an adaptive solution. An adaptive mapping function is introduced to guarantee the video content package will be delivered to most urgent user request. In this work we also reformulate the delivery problem into an optimization problem in allocation, which make our proposed delivery scheme have a consistent performance to the theory output.

The remaining of this chapter is organized as follows. In section 3.2 we present our utility-based model system and an approximation for implementation is proposed in section 3.3. Linear and Adaptive projection from utility to buffer window size is presented in section 3.4, and collisions are analyzed in section 3.5. The experiment setup and result will be presented in section 3.6 and also analyzed. Finally we will conclude our work in section 3.7 and propose some future ideas.

3.2 Utility-based System Model

Motivated by the observation above, we propose Utility Coordination Function (UCF) scheme in this work. The user's request is differentiated by converting the urgency of user demand into different random window size. User with higher urgency will report its corresponding random number from a smaller window size, therefore the chance of delivery is enhanced. However the proposed UCF scheme fails to achieve adaptation demand, i.e., the performance is highly parameter dependent. The parameter was fixed so that the performance will not be stable when the raw data rate is variant or there is user join and departure. In this work we proposed an Enhanced Utility Coordination Function (EUCF), which will automatically solve such problems.

3.2.1 Notations

In the online video playback case, the general scenario is shown in Fig.3.1, for the kth user in the network. The playback statistics is presented in the bar. In a real-time multiple user request video network, watching and downloading of video clip occurs simultaneously. We

denote the current playback time as $x_k(t)$, and the content until $y_k(t)$ is prepared for play. Under such assumption, at time t, there are still $\tau_k(t) = x_k(t) - y_k(t)$ to watch, if there downloading is terminated immediately. The concept of buffer time, which represents how much content is still left in the current playback, is defined to describe the urgency of the user request.

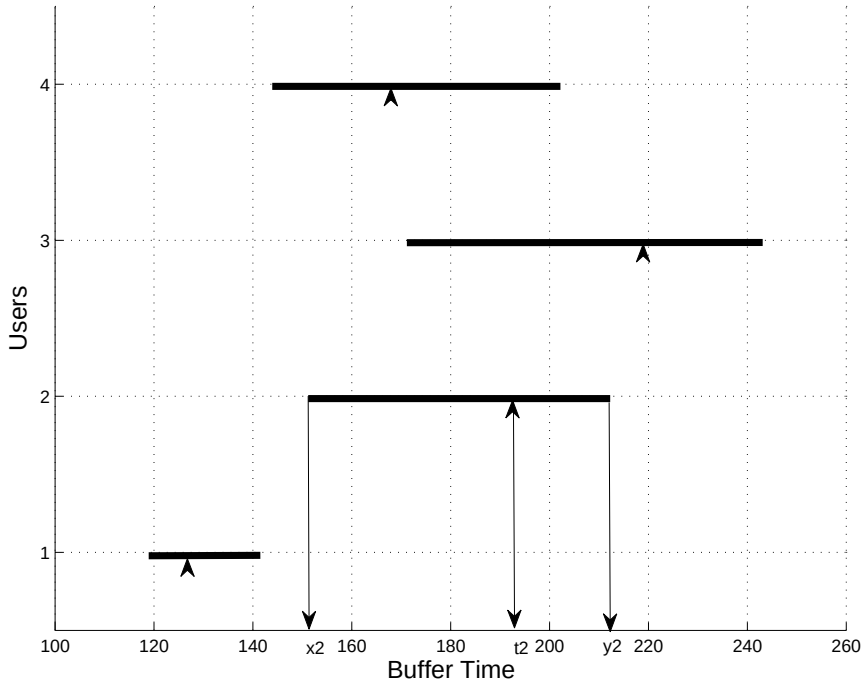

Fig. 3.1: Four video user buffer status

Denote the buffer time in the video player for user k at time t as $B_k(t)$, and for video content delivery we assume the distribution as a discrete process, i.e., each time the server distribute a given amount of data R_0 , in a time interval $\triangle(t)$ to one user. Therefore each user have their respective delivery probability, and a probability metric is introduced to describe the delivery possibility, denoted as $\{p_1, p_2, ..., p_k\}$. In fact, the server will only deliver the content to only one user. Therefore the numbers probability of distributed metric should be mutual exclusive, i.e., one of them is one with the others zero.

For the users in the network, they are requesting and watching various video clips. We assume each user has different data consumption rate, denoted as $\{R_1, R_2, , R_k\}$.

This problem is to minimize the variance of Buffer time metric $\{B(1,t), B(2,t), , B(k,t)\}$, which guarantee the fairness between users with different data rate. In this work we denote the mean value and variance of Buffer time metric as $E(B)$ and $Var(B)$ respectively. Then the buffer time metric can be computed as:

$$\begin{aligned} B(1,t) &= B(1,t-\triangle t) - R_1\triangle t + R_0 p_1(t) \\ B(2,t) &= B(2,t-\triangle t) - R_2\triangle t + R_0 p_2(t) \\ &\vdots \\ B(k,t) &= B(k,t-\triangle t) - R_k\triangle t + R_0 p_k(t) \end{aligned} \tag{3.1}$$

It is easily generated from Eq. 3.1 that the $Var(B)$ will be reduced in one deliver cycle if the packet is delivered to one user whose current buffer time is smaller than $E(B)$. The degradation in $Var(B)$ will achieved a maximal when the packet is delivered to the user with least buffer time. In other words, the fastest convergence will occur when the server deliver the content to the user with least buffer time in every delivery. However, without a coordinator, the best case is hardly to achieve due to we cannot detect the current buffer value at the server side. Therefore we are using the buffer window mechanism to determine which user to deliver. Our proposed scheme will approximate the process on minimizing the $Var(B)$, although it may be a little bit slower in convergence compared with the best case.

3.2.2 Buffer Time Optimization

In the previous section we target on minimizing the variance between buffer playback time $Var(B)$. As presented in DCF mechanism, a uniform window size is set for each user during the delivery. The delivery is determined by selecting a random number from each user's window and delivering the video content to the user with smallest selected random number.

However, the DCF mechanism does not take the different video content from each user request into account, and therefore results in an imbalance delivery among users. In this work we differentiate the request from each user and propose the following function, $f(.)$, to map the current playback time $B_k(t)$ to window buffer size $W_k(t)$.

$$W_k(t) = f(B_k(t)) \tag{3.2}$$

Since the delivery is directly determined by window buffer size, which result in corresponding delivery probability, as shown in Eq.3.3.

$$p_k(t) = \frac{W_k(t)}{\sum_{k=1}^{N} W_k(t)} = \frac{f(B_k(t))}{\sum_{k=1}^{N} f_k(t)} \tag{3.3}$$

where $p_k(t)$ is the delivery probability for user k.

To minimize the variance $Var(B)$, we are trying to compare the $Var(B(t))$ before and after

the delivery. Before the delivery, the $Var(B)$ is defined as,

$$Var(B(t)) = \frac{1}{N}\sum_{k=1}^{N}[B_k(t) - E(B(t))]^2 \tag{3.4}$$

And after the delivery, the updated $Var(B(t))$, denoted as $Var(B(t+\triangle t))$, is computed as,

$$Var(B(t+\triangle t)) = \frac{1}{N}\sum_{k=1}^{N}[B_k(t+\triangle t) - E(B(t+\triangle t))]^2 \tag{3.5}$$

Given the definition of $B(t+\triangle t)$ in Eq.3.1, we can expand the Eq. 3.5as following,

$$Var(B(t+\triangle t)) = \frac{1}{N}\sum_{k=1}^{N}[B_k(t) - R_k\triangle t - E(B(t)) + E(R_k\triangle t) - E(R_0 p_k(t))]^2 \tag{3.6}$$

Given $E(R_k(\triangle t)) = R_k(\triangle t)$ and $E(R_0 p_k(t)) = R_0 E(p_k(t)) = R_0\frac{1}{N}$, the computation of $Var(B(t+\triangle t))$ can be simplified by replacing the items in Eq. 3.6 into the following format,

$$Var(B(t+\triangle t)) = Var(B(t)) + \sum_{k=1}^{N} R_0(p_k(t) - \frac{1}{N})(2B_k(t) - 2E(B(t))) + R_0(p_k(t) - \frac{1}{N})]^2 \tag{3.7}$$

Therefore, the difference of variance before and after delivery is as follow,

$$\triangle = R_0(p_k(t) - \frac{1}{N})(2B_k(t) - 2E(B(t))) + R_0(p_k(t) - \frac{1}{N})]^2 \tag{3.8}$$

Assuming the packet is delivered to user k, then

$$p_k = \begin{cases} 0, & \text{when i = k} \\ 1, & \text{otherwise} \end{cases} \tag{3.9}$$

Therefore, Eq. 3.11 is formulated into,

$$\triangle_k(t) = 2R_0(1 - \frac{1}{N})B_k(t) - 2\frac{R_0}{N}\sum_{i=1,i\neq k}^{N} B_i(t) + C_1 \tag{3.10}$$

where C_1 is a constant independent of $B_k(t)$ and $p_k(t)$. The expectation of $\triangle_k(t)$ is,

$$\begin{aligned} E(\triangle(t)) &= \sum_{k=1}^{N} p_k(t)\triangle_k(t) \\ &= \sum_{k=1}^{N} 2R_0(1 - \frac{1}{N})p_k(t)B_k(t) - (N-1)R_0E(B(t)) \\ &= 2R_0(1 - \frac{1}{N})\sum_{k=1}^{N} p_k(t)B_k(t) + C_2 \end{aligned} \tag{3.11}$$

When the variance is decreased, the difference function $\triangle(t)$ should be minimized; it can be either positive or negative value. In this work we solve the problem by converting it into

an equivalent optimization problem:

$$\begin{aligned} &Minimize \sum_{k=1}^{N} p_k(t) B_k(t) \\ &s.t. \sum_{k=1}^{N} p_k(t) = 1 \end{aligned} \tag{3.12}$$

Given a coordinator in the network, the user k with minimal $B_k(t)$ can be selected to deliver. However, in most wireless network applications, there is no coordinator in the system. In our proposed approach, we utilize an optimal solution by matching function $f(.)$ to generate a proper transmission probability metric based on the current $B_k(t)$ metric.

3.3 Utility-based Approximation

In traditional delivery approaches, without a coordinator, the server does not have any information about the user's current playback status. The delivery fails to be efficient enough due to the miss of such information. In our proposed approach, we define a utility function to describe the current playback status for each user, which will be transmitted to the server together with the request. The utility function is an evaluation of the urgency for the user request, defined as follow,

$$U_k(\tau) = \frac{\tau_k^{1-\alpha}}{1-\alpha}, 0 < \alpha < 1 \tag{3.13}$$

In Fig. 3.2 we plot the shape of utility function. Generally in delivery, the urgency of

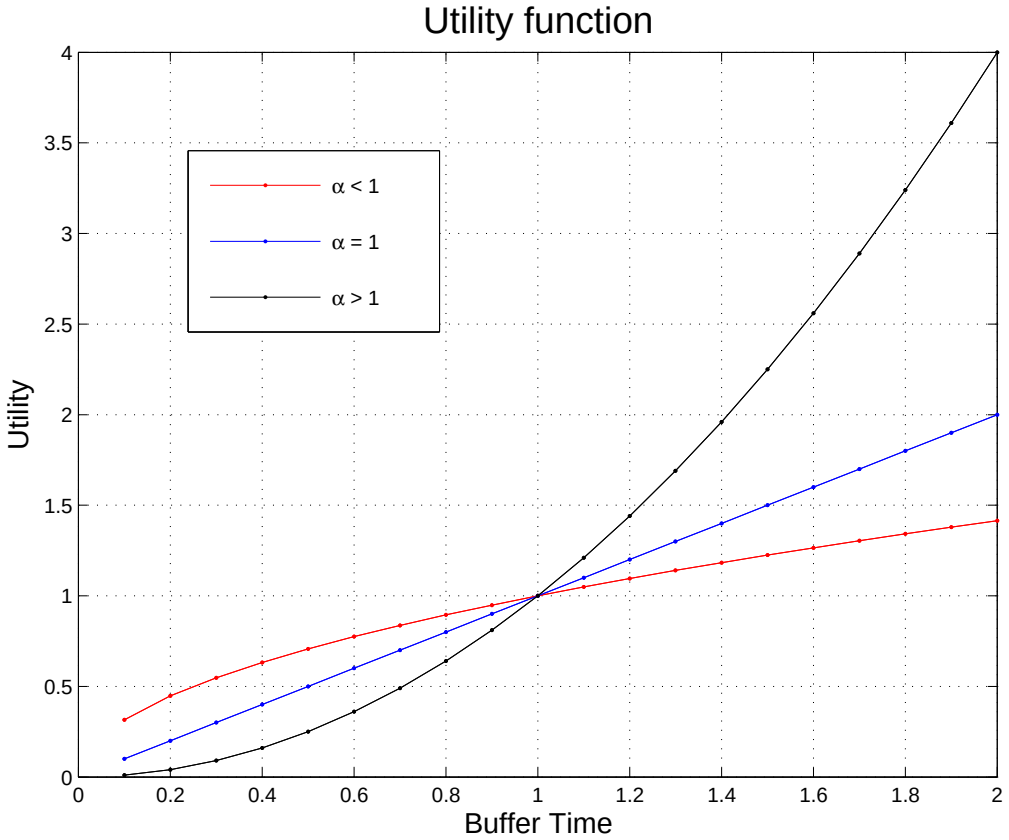

Fig. 3.2: Alpha-utility Function Shape

request is usually higher when the playback content is closer to zero. So in this work we adopt the convex utility function. The parameter α ranges from 0 to 1, to guarantee the convexity. The gradient of utility is decreasing in buffer time τ_k , i.e., a higher utility indicates a smaller τ_k and thus a higher priority is allocated to user k. Based on this, we develop a distributed backoff window based coordinating scheme that reflects the gradient by the backoff window size.

The urgency of video request could be described by the gradient of utility as follow:

$$U'(\tau) = \frac{\partial U(\tau)}{\partial \tau} = \tau^{-\alpha}, 0 < \alpha < 1 \tag{3.14}$$

The gradient of utility goes to infinity when the current playback time τ_k is closed to zero. Therefore the probability for delivery should be highly increased.

3.4 Projection from Utility to Buffer Window

3.4.1 Linear Projection

In this section we implement the utility into the scheduling, to make the delivery more efficient. In traditional DCF scheme, there is a random window for each user. Uniform window size is set in DCF delivery to equalize the weight of users, however, in real video delivery applications, users request should be differentiated due to different content. The differentiation will result in a better efficiency of delivery. Motivated by the differentiation, we propose a content-aware delivery scheme, named as Utility Coordination Function (UCF) to globally optimize the utility in the network. The window size for each user is dynamic and determined by the utility, which is a reflection of the urgency of video playback.

For each delivery, each user has a window size which is inversely proportional to the urgency function. The definition is,

$$W_k = (\beta R_0)^{-\theta} \tau_k^{\alpha\theta} = (\beta R_0)^{-\theta} \tau_k^{\lambda} \tag{3.15}$$

where the R_k is the allocated sending rate to user k, R_0 is the raw data rate of Wifi system, β is a normalized factor, and θ is the ratio to map current buffer time to window size. In this

way, the buffer time is mapped to the backoff window size by scaling and power mapping with exponent $\alpha\theta$. We name this approach as Utility Coordination Function (UCF). In this work we are using UCF to replace DCF in traditional wireless delivery to improve the quality of service in video transmission.

Compared with the content-free DCF delivery, the UCF scheme allocates the resource based on the request urgency, and thus takes the heterogeneous video contents into consideration. The proposed approach provides more flexibility and achieves a better balance among users. The set up of UCF system parameters are similar to the DCF case as shown in Table. I. The only difference is in the computing of the backoff window size as in Eq. 3.15.

The mapping of the buffer time to the backoff window size needs to be normalized in the actual implementation to reflect the system parameters, i.e., the maximum backoff window size and the system rates for the video. This will result in a variety of choices for the exponent l, as shown in the Fig. 3.3.

3.4.2 Non-linear Adaptive Projection

In previous section a proportional matching between deli-very chance and urgency is proposed. The urgency of user request is taken into consideration in the delivery, however, for the most urgent case; the importance is not highly emphasized. In this section we propose a nonlinear mapping between the buffer time and window size. This improves the robustness and is named as Enhanced Utility Coordination Function (EUCF).

In our proposed UCF delivery scheme, we are using the α utility function as shown in

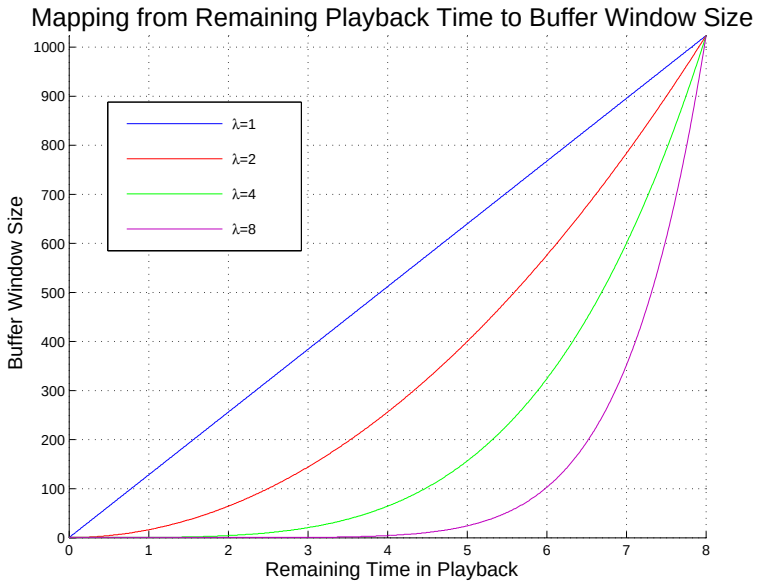

Fig. 3.3: Mapping from remaining playback time to buffer window size

Eq.3.13. In this utility function, larger α value will result in a higher urgency. Intuitively the most urgent request should be emphasized with a significant weight so that it is more likely to be delivered. Motivated by this we propose using variant α, instead of the uniform one in UCF, in this approach. The range of α value should between 0 and 1 to guarantee the convexity of utility function. The value of α is determined by the urgency of request, which is highly related with the current buffer time. Therefore, the less buffer time, the higher level of urgency with an α value more closed to 0. We are using

$$\alpha = arctan(\tau_k) \tag{3.16}$$

The $arctan(.)$ function maps the buffer time τ_k in $(0, \text{inf})$ into an interval $(0, 1)$, which is the corresponding α value. The function shape is concave so that the more urgent user will be served with a much higher probability.

Replace the corresponding α in Eq. 3.15, the formulation of window size in EUCF approach

is formulated as,

$$W_k = (\beta R_0)^{-\theta} \tau_k^{\alpha_k \theta} \tag{3.17}$$

where α_k is differentiated between different requests.

3.5 Collision Solutions

The algorithm operates as follows, in backoff window update iterations, for a given buffer status for video source k, the backoff window size is obtained from Eq.3.15. Then for a given backoff window W_k for user k, a random integer backoff counter can be selected from the interval $[1, W_k]$ with a uniform distribution. Each user sets its counter to be a random number from its own interval, and decreases the counter by one in selected slot-time [5]. All users will perform countdown simultaneously until one of the users reaches zero in the counter. If the counter of user k is zero and when the medium is sensed idle, it obtains the transmission opportunity and sends bits, where t_{data} is the data window size measured in time, and R_0 is the raw data rate of the Wi-Fi system. W_k is initially set to CW_{min}. After a successful transmission, the buffer of the user who gets the transmitted data increases and buffer size for the others reduces during the transmission time. So in the system, the buffer information is refreshed and the gradient of the utility function of every user will be updated in the next round. According to Eq.3.15, a higher utility gradient leads to a smaller W_k and thus a larger transmission probability.

In this way, the most urgent user will gain the largest resource allocation in this transmission cycle. The collision situations also need to be dealt with. It is possible that there are more than one user reached zero in a certain round, then concurrent transmissions collide with

each other. DCF scheme resolve this case by doubling the contention window for all users involved in the collision, We adopt a similar approach in our proposed scheme. If there is a collision at time t_i, the size of the contention window of each user involved in the contention will be doubled at time t_{i+1}. Let us denote the window size of user k at time t_i as $W_{k,i}$, then

$$W_{k,i+j} = 2 \times (W_{k,i} + 1) - 1, 0 \leq j \leq m \tag{3.18}$$

$$W_{k,i+j} = CW_{max}, i \geq m \tag{3.19}$$

where i represents the number of failed attempts. Here m is called the maximum backoff stage, which can be obtained by solving the following equation,

$$CW_{max} = 2^m \times (CW_{min} + 1) - 1 \tag{3.20}$$

After each successful transmission, W_k is reset for the new transmission attempt.

3.6 Experiments

In this section we present our simulation result with proposed and traditional approaches. We apply UCF and EUCF to solve a real video delivery problem in wireless network, and compare our result with DCF scheduling methods. The Wi-Fi System is typically set up with parameters in Table 3.1.

Table 3.1: MAC Layer Timing Parameters

Parameters	Value
Slot Time(μs)	9
SIFS(μs)	16
DIFS(μs)	34
CW_{min}	15
CW_{max}	1023
ACK(μs)	44
R_0(Mbit/s)	2

We study the performance of the proposed algorithm by running a MATLAB based simulation. We are using the standard video sequence to simulate 4-users case as an example. These four users are running 4 different video sequences with different rate-distortion characteristics, and they all have the live video sessions over the same 802.11 air interface at the same time. The sample of video content is displayed in Fig. 3.4. The data rate, video content and initial buffer time is listed in table 3.2. Assuming in the network there are some background traffics and overhead, we are having total rates of R_0 for the four video sessions.

Fig. 3.4: Sample Frames in our experiment

Table 3.2: Initial Information of Four Sessions

User ID	Video	Rates	Buffer(s)
1	NewsCIF@15Hz	208kbps	0.5
2	ParisCIF@15Hz	407kbps	1
3	StefanCIF@15Hz	649kbps	1.25
4	FlowerCIF@15Hz	801kbps	2.1

Table 3.3: Transmission Results for DCF Mechanisms

User ID	Video	τ_k (s)	Freeze Time(s)
1	NewsCIF@15Hz	11.81	0
2	ParisCIF@15Hz	2.94	0
3	StefanCIF@15Hz	0	6.77
4	FlowerCIF@15Hz	0	9.37

3.6.1 Invariant Network

In this section we implement our proposed scheme and compare our simulation result with DCF scheduling. To demonstrate the effectiveness of our proposed UCF scheme, we test our method with 2 different scenarios.

1) General Delivery Process;

2) Delivery Process in under-serve case.

We firstly evaluate our performance in a general delivery process, i.e., the raw data rate is sufficient to support all user's request. In this experiment we set the raw data rate to the summation of the consumption rate of each request. The corresponding buffer content status evolutions are shown in Table 3.3.

The DCF scheme is content-free and therefore the rates allocated over time are basically equal to all users. However, due to different video consumption rates for the four video requests, the 'stefan' video request goes into 'freeze' after 6.77 seconds and the 'flower' sequence goes into freeze at 9.37 seconds. The resulting video buffer states are plotted in Fig. 3.5. It is observed the two requests cannot be satisfied after a few seconds, or the users can only get a lower quality video in a scalable video coding system. This drastically degrades the Quality of Experience (QoE). Meanwhile, the 'news' video user obtains more resource than its playback requirement, and thus the video content buffered increases rapidly. This does not further improve its streaming quality but leads to a waste of the system resource and potential buffer overflow. The imbalance between users is due to the uniform window size setting and equalized delivery probability distributed to the similar resource allocation among users without considering their content differences.

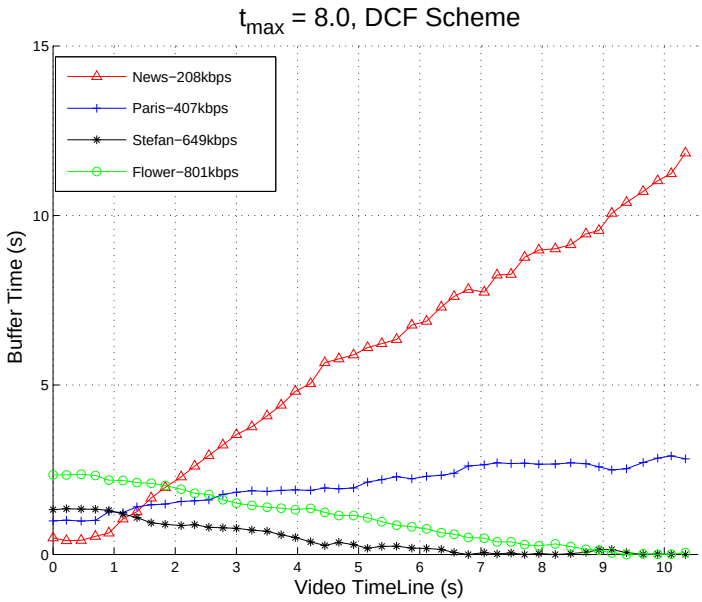

Fig. 3.5: Four users simulation for DCF mechanisms

For comparison, we simulate the same general delivery process with the UCF scheme. The resulting video buffer states are plotted in Fig. 3.6. The UCF scheme updates the backoff

window size every 20ms, and the resulting window size leads to a sending rate allocation that reflects the urgen-cy in playback buffer status. Notice that in this case, the playback buffer converges within 2 seconds, and the lower bit rate sequence news's buffer time is pulled down repeatedly in the process, while high bit rate sequence like flower's buffer is saved from underflow repeatedly in the process. Overall, the proposed UCF scheme works well with 4-user video delivery process. Balancing the different rates and buffer states among users over time, and the proposed backoff window control also works fine with exponent $\lambda = 4.0$ and max buffered video content size of $t_{max} = 8$ seconds. The final buffer time converges as the system resource and total video consumption rates reach equilibrium.

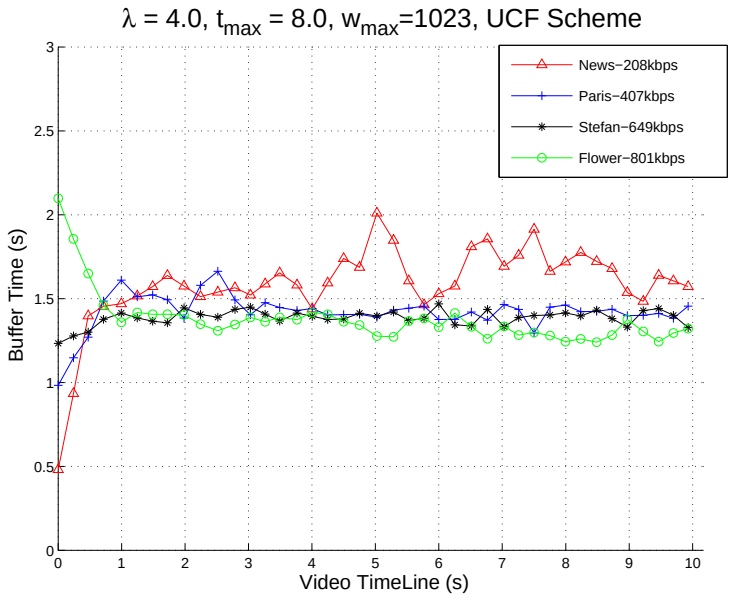

Fig. 3.6: Four users simulation for UCF mechanisms

However, in the real application, it is also an usual case that there is no enough resource to be allocated to users, which is names as under-serve. In order to demonstrate the effectiveness of our proposed method, we also test the under-serve scenario. In this case the raw data rate in the system is less than the total consumption of request, which is a worse and more challenging network condition. We plot the resulting buffer states for each request in Fig.

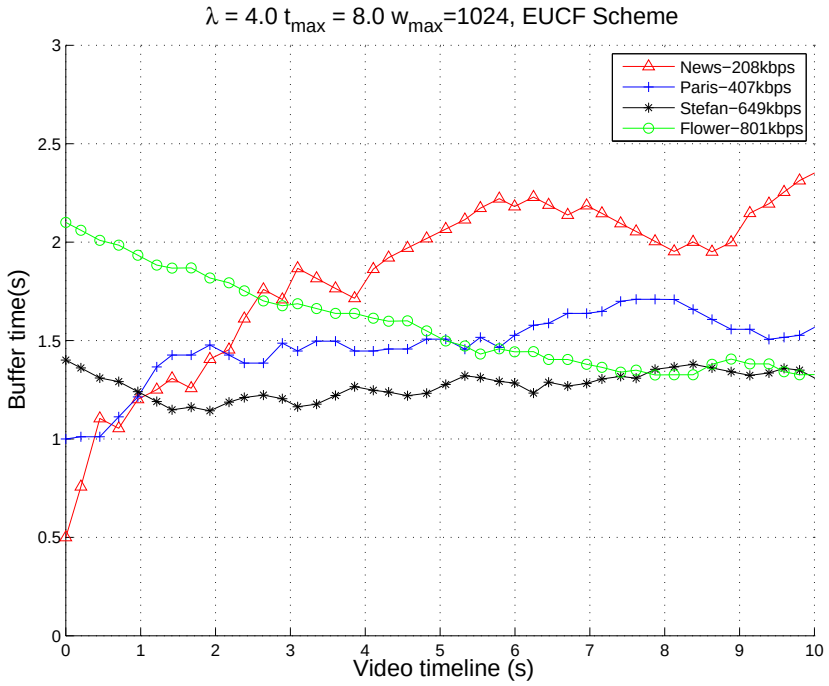

Fig. 3.7: Four users simulation for EUCF mechanisms

3.8 and Fig. 3.9.

UCF and EUCF scheme provides a smaller time scale distributed wireless resource allocation among video sessions, without the need to exchange information among video users or to a centralized coordinator. At a larger time scale, some limited exchange of information on the rate-distortion characteristics among video sessions can lead to much better rate-distortion performance than totally content blind DCF as well. This is illustrated in the Table 3.4. Without the exchange of information, DCF only allocates approximately the same rates among the video users, this can only achieve certain video PSNR quality which is suboptimal. Given the fact that the rate-distortion (R-D) characteristics are known, a resource pricing scheme [37] that searches on the slope of the R-D functions, can actually leads to an optimal rate allocation among video users. This serves as the session initialization set up and coupled with UCF/EUCF, we can achieve better PSNR qualities w.r.t to

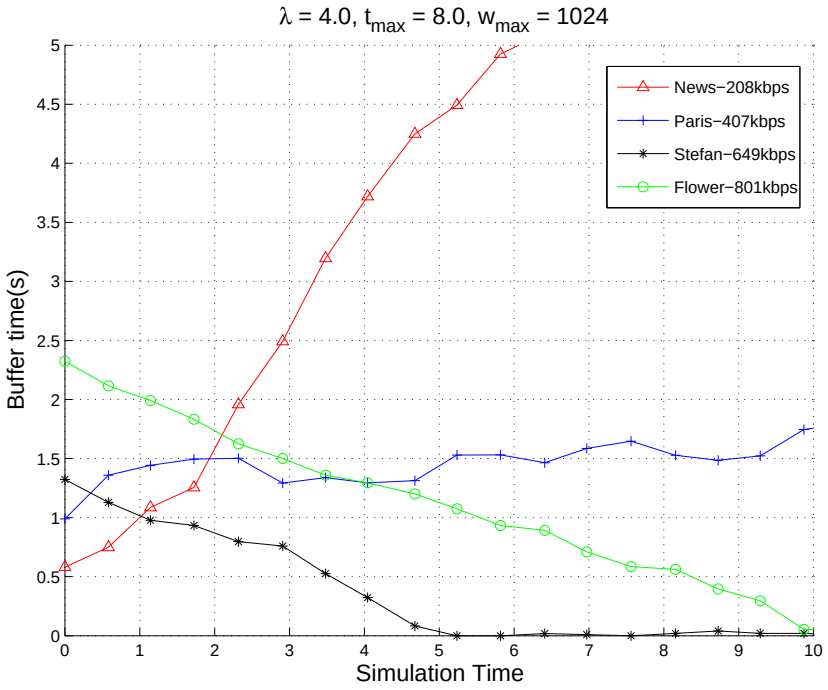

Fig. 3.8: Under-serve simulation for DCF mechanisms

Table 3.4: UCF/EUCF and DCF PSNR Comparison

User	Video	$PSNR_U$(dB)	$PSNR_D$(dB)
1	NewsCIF@15Hz	39.84	36.86
2	ParisCIF@15Hz	36.22	36.17
3	StefanCIF@15Hz	35.71	33.94
4	FlowerCIF@15Hz	36.06	32.57

the DCF schemes as well, in addition to avoiding buffer underflow.

In TABLE 3.4, the first two video sequences achieve the similar PSNR in both schemes. In fact, both sequences obtain more resources under the DCF scheme, but the streaming quality can not be further improved with the excessive resource allocation. Users 3 and 4 achieve significant performance increase in the proposed UCF/EUCF scheme due to content-aware utility driven resource allocation. The PSNR improvements compared with the DCF

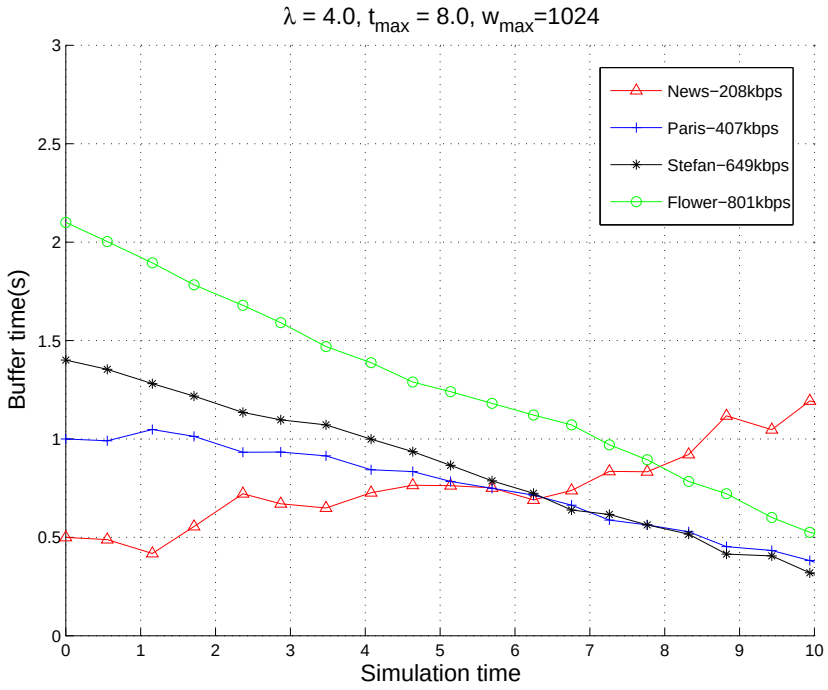

Fig. 3.9: Under-serve simulation for UCF mechanisms

case are $1.75dB$ and $1.49dB$, respectively.

3.6.2 Variant Network Simulation Result

In order to prove the effectiveness of our proposed schemes, we test our method with 3 different scenarios.

1) There is a new user join the network;

2) There is a user leave the network;

3) One of the users changed his playback video clips to another, so that his/her data rate is different.

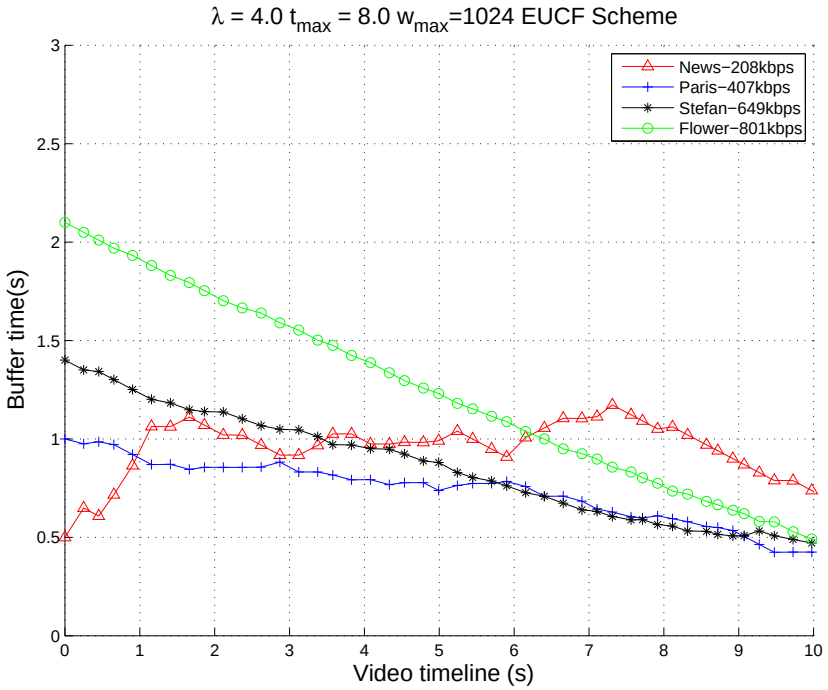

Fig. 3.10: Under-serve simulation for EUCF mechanisms

In this section we present our result for user join and departure scenarios. Given the 4 users case, we first test the case for a new user join the network. The parameter for the new user is set to a request of 1200$kbps$ video sequence with initial buffer time 0. Figure 3.11 shows the buffer time variation for each user in the network.

In the first half of the experiment, the buffer time of the initial four users are plotted. Even the initial buffer time is not same, they achieve a similar level in a short time. We can see from the result that the new comer successfully get his/her delivery in the first a few rounds, and the buffer time is converged into the same level very quickly.

Then we test the user departure case. Assuming that one of the user (user 4 in this case) leave the network, the buffer time result is plotted in Fig. 3.12.

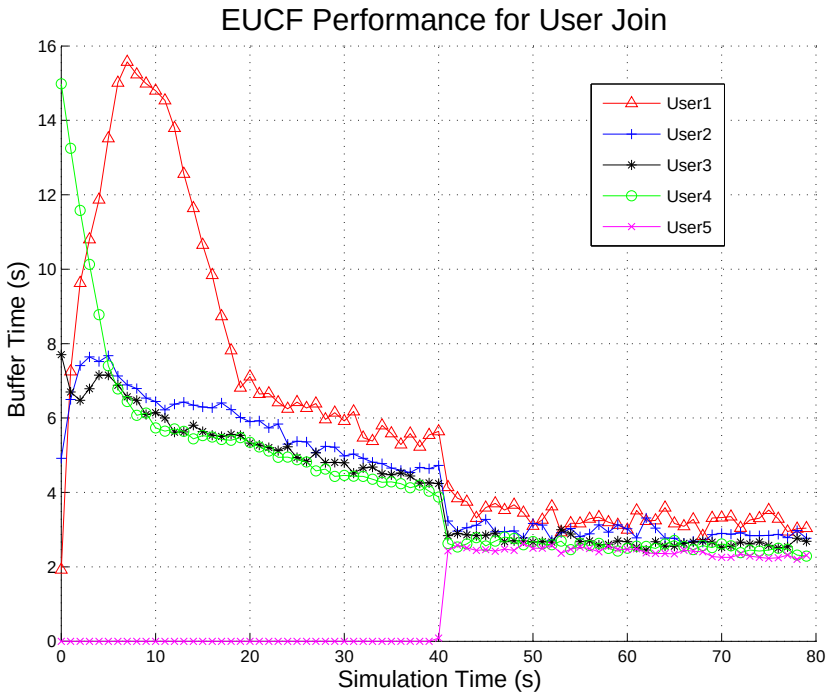

Fig. 3.11: User join with EUCF mechanisms

It is observed in Fig. 3.11 besides the left user, the buffer time of other users are uniformly increased, and they will achieve the previous balance.

Our 4-user experiment case can be extended to k users. The two experiments above proved that for any k-user situation, the EUCF system can provide stable performance to address user join and departure. For multiple user join/departure, the variance can be divided into several consecutive single user join/departure cases. The system is adaptive to user numbers from time to time, and the new equilibrium is achieved shortly.

In this section we also test our proposed EUCF scheme on user request variation. Consider the 4 user case, when one of the users (user 1 in this experiment) changes his/her request, and the data consuming rate changes correspondingly. In our experiment, we set user 1 to

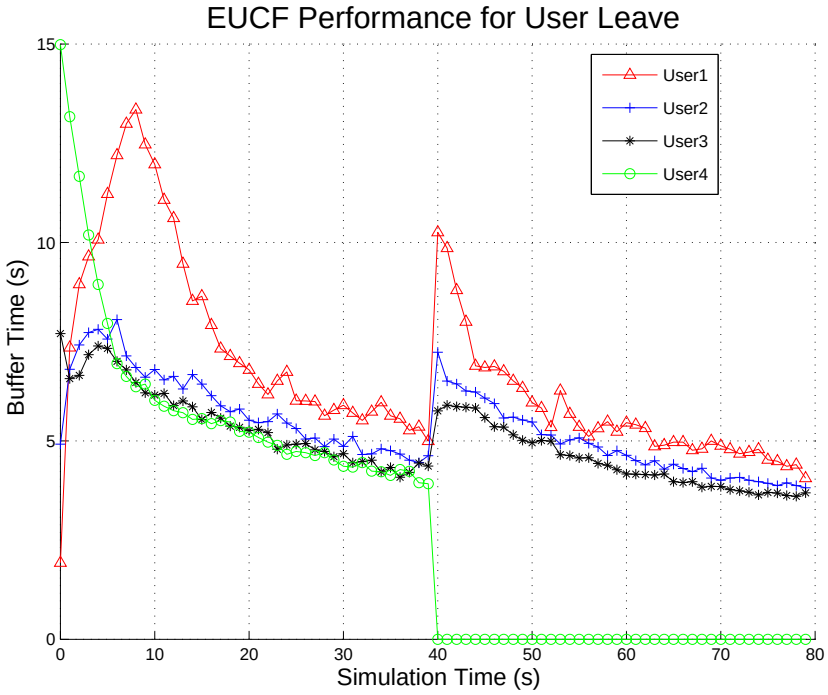

Fig. 3.12: User leave with EUCF mechanisms

change his/her consuming rate from 200$kbps$ to 1200$kbps$. The result is shown in Fig. 3.13.

From the Fig. 3.13 we can see the buffer time of user 1 (marked as red in the figure) dropped dramatically when the request changes. Then this user is delivered in a few cycles and the system converges to a new equilibrium.

In the experiment above, we prove that the proposed EUCF is more robust with variance in the wireless network. The algorithm deals with the change of user and request very quickly with effective result. The experiment focused on a 4-user case only, but the scheme can be extended to any number of users, which is a generally wireless network case.

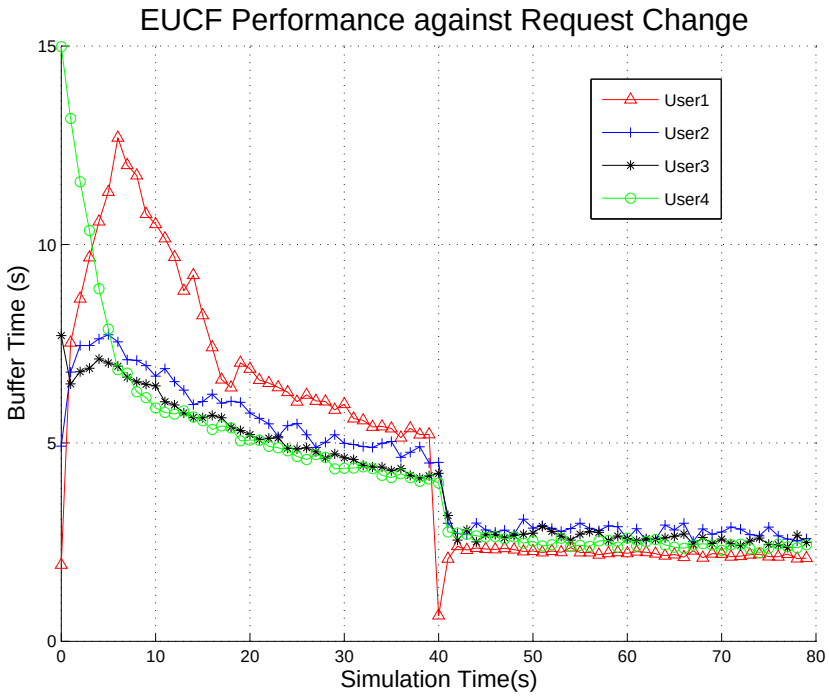

Fig. 3.13: Request change with EUCF mechanisms

3.7 Summary

In this chapter, we discuss the parameter selection in our experiment. Several parameters are introduced in our new proposed UCF and EUCF delivery scheme and these parameters will determine the performance of delivery directly.

Chapter 4

Spatio-Temporal Graph Embedding and Spline Modeling for Human Action Recognition

4.1 Overview

With the development of computing and communication technologies, video content analysis is becoming one of the most popular research areas in computer vision and machine learning. Video action pattern recognition has wide applicability in video surveillance, sports, entertainment, searching, human-computer interaction and many other activities in daily life. Basically, the problem can be defined as determining a query action into several pre-defined ones. The set of actions contains a semantic meaning in our daily life, such as running, clapping, or jumping.

Actions can be categorized into different spatio-temporal patterns. Therefore the extraction of appropriate features is critical to solve the problem. Various kinds of features such as

luminance points [40], human body detection [108], spatio-temporal interest points [48] were proposed in the literature and proved to have good performance on discriminative action recognition.

Although the central problem looks very simple, there are several quite challenging sub-problems, which are the subjects of intense research. Generally, these sub-problems can be categorized into low-level pre-possessing, human body representation, and subspace learning. The low-level pre-processing applied on the original video clip leads to the information extraction which is used as a representation of the human body for action recognition. The subspace learning basically aims to finding a subspace projected to which the discrimination can be preserved while reducing the dimensionality. A subspace can be learned in various ways to train a system, which can recognize the query video clip automatically [39] [50].

Some people-computer interactive systems are also developed in this task. In such a kind of a system, people can define some special points to help recognition. For example, in [108], the hands, feet and head of people are manually marked by the user before training and recognition. With this prior information, the performance is usually better than that of the automatically systems.

In this chapter we propose a spline-based method, which is based on luminance spatio-temporal features, to recognize different actions automatically. In the proposed method, luminance video frames are vectorized and projected into a high dimensional space thus forming a trajectory which is generated to represent the video. A spline approximation and resampling is applied afterwards to make the trajectory smooth for processing. Finally a distance metric is constructed in this way and a KNN classifier can easily be applied for

recognition.

The chapter is organized into the following sections. The formulation for our proposed methods is given in section 4.2. Solutions for proposed methods is presented in section 4.3, simulation results are shown in section 4.3 and compared with other recent works. Finally conclusions are drawn in section 4.5 and further works are included as well.

4.2 Problem Formulation

In this section we describe the problem formulation, along with several pre-processing steps: video representation, noise reduction and subspace learning. Challenges in the human action recognition tasks include human detection and representation, motion understanding and analysis. By solving these problems with appropriate algorithms, the signals can be prepared for learning and recognition.

Video clips are composed of frames which consist of pixel values. It is a challenging task to detect the human body in video sequences, especially with large visual variations and occlusions. Originally, researchers treated human as a single object in the frame so that the human body can be separated from the background. Many solutions based on this idea have already been proposed in the literature. Traditional methods focused on detecting and recognizing different human actions, such as in [18] and [111]. The main techniques involved are the so-called "object-extraction-based" method [85], which extracted a certain object by image processing techniques, such as edge detection and object segmentation processing. However, the appearance of human body in video sequences may not be very concrete and is

easily corrupted by noise. This approach suffers from lack of robustness to lighting, nature of the background, and occlusions.

To make algorithms more robust, different kinds of video representations were introduced to capture the invariance in the video, such as local image features or spatio-temporal interest points in [48], which provided a compact and abstract representation for patterns within a given image. The applications included object detection, tracking, and segmentation. The performance was demonstrated to be robust for variations of background. The so-called Scale Invariant Feature Transform (SIFT) points were then proposed in [59], and a method was designed for extracting distinctive invariant features based on the SIFT points. The SIFT points were selected by calculating the Difference of Gaussians at every pixel and representing as descriptors in different directions. Points of interest can also be encoded as a histogram [81] and this kind of representation is combined with a Support Vector Machine (SVM) [12] or some other probabilistic model. Using similar ideas, a generative graphical model in [66] used the interest points for human action recognition. This method analyzed the human action directly in the space-time volume without explicit motion estimation [82].

On the action understanding side, people focused on detecting the type of action by motion analysis. After extracting the human in the video clip, the human can be represented by several special parts, such as, arms and legs. The action is analyzed by detecting the motion of these parts and models for different actions can be learned from the given motions during the training process [108]. With the various backgrounds and different viewing angles, how to effectively detect the critical points on human body becomes the main difficulty for these approaches.

To avoid the detecting difficulties in critical points, appearance based approach is proposed to solve more general problems. Subspaces can be learned from the training video clips and used to model the query ones. Traditional subspace modeling includes Principle Component Analysis (PCA) [91] and Linear Discrimination Analysis (LDA) [4] and so on. Linear Projection Preserving (LPP) is proposed in [32] to build a graph on understanding the neighboring information.

Another popular approach is to treat the human action as a sequence and learn the model from the difference in the temporal domain. Besides, in [31], a non-linear principal curve approximation was developed. Intuitively, it is a curve passing through the center of the data points cloud, with a smoothness constraint. In [46] and [47], it was demonstrated that as long as the second moment of the data points cloud is finite, there must be a principal curve, and an iterative polygonal principal curve learning algorithm was developed.

In this work, we model human action video clips as manifolds in the scaled appearance space over time. Video clips of different human actions performed by different subjects under different image formation conditions span a space with complex structure and relationships. By scaling the original video frames into icon images, the local noise can be effectively attenuated while the information about the action is maintained. The formulation can be divided into video representation, subspace learning and matching problem.

4.2.1 Video Representation

Video representation is the first and also one of the most important sub-problems in video pre-processing. A good representation should include the key point and useful information

for discrimination while discarding unnecessary information.

Generally, in video processing, video frames are usually represented as a matrix. In our method, we use the luminance information to keep the data in every single frame, with a vector structure. To simplify pre-processing, pixel values are directly extracted as features. The video frame is first down-sampled to a smaller icon to reduce spatial redundancy together with noises, and then the icon is projected into a high dimensional space and become a point. In this way, different video clips of different human actions performed by different subjects under different image formation conditions span a space with complex structure and relationships. The spatial features in the clips are kept in a vector form while the temporal ones are included in the trajectory as well.

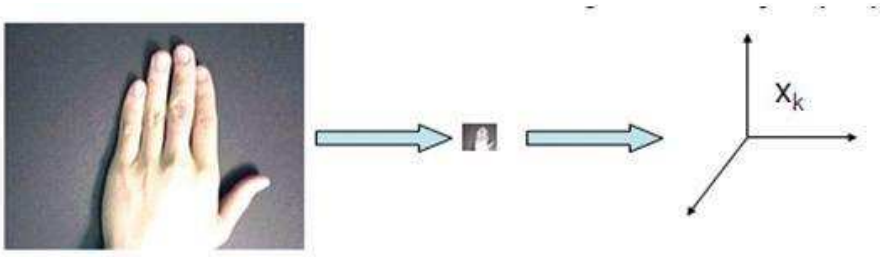

Fig. 4.1: Video frame representation by down-sampling and projection

Considering a video clip which contains n frames, with $W \times H$ pixels in each frame, the kth frame F_k can be represented as a point in the space $\Re^{W \times H}$. Actually the frame of size $W \times H$ still contains more information than necessary, so down-sampling it will reduce the number of elements while keeping adequate information for recognition. The down-sampling step reduces the original frame down to a smaller $w \times h$ one. By down-sampling each frame can be further represented as a point in a space of smaller dimension, i.e., $\Re^{w \times h}$. After this processing, the trajectory still contains sufficient statistical discriminative information for

classification. The left part of Fig. 4.1 shows the down sample processing.

4.2.2 Dimensionality Reduction

To further simplify the processing, another pre-processing step is introduced by subspace learning. In this step a global subspace is learned. By projecting the every sample points to the subspace, the discriminative information in the set is maintained while the number of dimension is reduced for faster processing. A global PCA [91] is applied here to reduce the dimensionality of the space, consider a n-frame video sequence, given a frame $F_k \in \mathfrak{R}^{w \times h}, k \in [1, n]$, the subspace learning can be expressed as:

$$x_k = AF_k = [a_1, a_2, ..., a_{w \times h}]F_k, a_j \in \mathfrak{R}^d \tag{4.1}$$

where the subspace projection A, of size d by $w \times h$, is obtained from an unsupervised local learning, with the objective of preserving the maximum amount of information, while keeping the number of dimension an acceptable level. The global subspace projection is shown in the right part of Fig. 4.1. Each a_j in 4.1 is a $d \times 1$ column vector of matrix A.

Figure 4.2 shows 3 groups of curves, each for a different human action in the Cambridge hand gesture dataset in 3-D space. In the figure, video clips containing different actions have different shapes, and as one can judge some actions are clearly different, while for others it is rather difficult to distinguish them since the 3-D view cannot offer enough visual information for doing so. Actually the geometry information of these curves already contains sufficient statistics to recognize the different human actions. In the next 3 sections,

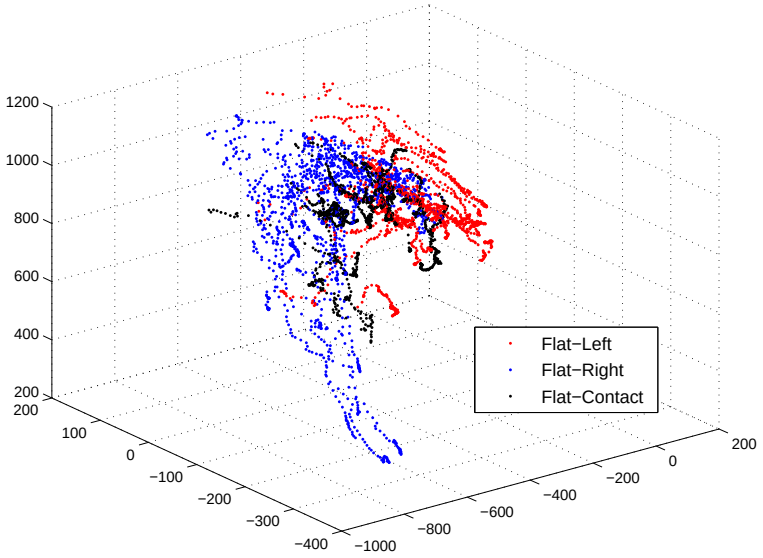

Fig. 4.2: 3-D example for Cambridge Handgesture Dataset

we will propose our approach based on thus statistical information.

4.2.3 Maximum Likelihood Detection

Video clips are represented as trajectories in a high dimensional space R^d after the preprocessing and dimension reduction. The representation is still not simple enough to discriminate the different action classes.

Assume that the training point set $\{x_1, x_2, ..., x_n\}$ in the R^d space is with Gaussian distribution, then we can obtain the mean m_x and the variance σ_x respectively. Given a query frame q, and a training frame x, the likelihood that q and x have same action label is also

under a Gaussian distribution, i.e.,

$$\mathcal{L}(q;x) \sim N(m_x, \sigma_x) \tag{4.2}$$

Since x is a Gaussian Mixture, the likelihood can be further computed as

$$\mathcal{L}(q;x) = \mathcal{L}(q;m_x,\sigma_x) = \frac{1}{2\pi^{d/2}\sigma_x^{1/2}} e^{-\frac{1}{2}(q-m_x)^T\sigma_x^{-1}\sigma_x(q-m_x)} \tag{4.3}$$

In this way, given a query trajectory $q(t)$ and a training sequence $x(t)$, $t = 1, 2, ..., n$, the likelihood between q and x can be computed as:

$$\mathcal{L}(q(t);x(t)) = \prod_{t=1}^{n} \mathcal{L}(q(t);m_x(t),\sigma_x(t)) \tag{4.4}$$

If there are totally k training trajectories, we can compute the k likelihood and then decide the action label of $q(t)$ by using Maximum Likelihood decision as follow,

$$k^* = \arg\max_k \prod_{t=1}^{n} \mathcal{L}(q(t);m_k(t),\sigma_k(t)) \tag{4.5}$$

However, in the trajectory matching, another challenge is to match up the sequences with different durations. Trajectories with different length may have different matching options. If we consider every possible matching option, the growth indifference of duration will make the matching complexity grows exponentially, which is computationally prohibitive. In this work we compute point-to-point likelihood, which strictly keep the temporal information

in the trajectory. It is constrained that points must be matched according to order. No skipping, repeating or crossing is allowed for matching. In this way, the matching is simplified into a linear level. In order to find the optimal matching, likelihood of all the possible matching offset between two trajectories are computed. The maximal value is selected as the best evaluation. Therefore, the likelihood computation is corrected into,

$$\mathcal{L}(q(t_q);x(t_x)) = \begin{cases} \min\limits_{h} \prod\limits_{t=1}^{n} \mathcal{L}(q(t_x+h);m_x(t_x),\sigma_x(t_x)) & \mathrm{t}_q > t_x \\ \prod\limits_{t=1}^{n} \mathcal{L}(q(t_q);m_k(t_x),\sigma_k(t_x)) & \mathrm{t}_q = t_x \\ \min\limits_{h} \prod\limits_{t=1}^{n} \mathcal{L}(q(t_q);m_x(t_q+h),\sigma_x(t_q+h)) & \mathrm{t}_q < t_x \end{cases} \tag{4.6}$$

In this way, a time align process is performed and the matching computation is simplified to a linear level. To achieve optimal matching, likelihood of all the possible matching offset between two trajectories are computed. The maximal likelihood is selected as the best evaluation. Then the decision can be made as,

$$k^* = \arg\max_k \sup_h \prod_{t=1}^{n} \mathcal{L}(q(t_q);x(t_x)) \tag{4.7}$$

4.2.4 Luminance Aligned Projection Distance Approach

The Bayesian-based likelihood solves the trajectory matching problem effectively, but not efficiently. The computation for likelihood in Eq. 4.3 is very complicated and limits the method to many real-time applications. In this section we propose a simplified version, the Luminance Aligned Projection Distance (LAPD) approach, to efficiently solve the matching problem.

Given a training set after dimension reduction, the parameter d, m_x and σ_x are constant for every query clip. So the computation of likelihood can be simplified by removing the first multiplier factor in Eq. 4.3, i.e.,

$$\mathcal{L}(q;x) = \mathcal{L}(q;m_x,\sigma_x) = e^{-\frac{1}{2}(q-m_x)^T\sigma_x^{-1}\sigma_x(q-m_x)} \tag{4.8}$$

The exponential function is a monotone increasing function, and the detection of maximum likelihood is equalized to finding the minimum Mahalanobis distance from query clip q to m_x. The distance measurement is more reliable when a subject is repeating same action under illumination conditions or in different background. Especially, the subspace can be directly obtained by decomposing the covariance matrix, $S = \sigma^{-1}\sigma = A^TA$. Based on this observation, we propose a distance-based approach to detect the maximum likelihood. Intuitively, the distance between two trajectories, is believed to be an effective and reliable measurement for similarity. Samples which have similar content should have smaller distance, as compared to those with dissimilar content. To make the decision more promising, we use a KNN classifier rather than maximum likelihood detection, which is equivalent to a 1-NN classifier. The classifier is applied after the distance metric is computed to label the query clip.

Distance from Point to Trajectory

In this section the distance between a single point and a trajectory in a d-Dimensional space, with subspace modeling A, is discussed.

Basically, the distance between two points in the subspace A can be defined as Eq. 4.9.

$$d(x, y) = ||A(x - y)||^2 = (x - y)^T A^T A(x - y) \tag{4.9}$$

where both x and y are points in $\Re^d$. Specially, when A is a unit matrix, the $d(x, y)$ is the Euclidean distance. Also, when A is the variance in the Gaussian training set, the distance definition becomes a special variation of the likelihood defined in Eq. 4.3.

Furthermore, consider a point x and a trajectory Y composed of a group of points $\{y_1, y_2, ..., y_n\}$. Then similarly, the distance from a point x to the trajectory can be defined as the minimal point-to-point distance, i.e.,

$$d(x, Y) = \min_i d(x, y_i) = \min_i ||A(x - y_i)||^2 \tag{4.10}$$

LAPD: Distance Between Trajectories

The distance between trajectories shows the similarity between two video clips. In previous sections we compute the point-to-point and point-to-trajectory distance. In this section we compute the inter-trajectory distance in a similar way.

To compute an effective distance which gives reliable similarity representation, the Luminance Aligned Projection Distance (LAPD) is proposed based on the following idea: given a pair of trajectories, finding an optimal matching offset h in the longer trajectory started at where the afterwards average point-to-point distance is minimized.

Suppose trajectories are denoted as $x_{j,k}(t)$, for curve j belonging to action class k, and for each class, there are $j = 1..n_k$ curves, t is the frame index which varies from 1 to n. Then

for an unknown video clip trajectory $y(t)$ with m frames, and a known action video clip $x(t)$ of n frames, assuming $m < n$, the LAPD between x and y is defined in Eq. 4.11.

$$d_{\mathrm{LAPD}}(x, y) = \min_{h} \frac{1}{m} \sum_{t=0}^{m-1} ||A(x(t+h) - y(t))||^2 \tag{4.11}$$

Let us assume that we have K action classes and each has $j = 1..L$ training clips, $x_j^k(t)$. Then for an unknown clip $y(t)$, recognition can be implemented based on the minimum LAPD,

$$k^* = \arg\min_k \min_j d_{\mathrm{LAPD}}(y(t), x_j^k(t)) \tag{4.12}$$

For each incoming query video clip C, we calculate the distance between C and those clips in the training set which contains N training samples. An $N \times 1$ distance array D_N is generated with different action labels. After sorting the entries of the distance array D_N, the M smallest values are selected from the training distance array with the corresponding action labels. Therefore, given the first M labels, voting is applied to count the number of labels for each action class. The final decision is based on the label with the most votes.

In this method, we focused on finding the relationship between the two trajectories in subspace. Instead of computing the distance directly, a best matching point is firstly found by trying every possible offset in matching the two trajectories which minimize the distance between them. The spatial information is maintained in the trajectory coordination, while the temporal features are kept by continuous point-to-point matching. This processing removes the effect brought by noise, and proved to be robust against some other factors such as scaling and background changes.

4.2.5 Spline Approximation and Graph Embedding

In this section, we are taking an appearance modeling approach based on graph embedding [32]. Gesture video clips are modeled as trajectories in the scaled appearance space. Then the trajectory Spline modeling and re-sampling and prototype merging are performed to reduce the problem size, i.e., the affinity graph size, for the embedding part. Gesture recognition is achieved by aligned projection distance metric in the appearance space from spatio-temporal graph embedding.

The Locality Preserving Projection (LPP) converted the affinity matrix to a transformation A by decompose eigenvalues as in Eq. (2), which is used as localized subspace modeling in the later step.

$$xLX^T\alpha = \lambda XWX^T\alpha \tag{4.13}$$

However, the problem complexity is directly tied to the size of the affinity matrix W. Direct embedding of all training dataset points is not feasible. On the other hand, in the dataset each gesture action class is represented by many action video clips, and each clip with many frames. For the limited number of data points that can be used in the graph embedding appearance model, an efficient solution of allocating these points among training video data set is required.

On the other hand, there is usually a lot of redundancy in the training dataset. Video clips with same action are captured multiple times, which introduce many repeating statistical

information. Based on the representation in previous section, we tested on some specific trajectories and get the following figures, as shown in Fig. 4.3 and 4.4 respectively.

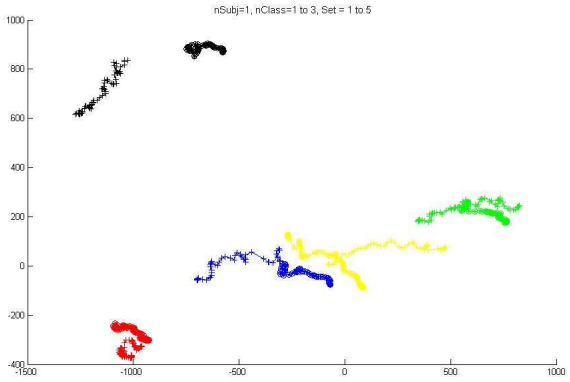

Fig. 4.3: Observation of Trajectories: Case 1

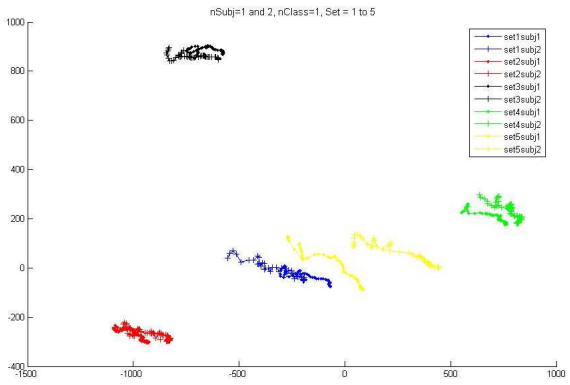

Fig. 4.4: Observation of Trajectories: Case 2

In Fig. 4.3 we plot the case that two different hands are doing the same action under different luminance conditions. We are using the first two dimensional results obtained from dimensionality reduction in the previous steps. From the figure we can see that the illumination affect the result very much. The different illumination will enlarge the within class distance, while the difference in subject is not that much.

Similarly, in Fig. 4.4 we plot the situation that same hand doing different actions in various

illuminating background. We can see from the figure that the actions in different illuminating background (plotted in different colors) is faraway with each other. Therefore, a lot of redundancy can be detected within the dataset.

With the observations above, a prototype merging method with spline approximation is proposed to address the issue. For each action class, only L prototype trajectories are allowed, and for each trajectory, only M data points are allowed to meet the total data points number constraint. The L prototype is obtained by trajectory merging, which is similar to the Vector Quantization(VQ). But instead of operating on points, curves are merged with the LAPD metric, which searches a matching temporal offset among sampling points on two trajectories. The detail of technique is already introduced in previous section and we rewrite the formulation as follow,

$$d(X,Y) = \min_{h\in[0,n-m+1]} \frac{1}{m} \sum_{k=1}^{m} \|X(k+h) - Y(k)\|^2 \tag{4.14}$$

where X and Y are two different trajectories and m and n are the number of frames in X and Y. The merging is achieved by align two trajectories at the LAPD offset, merging matching points by their average. A toy example is plotted in Fig 4.5. In each round the closest two trajectories is merged into one and finally only L curves are left. Suppose two curves of n-frame $X_0(t)$, and m-frame $X_1(t)$, with $n > m$, then the merged trajectory $Z(t)$

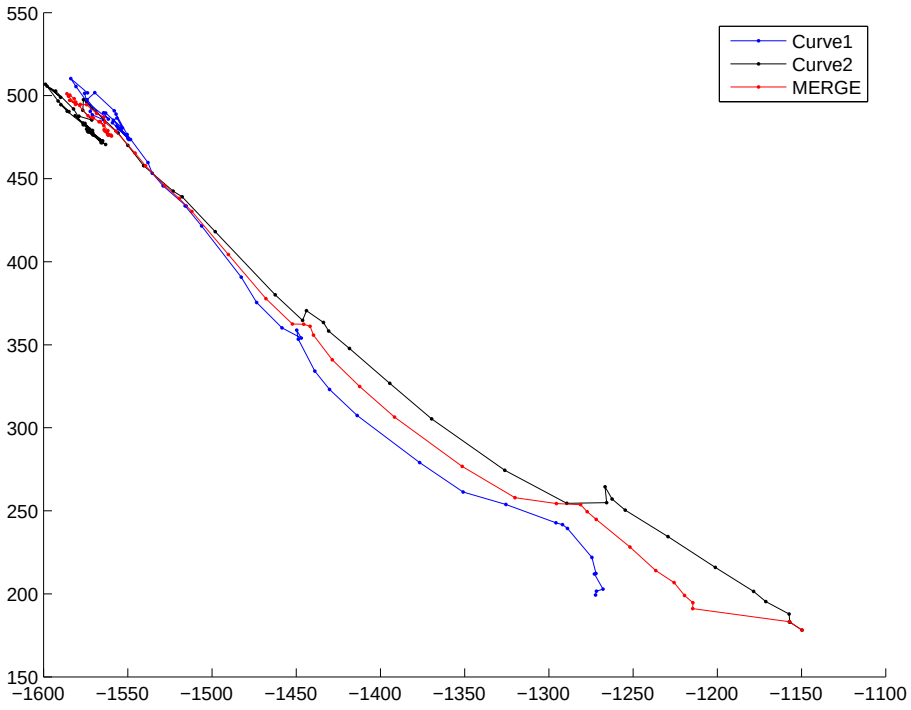

Fig. 4.5: An example for Prototype Trajectory merging

can be given as,

$$Z(t) = \begin{cases} \dfrac{X_0(t+h^*) + X_1(t)}{2}, & h^* \leq t \leq h^* + m - 1, \\ X_0(t), & \text{otherwise} \end{cases} \tag{4.15}$$

therefore, we can have,

$$h^* = \arg \min_{h \in [0, n-m+1]} \frac{1}{m} \sum_{k=1}^{m} \|X_0(k+h) - X_1(k)\|^2 \tag{4.16}$$

Spline [92] has a rich history in signal processing and in this work, we use spline approximation to model video action trajectories. Training data points are obtained from resampling

the splines to meet the constraint on M and find better representation by equal distance re-sampling and then sampling at equal curve length. For example, if M points are to be sampled from a trajectory, $X(t)$: $t \in [0,1]$, we have the n-point approximate curve length of X at point k by,

$$Length(X) = \sum_{k=2}^{n} \|x(t_k) - x(t_{k-1})\|, t_k = k/n \tag{4.17}$$

Therefore, the curve length is given by $Length(X) = Length(t_n)$, and those M curve sampling points can be found by equal curve length sampling, i.e., the m-th point is found by,

$$\arg\min_k \|Length(t_k) - \tfrac{m-1}{M} Length(X)\| \tag{4.18}$$

An example of M=7 is plotted in the Fig. 4.6.

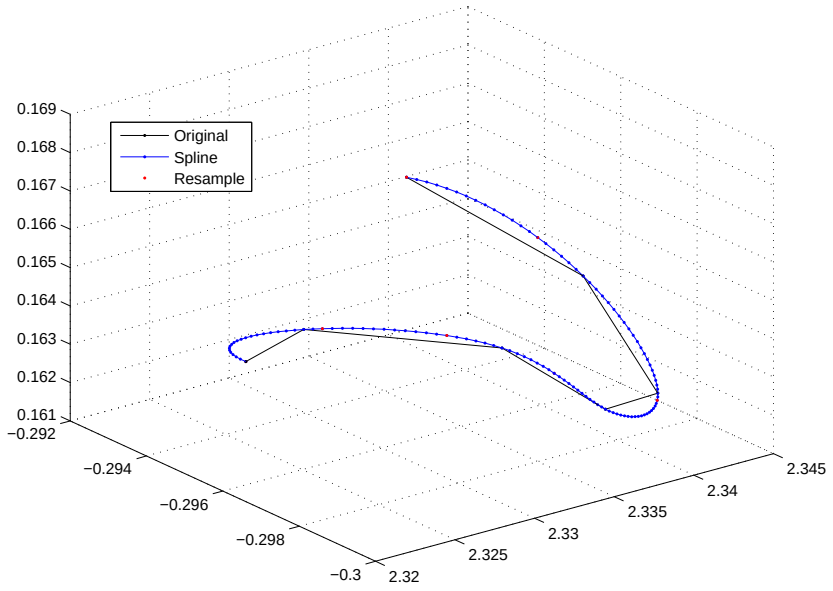

Fig. 4.6: Spline and Equal Distance Resample

Finally the size of affinity matrix for LPP will become an acceptable number for processing by choosing a proper L and M. Furthermore, we can control the kernel size parameters to reflect different affinities among intra and inter class data points on re-sampled training curves, i.e., L training curves for each action class with M data points each,

$$W_{j,k} = \begin{cases} e^{\alpha d(x_j,x_k)}, & \text{Case(i)}, \\ e^{\beta_1 d(x_j,x_k)}, & \text{Case(ii)}, \\ e^{\beta_2 d(x_j,x_k)}, & \text{Case(iii)}, \end{cases} \tag{4.19}$$

where parameter α is used for (i) intra-trajectory data points affinity, β_1 and β_2 are used for (ii) intra-set, inter-trajectory points affinity and, (iii) inter-set, inter-trajectory affinity, respectively. *Set* here refers to trajectories belonging to the same gesture class but different lighting conditions as in the Cambridge Hand Gesture dataset. By introducing these parameters, the subspace learned from the training set will be more reliable since trajectories belonging to the same gesture class can be closer than those belonging to a different class.

With these M trajectory points in each of L curves per gesture set, a graph affinity matrix W is obtained by Eq. 4.19 . The choice of parameters M and L reflects a tradeoff between the computational complexity and model effectiveness. The choice of kernel sizes and allows for flexibility in re-shaping the local distances among trajectories.

4.3 LAPD and Graph Embedding Solutions

4.3.1 LAPD Solutions

In this section we introduce our solution of DLFT and LAPD separately. The KTH data set, which is tested in [48] and [81], is used as an example to illustrate the proposed method in detail.

In this solution, given a query clip, the distance between the query and each training trajectory is pairwise computed by Eq. 4.14, and a distance array is generated for each query. The distance from the query to each action category is computed and a histogram is shown in Fig. 4.7. The decision is made by selecting the smallest distance, which is quite obvious in the figure.

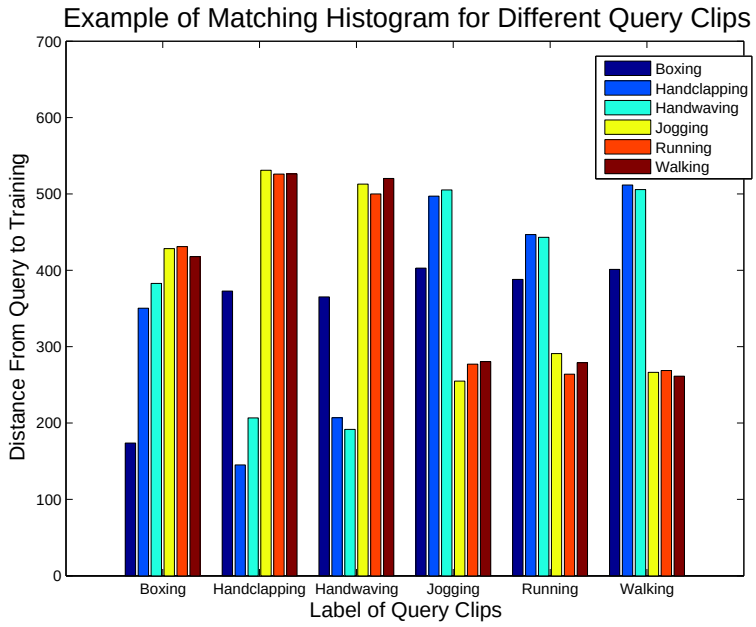

Fig. 4.7: Examples for Differential Trace for Different Actions

The classification is based on a simple K Nearest Neighbor (KNN) classifier. The training clips projected in LPP or LDA space are modeled as a 6-class Gaussian mixture, and the classification is done by assigning maximum likelihood action labels. Experimental results and discussions are presented in Section 4.4.

4.3.2 Graph Embedding Solutions

In this section we present our solution in spline approximation and graph embedding. The processing includes trajectory merging, spline approximation, resample, and finally there is an affinity metric construction session.

By trajectory merging we focused on reducing the number of trajectories in the training dataset, trajectories that represent the same action class are iteratively merged until termination, and then spline approximation is applied to smooth the merged trajectories.

The parameter L and M is highly application dependent. L is the number of trajectories left per class. The value of L is a reflection of within class statistical information. Too large L will not reduce the number of training trajectories very much, while too small L may cause the left trajectory not representative. In our experiment we manually set $L = 3$ and we plot a merging example in Fig. 4.8.

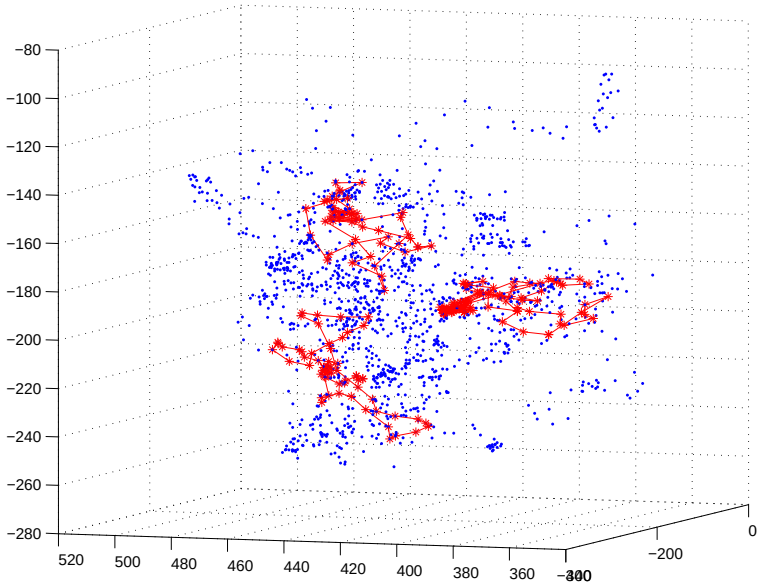

Fig. 4.8: Examples for Differential Trace for Different Actions

4.4 Experiments

We have tested our methods on two different datasets, the *KTH human action dataset* and the *Cambridge Hand Gesture dataset* for human action recognition and hand gesture recognition separately. These datasets cover variations in appearance, illumination, background and spatio-temporal cues. Besides proposed approaches, we also tested the Dynamic Time Warping method on the dataset above, and compare with some results obtained by other approaches in the recent literature.

For all the datasets, our implementation is based on the leave-one-actor-out setting, where the classifier is trained using all video sequences except those corresponding to the actor in the test video. This processing is repeated many times until each video has been treated as the test video.

4.4.1 KTH Human Action Dataset

Dataset Introduction

To test the developed algorithm, we use the human activity data set from [81], which contains 6 human actions, *'boxing'*, *'handclapping'*, *'handwaving'*, *'jogging'*, *'running'*, and *'walking'*. Actions are performed by a total of 25 subjects in 4 different settings:

S1: outdoors;

S2: outdoors, with camera zooming;

S3: outdoors, with different clothes on;

S4: indoor.

For each setting, each action has 4 video clips, with each segment's start and end frame number listed as a ground truth file. Each setting will have $4 \times 25 \times 6 = 600$ actions, and the data set comprises of a total of $2,391$ clips, with a small number of entries missing.

The video clips are of 160×120 pixel resolution, and in processing stage, we down convert the sequence into 20×15 icon image sequences for trajectory modeling. Some examples from the action clips in [81] are plotted in Fig. 4.9. From left to right, the top row actions are, walking, jogging, and running, and the bottom row actions are, boxing, hand waving and hand clapping.

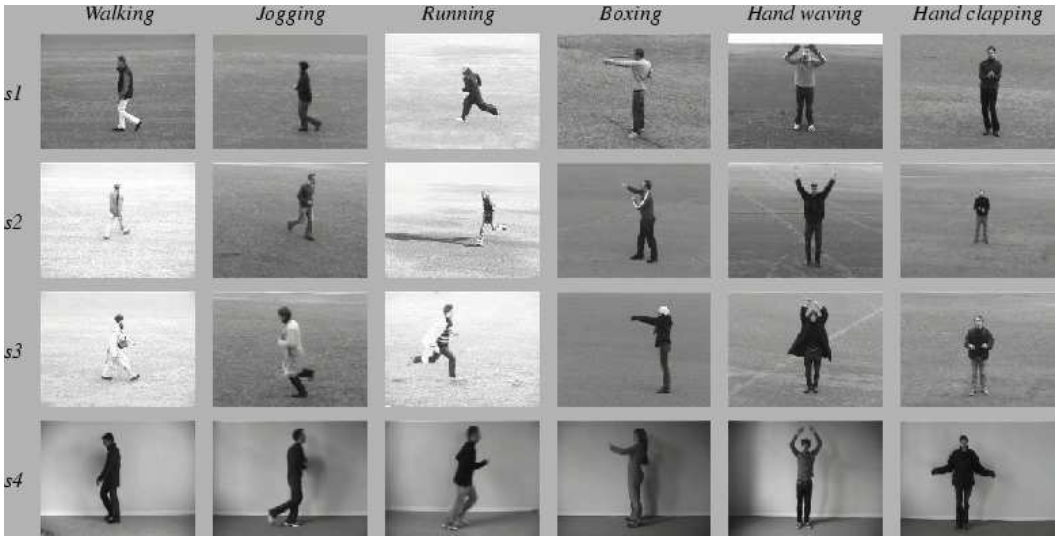

Fig. 4.9: Sample frames in KTH human action dataset [48]

Simulation Result

In the pre-processing stage, we down-sampled the original video frame to 20×15 icons and applied a global PCA to reduce the number of dimension to 64. Thus the video sequences are represented by trajectories in $\mathfrak{R}^{64}$.

For the LAPD approach, the 64-dimensional feature is used again after pre-processing for dimensionality reduction. We test two different scenarios, unsupervised learning and supervised learning respectively. The global PCA is an unsupervised subspace learning together with dimensionality reduction and extraction of statistical information. We use this unsupervised subspace for computation of distance between the query clip and the training ones, and apply a KNN classifier to complete the classification. A confusion matrix is utilized to describe the performance of pattern recognition. The numbers in the diagonal line is the percentage for correct class, while the other place is the wrong cases. The confusion matrix of unsupervised LAPD is plotted in Fig. 4.10, with an overall accuracy of 78.9%.

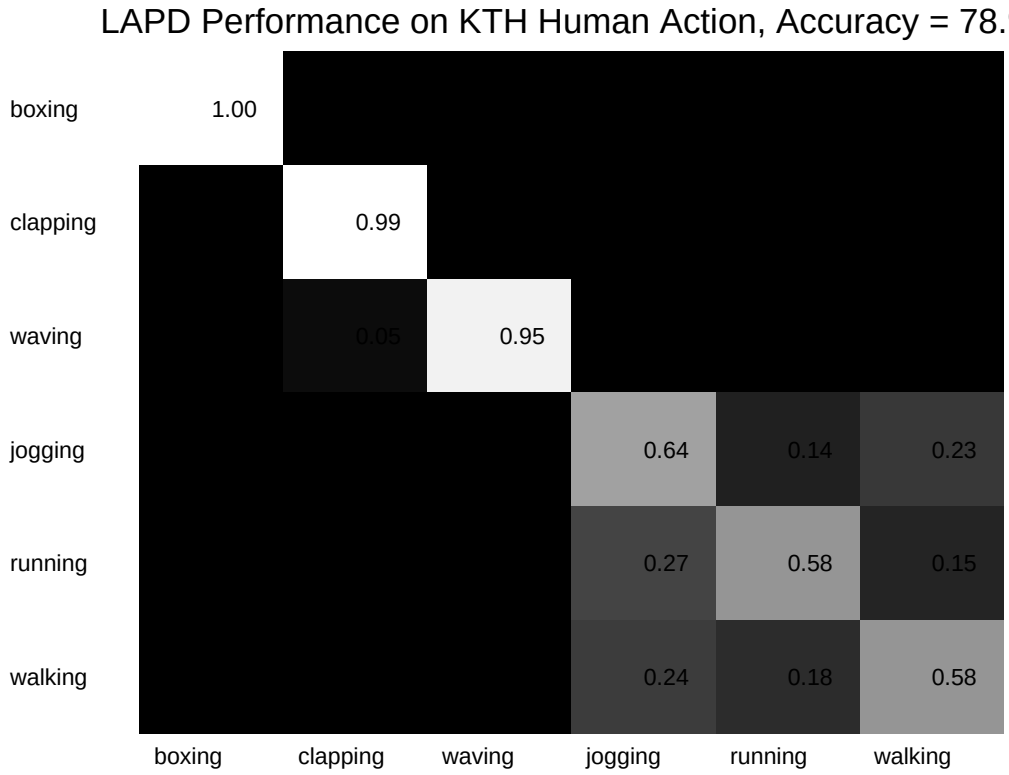

Fig. 4.10: Performance on KTH human action recognition by unsupervised LAPD

For supervised learning, a LDA subspace learning is later applied to discriminate the different patterns and to further reduce the number of dimension for easier LAPD computation. The result is shown in Fig. 4.11, with an overall accuracy of 81.3%.

Our spline approximation method is also implemented in the experiment to prove the robustness. Given the 64-dimensional features extracted in preprocessing step, the trajectories are merged one by one until there are only a few discriminative curves per action class. Spline approximation and resampling are then applied to smooth the curve for better representation. Finally graph embedding is applied and KNN classifier is utilized to categorize the query trajectory into different pre-defined action classes. The confusion matrix is shown in Fig. 4.12.

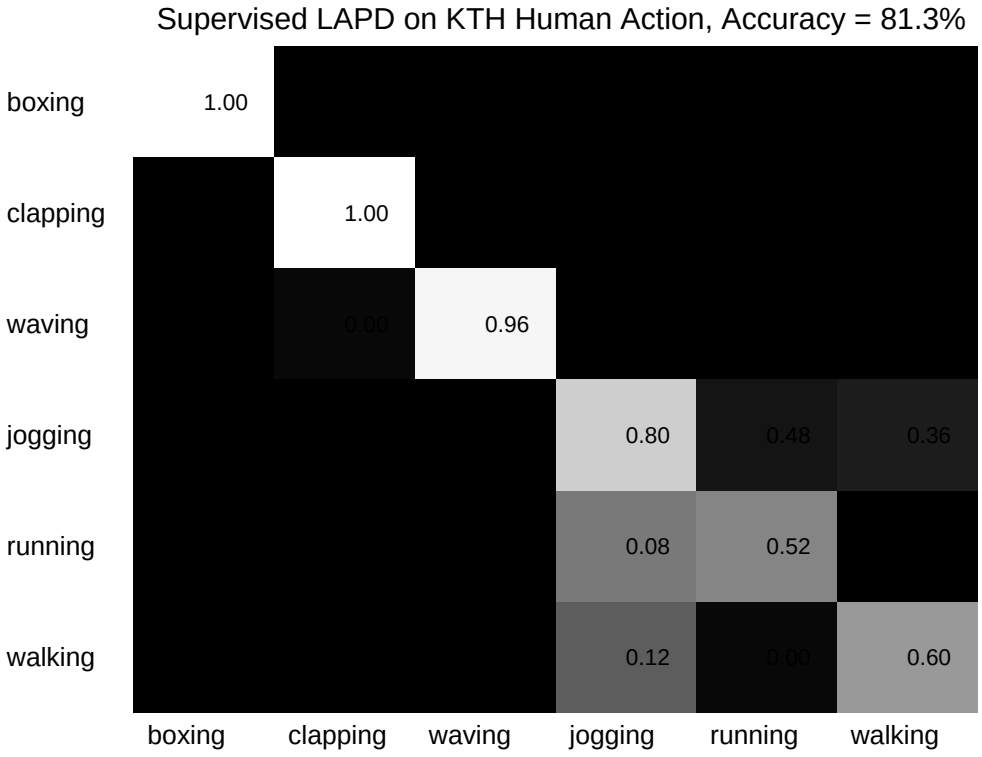

Fig. 4.11: Performance on KTH human action recognition by supervised LAPD

4.4.2 Cambridge Hand Gesture Dataset

Dataset Introduction

To demonstrate the robustness and versatility of the algorithm, we also tested another dataset, the *Cambridge Hand Gesture Dataset*, which is composed of 900 image sequences with 9 different hand gesture classes [43]. These classes are defined by 3 primitive hand shape: *Flat (F)*, *Spread (S)* and *V-shape (V)*, and 3 primitive motion directions: *Leftward (L)*, *Rightward (R)* and *Contract (C)*. There are totally 9 gesture classes by combining the two factors above. Each class contains 100 image sequences, with 5 different illumination cases. An example is shown in Fig. 4.13.

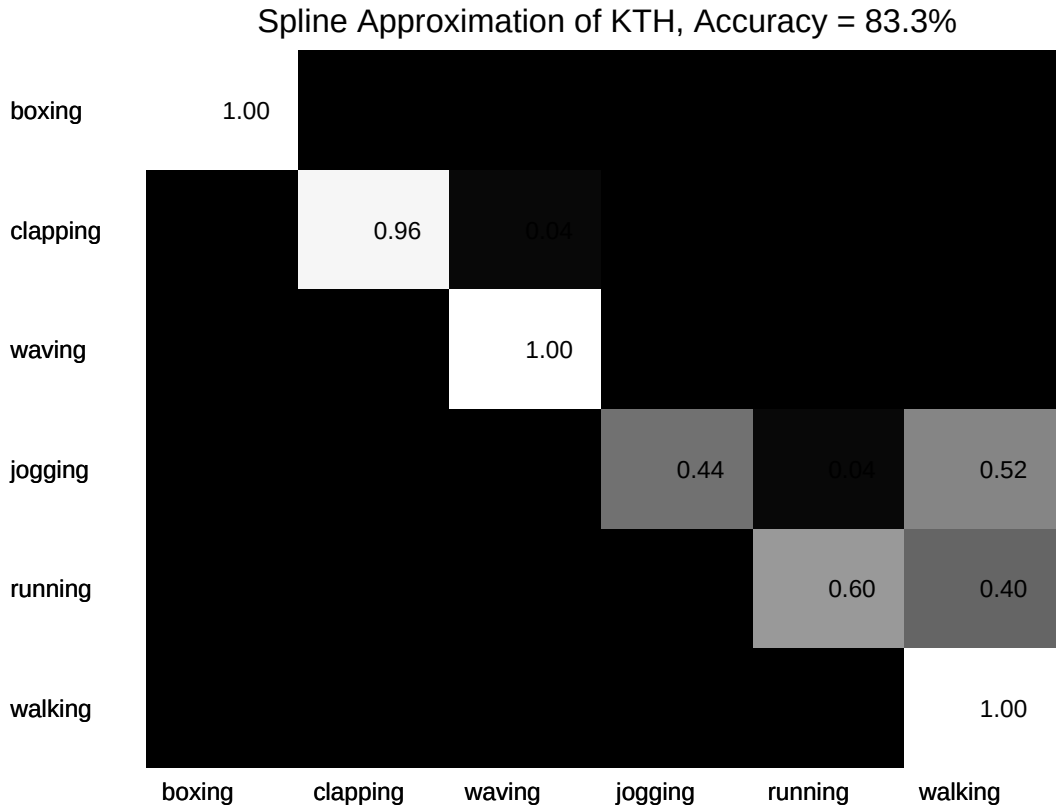

Fig. 4.12: Performance on KTH human action recognition by Spline Approximation and Graph Embedding

Each sequence in the dataset has a different number of images, varying from 37 to 119, and the total number of image is $63,188$. The original image is a 320×240 color image, and we firstly reduce the number of data by converting it to gray image and then down-sample it to a 32×24 icon. We then process these icons with the LAPD and spline approximation approach, respectively.

Simulation Result

The hand is composed by many connected parts and the motion is highly articulated. Without prior information, it is difficult to guess what kind of appearance is contained in

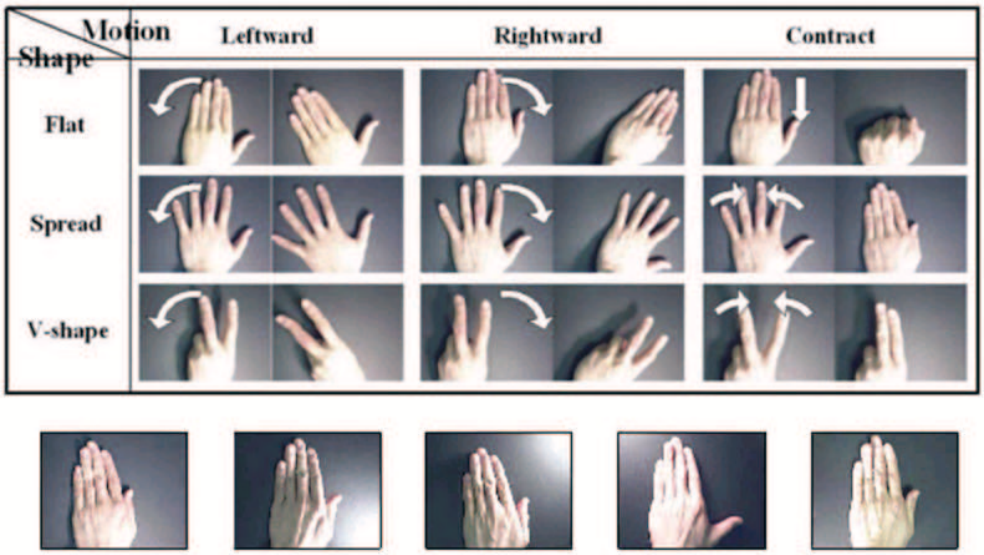

Fig. 4.13: Sample frames in Cambridge hand gesture dataset, (top) 9 gesture classes by 3 motion directions and 3 hand shapes; (bottom) 5 different backgrounds with different illuminations [43]

the image even if it is known to be a hand. To better discriminate the appearance, we keep 32×24 pixels as icons in the preprocessing. For a global dimensionality reduction, a PCA is applied on $\Re^{32\times24}$ and 64-dimensional trajectories are treated as representations for the image sequences for further processing.

In the LAPD approach, we test two different scenarios, unsupervised learning and supervised learning respectively. In both cases we repeat the computation steps as the ones in the KTH dataset, and report the corresponding result in Fig. 4.14 and Fig. 4.15. The numerical recognition accuracy is 80% and 85.1% respectively. For the individual set testing, the comparison results are listed in Table 4.1.The numerical result is comparative to the one in [43] and better than some other results reported in the literature.

The result obtained from spline approximation and graph embedding is presented in Fig. 4.16.

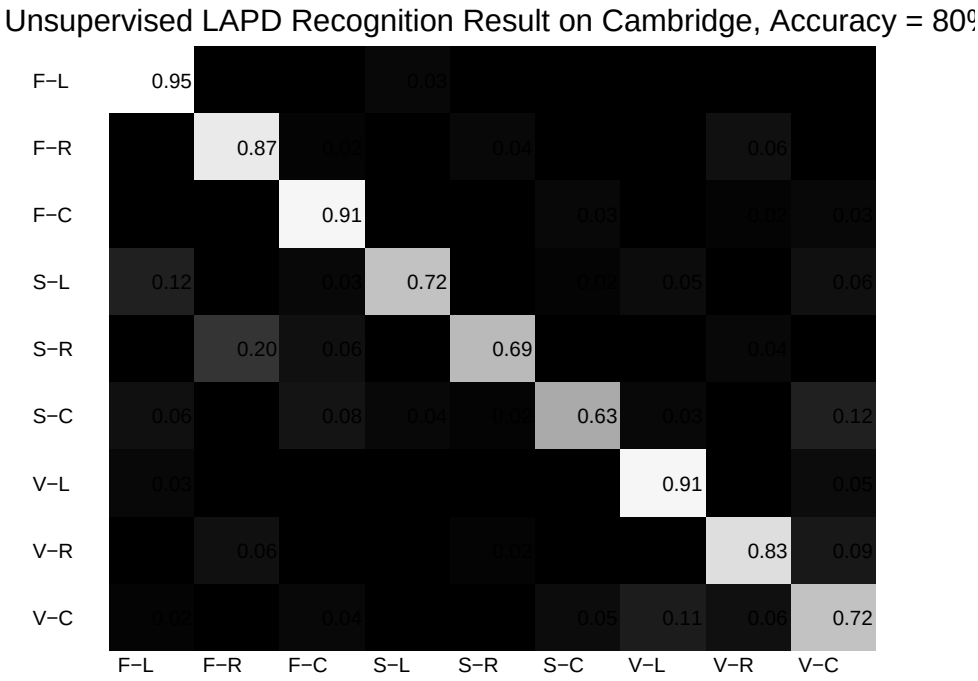

Fig. 4.14: LAPD Performance on Cambridge Hand Gesture Dataset

Table 4.1: Hand Gesture Recognition Accuracy Comparison(%).

Method	Set1	Set2	Set3	Set4	Set5	Average
LAPD(Unsupervised)	83	81	77	80	79	80.0
LAPD(Supervised)	91	85	78	84	87	85.1
Spline	87	76	75	84	87	81.5
TCCA[43]	81	81	78	86	–	81.5
Nieble[66]	70	57	68	71	–	66

The numerical result is about 81.5%.

The object of appearance information in the hand gesture data set is mainly the hand, and the change in the background is a factor to test the robustness of any algorithm. From our KNN result, we found out that in the correct cases, the nearest trajectory in the training set is always in the same class and same background as the query clip. The effect of the changing of the illumination of the background will not influence the recognition performance.

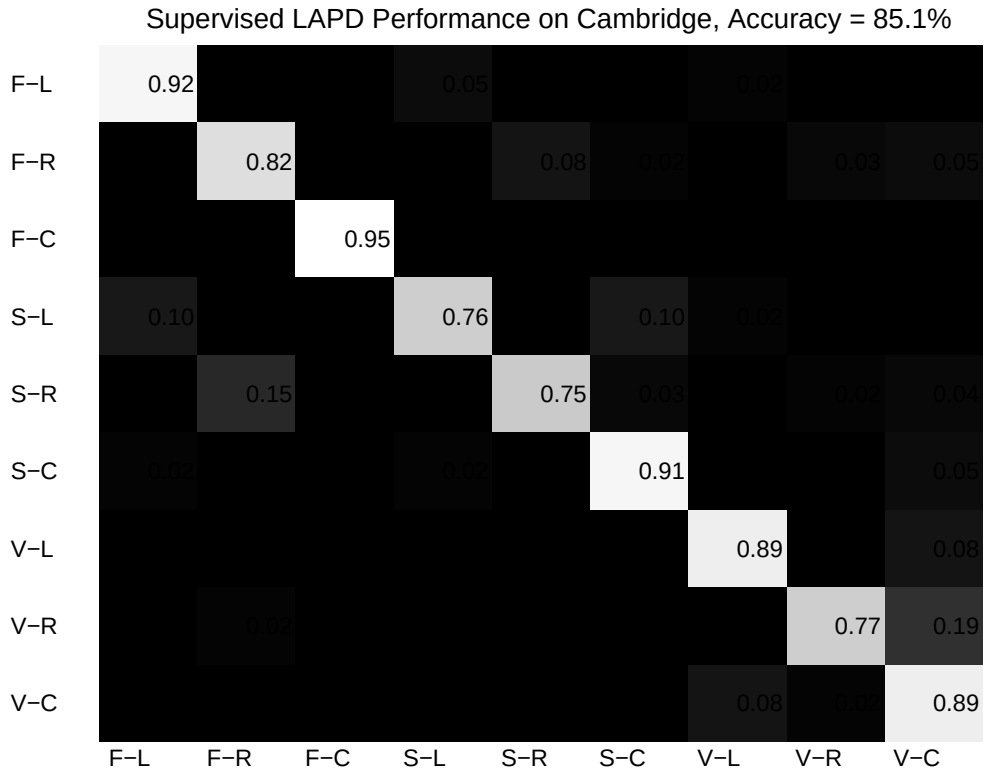

Fig. 4.15: LAPD Performance on Cambridge Hand Gesture Dataset

As shown in both [43] and our LAPD solution, there are several confusions between the class *spread* and *flat*, for either left or right direction. These are mainly due to very little difference in appearance, and the details are lost when sampling the original frames down to a small icon.

4.4.3 Parameter Selection in Experiment

In the experiment, there are some degree of freedom to choose parameter settings. Some of the settings will affect the performance of proposed method. In this section, we present our selection of parameter and give corresponding analysis on the reason to select.

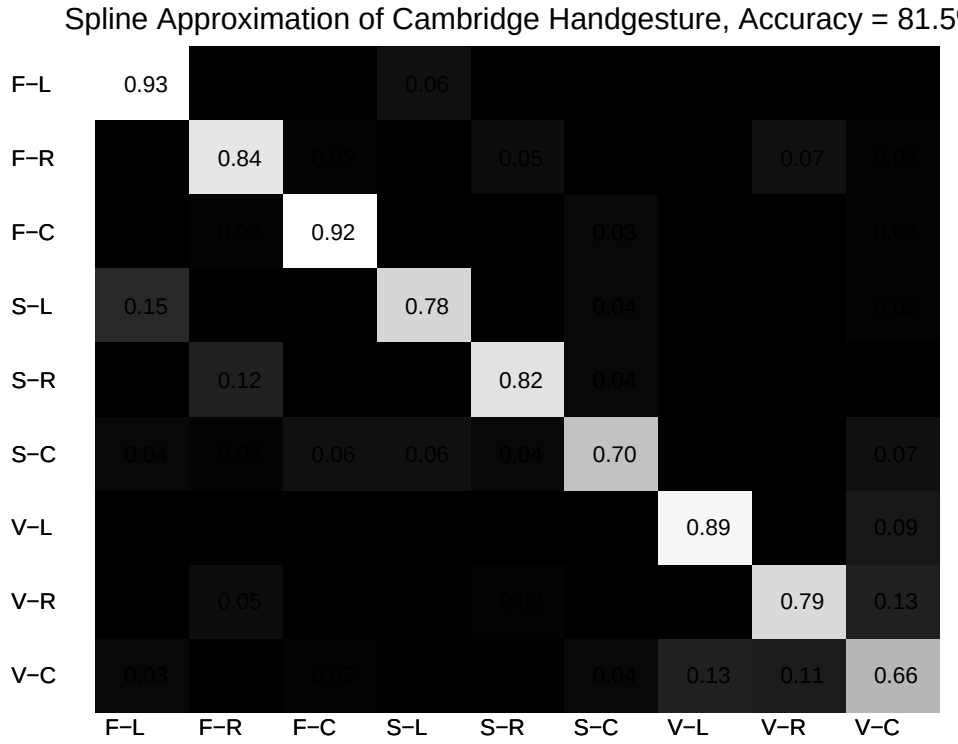

Fig. 4.16: Spline Performance on Cambridge Hand Gesture Dataset

Resolution and Dimensionality

As presented in Fig. 4.1, the preprocessing of the video sequence is composed by two sequential steps, down sample and PCA. In this section we will analyze the related parameter selection in these two steps.

The first step is down sample. Given the original video frame with resolution $W \times H$, the preprocessing down sample it to a smaller icon $w \times h$. In our experiment, we find that the resolution of icon will not greatly affect the recognition accuracy, but larger icon will take longer time for processing. Therefore, according to the size of each dataset, we choose the minimal icon size that can maintain the statistical information. In the KTH human action

dataset, we are using 20×15 icon. In the Cambridge hand gesture dataset we are using icons with resolution 32×24 due to more detail required.

The second step in preprocessing is PCA for dimensionality reduction. This operation is a tradeoff between the speed and accuracy. More dimension will preserve more useful information for discrimination, but slow down the recognition processing as well. The performance will also be drastically degraded if there is no sufficient features. Therefore a proper number of dimension is very important for recognition performance. In our experiment we are using the following approach, during PCA we have the eigenvalue for each dimension in subspace. The importance of dimension is reflected by the corresponding eigenvalue. Eigenvectors with larger eigenvalue will have more contribution in discrimination. So we sort the eigenvalues in descending order and preserve the first k dimension.

The first k eigenvector contains the most important statistical information in recognition, and in Fig. 4.17 we plot how many percentage of information is preserved of KTH action dataset as an example. The percentage is defined as follow,

$$\rho(k) = \frac{\sum_{n=1}^{k} \lambda_n}{\sum_{n=1}^{N} \lambda_n} \tag{4.20}$$

where λ_n is the nth eigenvalue in PCA computation, and totally there are N eigenvalues. We summarize some important value in Fig. 4.17 and show it in Table 4.2,

From table we can see that more than 97% of information is preserved in the PCA when

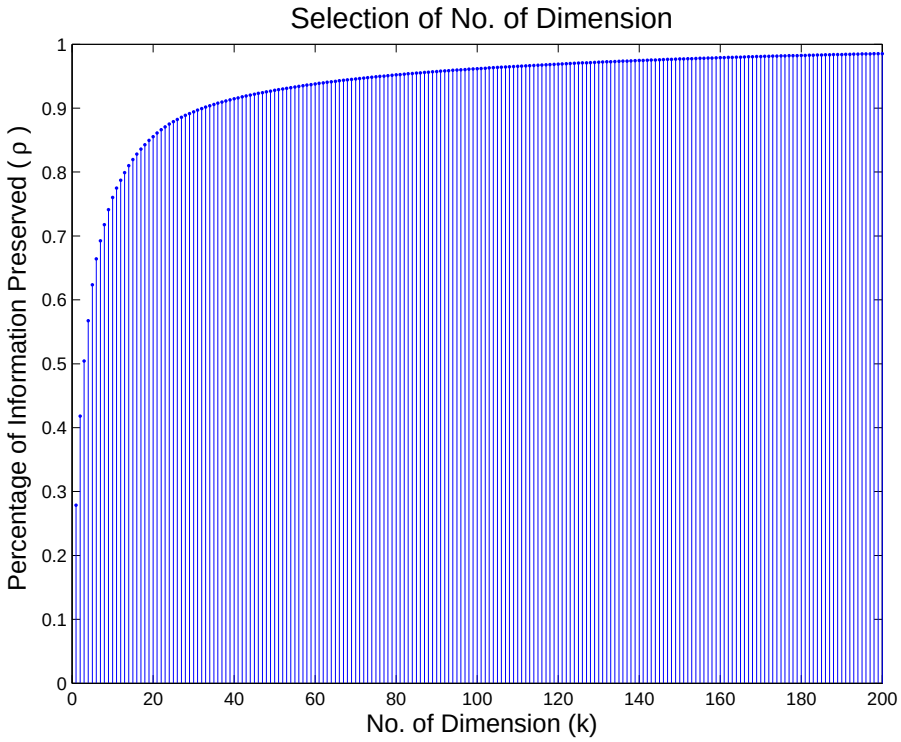

Fig. 4.17: Parameter selection in PCA for dimensionality reduction

Table 4.2: Selection of number of dimensionality.

Number of Dim.	Percentage (%)
16	83.1
32	89.9
64	95.1
128	97.0
256	99.4

64-D features are extracted, which is already enough to achieve a satisfactory recognition accuracy. Therefore in our experiment, we finally reduce the number of dimensionality to 64. Similarly, in the Cambridge hand gesture dataset, we adopt the same setting for dimensionality reduction.

Table 4.3: Parameter selection in KNN classifier.

Value of k.	Recognition Accuracy (%)
1	73.3
3	75.0
5	78.9
7	77.8
9	75.0

Classifier Parameters

As presented in previous sections, we are using KNN classifiers to decide the query label in recognition in our proposed LAPD approach. In the classifier, k nearest neighbors are used as reference, so the number of k is a parameter which can be freely set in the experiment. Too small k value will cause unreliable recognition result, while too large k will result in unrelated samples in the nearest neighbor set. Therefore, a proper k value is necessary to achieve high recognition accuracy.

Since the query is classified by a majority vote of its neighbors, the number of k is better to be odd. In 4.3, we list the possible values of k and their corresponding recognition accuracy on KTH human action dataset.

From the table we can see that the performance is slightly different. The performance is best when the value of k is set to 5. Therefore we select $k = 5$ as the parameter for KNN classifier. The result is similar in other dataset, so to be fair, in our experiment we are all using $k = 5$ for the classifier setting, in all the dataset.

Parameters in Spline Approximation

Compared with LAPD approach, the spline approximation has more flexibility. In the curve merge stage, there are two parameters can be tuned, the number of trajectories kept per class and the number of points kept per trajectory.

By iteratively applying curve merge on the original trajectories, there are less and less trajectories per class. However, the number of trajectories we keep will also affect the recognition performance. Too little curves may not be representative enough to statistically describe the action, while too much curves will slow down the online recognition speed. In our experiment, there is 100 trajectories per class. Before the merging, a LAPD distance metric is pre-computed, mean and variance of the distance can be obtained from the metric and utilized to control the processing of curve merging. Then the merging is started from the trajectory pair with minimal LAPD and terminated until the minimal LAPD of left trajectory achieves the mean LAPD in the distance metric which is pre-computed. However, this may cause the imbalance between different action classes. To address this issue, we observe the merging process and manually set the number of trajectory per class to 15, which is closed to the criteria mentioned above.

Another parameter to set is the number of points per trajectory, The meaning of this parameter is similar with the previous one: too many points will cause complicated computation during the online matching, while too few points will lose the statistical information in the trajectory, which will represent the action class. In our experiment, we reference to the number of frames per video clip, and set the number of points per trajectory to 50, which is closed to the median of all trajectories.

Table 4.4: Parameter selection in graph embedding.

Value of α	Recognition Accuracy (%)
0.25	81.0
0.5	82.1
0.75	81.3

Besides the spline approximation and graph embedding, there should also be parameter selection issues in graph embedding section. According to the affinity matric construction process, the parameter α, β_1 and β_2 will also result in different recognition accuracy.

The relationship for these graph embedding parameters is correlated with each other. It does not make sense to set these parameters in a very large range, so we assume that all of them are in the interval $[0, 1]$. Basically they are a description with different emphasis. The parameter α is used to describe the within class affinity, so we set it as the basic value in the processing. In our experiment, we test three different value, $\alpha = 0.25$, $\alpha = 0.5$, and $\alpha = 0.75$ respectively. For a fair comparison, we set the other two parameters $\beta_1 = \beta_2 = 0.5$. The result is shown in Table 4.4.

From the table we can see that the parameter α will slightly affect the recognition accuracy. In our experiment it is found that the recognition accuracy is best when α is set to 0.5. Therefore, we use this value in our other experiment.

The selection of parameter β will be more trivial, in our experiment we just investigate the relationship between two β. According to previous result, we set the value of α to 0.5, and test three different scenarios, $\beta_1 = 2\beta_2$, $\beta_1 = \beta_2$ and $2\beta_1 = \beta_2$ respectively. The numerical result is listed in table 4.5.

Table 4.5: Parameter selection in graph embedding.

Value of α	Recognition Accuracy (%)
$\beta_1 = 2\beta_2$	81.1
$\beta_1 = \beta_2$	82.1
$2\beta_1 = \beta_2$	81.6

According to the experiment result, we find that the numerical performance is best when $\beta_1 = \beta_2$. The physical meaning for this result is that the inter trajectory affinity should be uniform, no matter which set it belongs to.

4.4.4 Result Discussion

The proposed LAPD and spline approximation approaches are mainly aiming at keeping both spatial and temporal features, which are crucial for action recognition in video sequences. In the KTH dataset, the 6 classes of human action have different spatio-temporal content: the three hand actions have more spatial information than the temporal one, while the body action mainly consists of temporal features.

For LAPD, the dominant feature for recognition is the distribution of frames in the transformation domain. The aligned method is designed to address the difference in video durations. As is proved in the experiments the performance is quite accurate, even with a small number of training samples.

For spline approximation, the dataset is greatly simplified. There are redundancy in the database, and the discriminative information can be maintained during the simplification.

The spline approximation and resample is applied to smooth the trajectories. Graph embedding is used to provide intra-class and inter-class relationship, in both spatial and temporal domains.

Both methods above still have a common advantage: they are both content independent and only very simple pre-processing is needed. They can be applied to both human action recognition and hand gesture recognition. Theoretically the algorithm itself, is not dependent on the video content, and for every possible video sequence, the method can offer good recognition accuracy as long as the classes are defined clearly.

4.5 Summary

In this chapter, we proposed two approaches for video content recognition without object level learning. The LAPD solution method we proposed is to utilizing aligned projection distance approach. We still use the trajectory to represent the video clip and calculate the distance between every two clips to define how similar they are to each other. The spline approximation and graph embedding method is to apply statistical tools to simplify the computation. The results show that the recognition accuracy is comparable or better than other techniques in the literature. Regarding complexity, the off-line generation of the distance matrix costs some time, and it will also increase with the size of the training set. However, the on-line recognition for the query is very fast and is also suitable for most applications when the training set is fixed.

Chapter 5

Real-time mobile based Video Recognition (RVR) system

5.1 Overview

In the previous chapter we proposed a global subspace learning approach, luminance aligned projection distance, to solve the human action recognition problem. The video sequences are represented as trajectories in the high-dimensional space, spatial and temporal statistic information are utilized to find the similarity between such trajectories. The recognition accuracy is guaranteed but the problem is not efficiently solved enough. Another issue need to be concerned in the video pattern recognition is the timing complexity. Online applications are popular nowadays with highly demand on efficient processing methods.

In this chapter we focus on improving the speed for recognition, which will be applicable for real-time application, with satisfactory recognition accuracy based on the previous proposed methods. We report and analyze the time consumption in each stage of the processing

method and propose a scheme on how to complete the task in real-time. The proposed approach can compute real-time pattern when capturing query video clips, given subspaces training before the online application starts. In this way, the proposed method can balance the recognition accuracy together with the processing speed.

In this chapter we will use the same video representation processing steps as in the previous chapter. We report the detail time consumption of each processing step and focused on finding and improving the bottleneck. Every step in the approach can be treated separately for timing analysis.

In this problem the timing can be divided into two different types: online recognition and offline training. Basically most applications are focused on improving the online recognition time, while the offline training is not highlighted at all. On the other hand, there is a tradeoff between the online and offline computation, so it is possible to improve the online computation with the cost of offline.

The rest of this chapter is organized as follows. We will briefly introduce large-scale video recognition problem by providing an overview in representation and indexing in section 5.2. In section 5.3 we will present the efficient spatio-temporal modeling method based on the approach proposed in chapter 4. In section 5.4, we present our real time application by a demo. Finally a summary session will be in section 5.5.

5.2 Large Scale Database Processing

In this section we discuss the modeling problem for larger scale database processing. The previous approaches for video feature extraction and modeling may not be reliable and efficient enough. Faster solutions, especially those satisfy the real-time requirement, are highly demanded in the literature

5.2.1 Preprocessing

In the preprocessing stage we convert a video sequence into a trajectory in a high dimensional space. The preprocessing in this chapter follows the one in the Chapter 3, and we briefly repeat the technique as follow.

The preprocessing starts from converting each video frame into a vector. Originally each frame is represented as a RGB metric with an M by N by 3, where M by N is the original resolution. Since the data is too large to efficient processing, a downsample is applied to reduce the original image down to a small icon, with size m by n. Then the small icon is resized as a vector as 1 by mn, therefore a video sequence with k frames can be represented as a matrix k by mn.

A global PCA is applied to further reduce the number of dimension in the video processing. The number of dimension per frame will be reduced down to d after the computation. Up to now, one single video sequence with k frames is represented as a k by d matrix, and therefore a dataset composed with N video frames will become N vectors in the d dimensional space.

Table 5.1: Processing time for each computation step.

Computation	Time Consumption(s)
Preprocessing	171
Global Learning	50
Dist Metric Comp.	6407
Curve Merge	26.4
Spline Approximation	81
Affinity Metric	70

5.2.2 Current Timing Analysis

As the training data set grows in size, it is a great challenge to model this local appearance information. In this chapter, we focus on speeding up the processing by rearrange the order of computation. We analyze the time consumption in our proposed method and locate the bottleneck of processing speed. Then we focused on reducing the processing time so as to shorten the total computation.

Our proposed approach, spline approximation and graph embedding are based on the LAPD similarity measurement. After the preprocessing, trajectories are merged based on LAPD and then the remained trajectories are smoothed by spline approximation. Then a graph is embedded to complete the offline training. For online matching, the query video clip is preprocessed into a trajectory and then matched to the training clips. We take Cambridge Hand Gesture Dataset for example, and present the time consumption as follow.

In the table we summarize the time consumption for the whole dataset processing. From the table it is easily observed that the distance metric computation and update is the bottleneck of time consumption, which is also the target we are improving. In our proposed method,

once we complete the merging of two trajectories, the distance metric is updated. This will cause the computation of millions pairs of LAPD, which cost a lot of time.

5.3 Efficient Learning and Real-time Matching

In this section we propose a tree-based trajectory merging method, which can greatly reduce the computation time with little degradation of recognition accuracy. In the proposed method, we complete the curve merging in a parallel style. Once we compute the LAPD distance metric, for the first half which are closed with each other, we complete the merging for multiple pairs simultaneously. In this way, half of the trajectories will be merged, and the other half remains unchanged. one trajectory in four will be removed in each merging round. Therefore the totally processing time is greatly reduced.

5.3.1 Efficient Curve Merging

In our previous method, only one pair of curves is merged in each round, and the LAPD metric is updated. The computation of LAPD metric is very time-consuming, as presented in Table 5.1. Given n training trajectories per action class, $\frac{1}{2}n^2$ LAPD computations are required in computing the distance metric. Therefore the totally number of LAPD is,

$$\sum_{n=L}^{N} \frac{1}{2}n^2 \times nClass \tag{5.1}$$

where N is the number of trajectories per class, L is the number of trajectories left per class,

and $nClass$ is the number of classes. From this equation we can see that the computation of LAPD is not critical, but in our previous solution, the number of computation is too much to handle for real-time application. Therefore in this section we give our solution as follow.

Instead of computing one pair of trajectories for merging, multiple merging are applied in a parallel style, i.e., more than one merging are completed simultaneously. Denote ρ as the percentage of trajectory merged per round, then $q = 1 - \frac{1}{2}\rho$ percentage of trajectory will be left. The range of q is between 0.5 and 1. As an extreme case, all the trajectory pairs are merged and the number of trajectories will become half of original after merging. Compared with Eq. 5.1, we update the new computation analysis as follow.

$$\sum_{n=L}^{N} \frac{1}{2}(qn)^2 \times nClass \tag{5.2}$$

The limitation of computation is converged to $\frac{1}{1-q^2}N^2$ when the number of n is closed to infinity. For comparison, the computation in Eq. 5.1 is about $O(N^3)$. With the growth of dataset, the computation timing complexity is drastically reduced, and the corresponding processing time, together with recognition accuracy, in Cambridge Hand gesture is listed in Table 5.2,

From the table we can see that when the parallelization merging is applied, the processing time will be greatly reduced, together with some accuracy degradation. Basically, the time is greatly reduced when parallelization is applied, but not totally determined by the degree of parallelization, in other words, when more pairs are merged in each computational round,

Table 5.2: The updated processing time for distance metric update by different q

Number of q	Time Consumption(s)	Recognition Accuracy(%)
Original	6407	82.1
0.9	379	82.0
0.8	201	81.7
0.7	141	80.7
0.6	109	78.1
0.5	98	70.4

the time is not further reduced very much. However, on the other hand, the recognition accuracy is degraded due to the inaccurate merging. Therefore, we can find out a balance between the timing and accuracy.

For some online network or cloud computing, the saving in time is of great contribution. The N times saving in time complexity can be translated to the N times larger scale one algorithm can handle. The efficiency of algorithm will be greatly improved once the timing complexity is reduced.

5.3.2 Real-time Recognition

Besides the offline training processing, the timing complexity in the online recognition is much more important in the cloud-based video pattern recognition. A query video clip is uploaded to the cloud and request a class information, the matching should be fast enough to satisfy the user demand. In this section we propose our real-time recognition solution.

Given a query case, there are three potential approaches to solve the recognition problem, i) the client upload the video clip to the cloud and the cloud process the request by applying

a classification algorithm, then return the result; ii) the client download the whole video database from the cloud and complete the classification at the client end; iii) the client extract a uniform representive features with the cloud and send the features to the cloud, the cloud process the uploaded features and compute the corresponding result, and then return it to the client. Intuitively the approach ii is impractical due to the limited bandwidth and the delay is not acceptable by most applications.

For approach i) and iii), it is nowadays more popular to do a feature extraction on the clint side. Given a original video clip, the data will be several megabyte per second. The uploading of original video clip with one hundred frames generally cost tens of seconds in real application, which is not a very satisfactory user experience. Therefore in this section a local feature extraction is applied. In our proposed approach, the video clip is represented as a trajectory and uploaded to the cloud. The processing is not complicated and is easy to be handled in real time. Based on our test result, the preprocessing time is $18ms$ per frame, i.e., 30 frame can be processed in about $0.5s$. As in most video standard, there is no more than 30 frames captured per second, the preprocessing can be completed during the capturing of live video clip, which achieve a real-time requirement.

With the growth of popularity for mobile applications, the video-based real-time processing is also of highly demand today. In our proposed approach, the preprocessing is composed by image resizing and reshaping, which is easy to be completed at the client side, by the mobile devices.

Besides the client side, the cloud computation speed is also influencing the response time, which is an important part of user experience. In this section we also present the detail of

Table 5.3: Detail of Time consumption for Real-time Matching

Processing	Time with Merging	Time without Merging
Subspace Projection	$4.85ms$	$4.85ms$
LAPD Computing	$43.2ms$	$1469ms$
Classification	$1.09ms$	$36.3ms$
Total	$49.14ms$	$1510.18ms$

time consumption in online matching session, as listed in table 5.3.

In the table we also list the time consumption without trajectory merging. It is observed that the timing complexity is greatly reduced by more than 95% after curve merging. The $49.14ms$ online matching speed enables the proposed approach to handle most real-time video processing applications, also makes the proposed method be suitable for larger database processing.

5.4 Real-time Demo

In this section we provide a video demo to show an application with our proposed method. In the demo we proposed to use a smart cell phone as a user and use a personal computer as the cloud. The mobile user will capture an action video clip and then ask the cloud for recognition. The task for mobile devices and the cloud is listed in Fig. 5.1.

As shown in the figure, after live video is captured from the mobile device, there will be a computation partition session. In this session the technique steps will be divided into "processed in cloud" and "processed in mobile". In general, there can be three potential approaches for the recognition: 1) mobile user uploads the captured video to the cloud, and

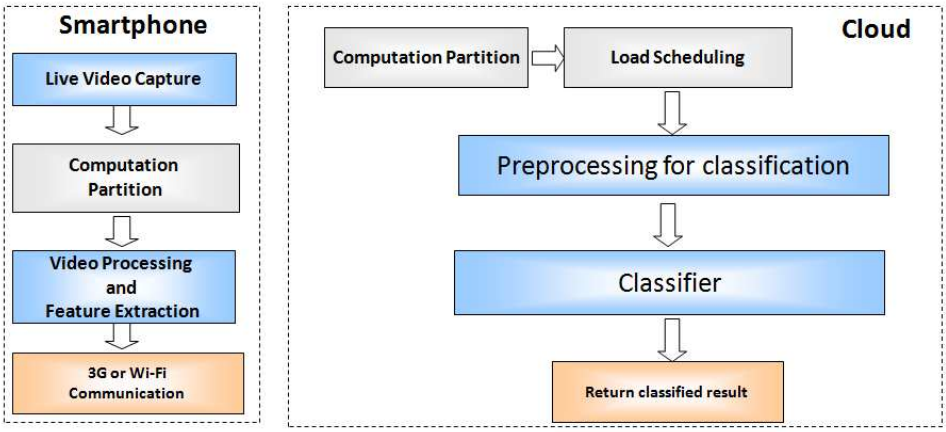

Fig. 5.1: Flow Chart of the Demo

then the cloud will compute the recognition result and reply to the user. 2) the mobile user download the dataset on the cloud and complete the recognition by itself. 3) the mobile device extracts some representative features and upload these features to the cloud, the cloud complete the recognition with the features and send back the result to user. Due to the complicated computing algorithm and limited computation and storage resources in the mobile devices, the first two approaches are proved to be less efficient, in this demo we are using the third approach for processing.

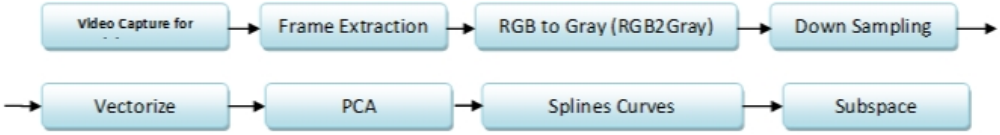

Fig. 5.2: Training Video Processing Procedure

In our processing, the mobile user will do the following jobs, showing in Fig. 5.2: capture live query video clips, and extract the trajectory of video as presented in Chapter 3, a PCA will be applied to further reduce the number of dimensionality, and the trajectory after PCA will be uploaded to the cloud. On the other side, the cloud will receive the

lower-dimensional query trajectory, and complete a spline approximation with resample as described in Chapter 3. Besides, the cloud will also complete pre-processing, training, and matching algorithm with its powerful computational ability and its sufficient storage space.

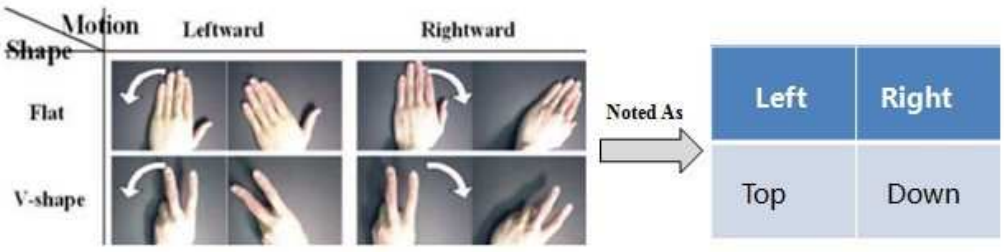

Fig. 5.3: The Semantic Meaning of the Actions

Fig. 5.4:Testing Video Processing Procedure

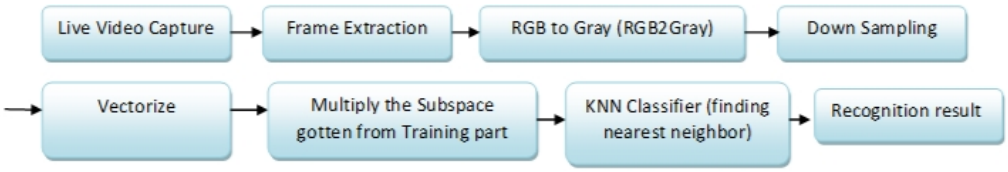

Fig. 5.4: Testing Video Processing Procedure

Fig. 5.5:Testing Video Mobile Implementation Procedure-Method 1

Fig. 5.5: Testing Video Mobile Implementation Procedure-Method 1

Fig. 5.6:Testing Video Mobile Implementation Procedure-Method 2

To simplify the implementation, we take four out of the nine classes in the Cambridge dataset as used in Chapter 3. The sematic meaning of the action is shown in Fig. 5.3. The four actions represent the action “up”, “down”, “left” and “right” respectively. This can be used for some game operation, direction and other related applications. The performance

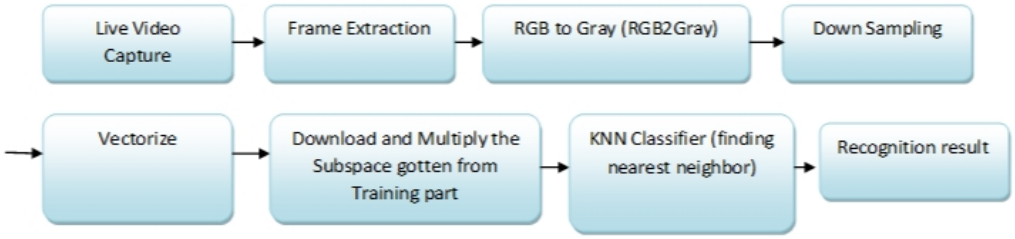

Fig. 5.6: Testing Video Mobile Implementation Procedure-Method 2

is proved to be real-time. A snap shot of video is shown in Fig. 5.7.

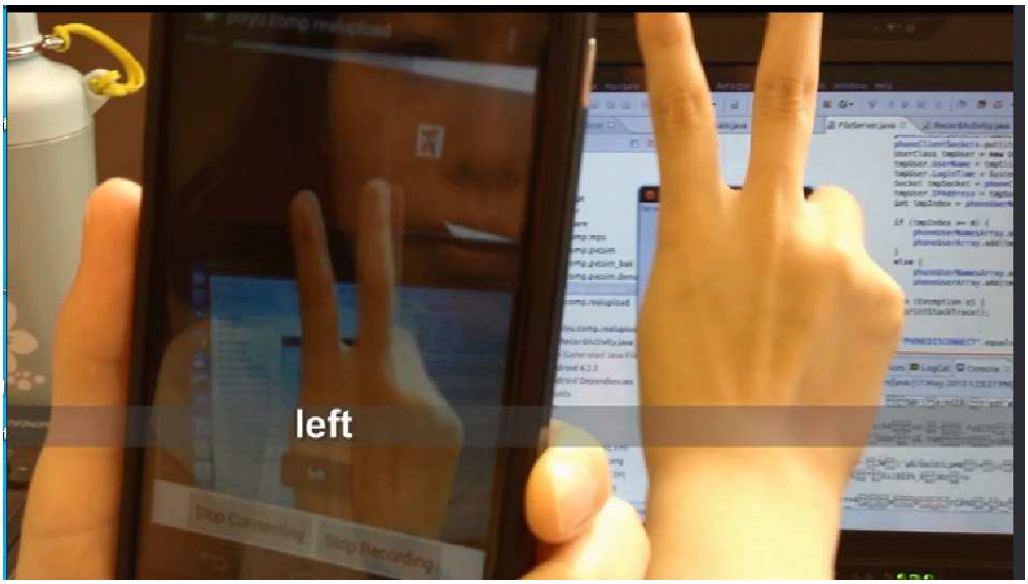

Fig. 5.7: Snapshot in the Real-time Demo

5.5 Summary

In this chapter we investigate the real-time action recognition problem, which is a challenging problem due to the large amount of video data and the difficulties in pattern recognition. We analysis the time consumption of each sub-step and propose some operations for saving computation time without performance degradation. We target on both effective and efficient recognition method, and it is proved to be so in the demo.

Chapter 6

Conclusion and Future Work

6.1 Conclusions

Content-aware video delivery, analysis and understanding are highly demanded for various online application nowadays. There are already lots of related research work and prototype in the literature. The main problem for video processing is that the large amount of data, the transmission will be less efficient if the content is not considered. But on the other hand, the transmission will not require any kind of complicated computation. For another problem, video pattern recognition, the statistical information for understanding is difficult to be detected due to a lot of redundancy in the raw data. On the other hand, for content analysis problem, there are only several potentially answers, which can be easily described. The main challenge in such problem is how to effectively extract the useful information from large amount of data, especially in a very short time for online applications.

In this thesis, we investigated the video delivery and pattern recognition in detail and

improve the QoS and action recognition accuracy by proposing utility-based scheduling method and efficient representation and matching solutions. We target on finding a content-aware solution which is suitable for a global optimal delivery scheme. The urgency of various request is evaluated by utility, and resource is allocated to the highest utility in the long run.

For video pattern recognition, we aim at finding a comprehensive representation for each video and achieve the reduction in data amount together with preserving the critical information. We focused on three different aspects for the processing, effective machine learning, global representation and processing, and timing analysis. Each of these aspects improved the performance either by enhance the recognition accuracy or reducing the computation cost and processing time.

- For the wireless video transmission and delivery problem, we focused on improving the global user utility by evaluate every request in the format of utility. Traditionally, every user is equally treated with each other, and resources are strictly averagely allocated to users. However this scheme is not suitable for some video applications. In video delivery the users requests are different with each other due to different rate on different video content. In this thesis content of the video request is taken into account in this thesis as a reference for the user's condition. We propose a utility as a measurement of the QoS and optimize the global utility. It is proved by experiment that the content-aware method is better than traditional methods.

- For the video representation and processing, we detect the visual information from each pixel value. However, millions of pixels exists in one video frame and it is infeasible to process all of them. Down sample is applied to reduce the number

of pixels and remove the image noises, followed vectorization of video frame and projection to a high dimensional space. In this way each video clip is represented as a trajectory in the space. We then use PCA to globally reduce the number of dimensions for further simplify the data to be processed together with preserving the most important information for each frame. Similar statistical data are found in similar video sequences, which is embedded in the trajectory. We apply high-dimensional curve merge to further reduce the number of trajectories per class, and then a spline approximation is applied to smooth the merged curve. Thus merging operation will align the trajectories with different durations. After that there will be a graph embedding session, which will emphasize or deemphasize the inter-class differences and the intra-class differences. In the online matching stage, every query video clip is preprocessed and a KNN classifier is applied to categorize the query into one of the pre-defined classes. Simulation result proved the effectiveness of both approaches, together with the robustness by using multiple dataset.

- For the real-time application part, we extend our approaches into more detail timing analysis. Both of the offline training and online matching time complexity is analyzed. For offline training, a parallel curve merging scheme is proposed to save the computational time, which drastically reduce the time consumption and accelerate the processing. For online matching, we apply the feature extraction in the client side and upload the features to cloud for categorization. Demo is proveded to prove our result by giving a real-time performance.

The proposed schemes are tested in Ad-hoc wireless video transmission problem and human action recognition problem. However it is not limited in such applications since the algorithm proposed can be generally applied for other kind of media processing, for example

image transmission, text classification, and so on. The exploration of our proposed methodology can create a comprehensive understanding that improves upon the state-of-the-art. Numerical result is competitive or better than the ones in the literature. Robustness of the proposed schemes is also demonstrated by result from various dataset with different challenges.

6.2 Future Research

The work presented in this thesis can be extended in different aspects in the future. We summarize some potential directions as follow.

- First, for the wireless video transmission problem, currently the formulation is not in a close form. The performance is proved to be very closed to optimal but still a small distance from optimal. For more users case, the computation in the server side will become more and more complicated, so there should be some capacity for the problem handling. We can also extend the problem in some other network protocol.
- Second, for our proposed global representation and subspace learning, there is still room for improvement. Various kinds of classifiers could be used to replace the current GMM and KNN one, such as SVM, in our proposed solution. For LAPD approach, besides the spline approximation based processing method, there are still other operations, such as combining the modeling problem with some classical modeling, such as HMM.
- Third, there is always a trade-off between the performance and time complexity. There can be more alternative solutions to our proposed method. The approach should be

highly dependent on application, which will be the mainstream in future.

- Finally, a possible research direction apply our proposed framework in other problems, such as video searching and retrieval.

Bibliography

[1] J. K. Aggarwal and Q. Cai, Human Motion Analysis: A Review, *Computer and Image Understanding*, Vol. 73, No. 3, pp.428-440, 1999.

[2] R. Agrawal and V. Subramanian, and R. Berry, Joint Scheduling and Resource Allocation in CDMA Systems, *IEEE Transactions of Information Theory*, vol. 56, no. 5, pp.2416-2432, 2010.

[3] C. Bahlmann, B. Hasdonk and H. Burkhardt, On-Line handwriting recognition with support vector machine: a kernel approach. In *Proceedings of the 8th International Workshop on Frontiers in Handwriting Recognition*, pp. 490-495, 2002.

[4] P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, *IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI)*, Vol. 19, No. 7, pp. 711-720, 1997.

[5] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", IEEE Journal on Selected Areas in Communications, Volume 18, Issue 3, March 2000.

[6] M. Bilenko and S. Basu and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of 21st International Conference on Machine Learning (ICML)*. pp.81-88, 2004.

[7] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, Actions as space-time shapes. In *Proceedings of 10th IEEE International Conference on Computer Vision (ICCV)*, pp. 1395-1402, 2005.

[8] A. F. Bobick and J. Davis, An Appearance-Based Representation of Action, *Proceedings of IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp.307-312, 1996.

[9] A. F. Bobick and J. Davis, The Recognition of Human Movement using Temporal Templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Vol. 23, No. 3, pp. 257-267, 2001.

[10] U. Brefeld and T. Scheffer. Co-em support vector learning. In *Proceedings of International Conference on Machine Learning(ICML)*, 2004.

[11] VCA usage increase in British Security, BSIA report.

[12] C. Burges, A Tutorial on Support Vector Machines for Pattern Recogntion, *Data Mining and Knowledge Discovery*, Vol. 2, No. 2, pp. 121-167, 1998.

[13] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick and A. Pentland, Invariant features for 3d gesture recognition, *Proceedings of international workshop on automatic face and gesture recognition,* pp.157-162, 1996.

[14] J.P. Campbell, Speaker Recognition: A Tutorial. *Proceedings of the IEEE,* Vol. 85, Issue 9, pp. 1437-1462, 1997.

[15] P. Canotilho and R. P. Moreno, Detecting luggage related behaviors using a new temporal boost algorithm. In *Proceedings of IEEE International Workshop on Performance Evaluation of Tracking Surveillance*, pp. 1-6, 2007.

[16] D. B. D. Cao, O. Masoud, and N. Papanikolopoulos, Online motion classification using support vector machines. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2291-2296, 2004.

[17] TIA/EIA IS-856 CDMA 2000: High Rate Packet Data Air Interface Specification 2000.

[18] S. F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, A Fully Automated Content-Based Video Search Engine Supporting Spatio-Temporal Queries. *IEEE Transaction on Circuits and Systems for Video Technology (CSVT)* , vol. 8, no. 5, pp. 602-615, September 1998.

[19] S. Cherla, K. Kulkami, A. Kale, V. Ramasubramanian, Towards Fast, View-Invariant Human Action Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Wordshops (CVPRW)*, 2008.

[20] C. Cedras and M. Shah, Motion based Recognition: A Survey. *Image and Vision Computing*, Vol. 13, No. 2, pp. 129-155, 1995.

[21] S. B. Davis, P. Mermelstein, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 28, No. 4, 1980.

[22] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, Berlin, 1997.

[23] A. Efros, A. Berg, G. Mori and J. Malik, Recognition Action at a Distance, In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 726-733, 2003.

[24] M. Fleischman, P. Decamp, and D. Roy, Mining temporal patterns of movement for video content classification. In *Proceedings of 8th ACM International Workshop Multimedia Information Retrieval (MIR)*, pp. 183-192, 2006.

[25] N. Ghanem, D. DeMenthon, D. Doermann, and L. Davis, Representation and recognition of events in surveillance video using Petri nets. In *Proceedings of International Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pp. 112-121, 2004.

[26] B. Girod and N. Farber, Wireless video, in *Compressed Video Over Networks*, 2000.

[27] C. Godin and P. Lockwood, DTW schemes for continuous speech recognition, a unified view, *Computer Speech and Language*, Vol. 3, No. 2, pp. 169-198, 1989.

[28] J. Goldberger; S. Roweis; G. Hinton; and R. Salakhutdinov, Neighbourhood components analysis. In *Advances in Neural Information Processing System (NIPS)*. pp. 513-520, 2005.

[29] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as Space-Time Shapes , *IEEE Transaction on Pattern Analysis and Machine Intelligence(PAMI)*, vol. 29, no.12, pp. 2247-2253, 2007.

[30] D. Hardoon, S. Szedmak and J. Shawe-Tayler, Canonical Correlation Analysis: An overview with application to Learning Methods, *Neural Computation*, Vol.16, No. 12, pp. 2639-2664, 2004.

[31] T. Hastie and W. Stuetzle, Principal curves, *Journal of the American Statistical Association*, Vol.84, pp. 502-516, 1989.

[32] X. He, and P. Niyogi, Locality Preserving Projections, In *Advances in Neural Information Processing Systems(NIPS)*, 2003.

[33] H. Hermansky, N. Morgan, A. Bayya, P. Kohn, RASTA-PLP speech analysis technique. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp.121-124, 1992.

[34] S. Hongeng and R. Nevatia, Multi-agent event recognition. In *Proceedings of International Conference on Computer Vision (ICCV)*, pp. 84-93, 2001.

[35] R. Hu, R. I. Damper, Fusion of two classifiers for speaker identification: removing and not removing silence. In *Proceedings of International Conference on Information Fusion*, pp. 429-436, 2005.

[36] J. Hu, B. Ray and L. Han, An Interweaved HMM/DTW Approach to Robust Time Series Clustering. In *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*, pp. 145-148, 2006.

[37] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, "Joint Source Adaptation and Resource Allocation for Multi-User Wireless Video Streaming", *IEEE Transactions on Circuits and System for Video Tech*, Volume 18 (5), May, 2008.

[38] ITU- T Rec. H.264/ISO/IEC 14496-10(AVC), "Advanced Video Coding for Generic Audiovisual Services," 2003.

[39] Z. Li, Y. Fu, S. Yan, and T.S. Huang, Real-Time Human Action Recognition by Luminance Field Trajectory Analysis. In *ACM Multimedia*, pp.671-675, 2008.

[40] J. Hoey and J. Little, Representation and Recognition of Complex Human Motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1752-1759, 2000.

[41] S.C.H. Hoi, L. Wei, M.R. Lyu, W. Ma; Learning Distance Metrics with Contextual Constraints for Image Retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2072-2078, 2006.

[42] A. K. Katsaggelos, Y. Eisenberg, F. Zhai, R. Berry and T. Pappas, Advances in Efficient Resource Allocation for Packet-Based Real-Time Video Transmission, *Proceedings of IEEE*, vol. 93, no. 1, pp. 135-147, January, 2005.

[43] T. Kim, S. F. Wong and R. Cipolla, Tensor Canonical Correlation Analysis for Action Classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[44] T-K. Kim, J. Kittler, and R. Cipolla. Discriminative Learning and Recognition of Image Set Classes Using Canonical Correlations. *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, Vol.29, No.6, pp. 1005-1018, 2007.

[45] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 166-173, 2005.

[46] B. Kegl, A. Krzyzak, T. Linder, and K. Zeger, Learning and design of principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol.22, no.3, pp.281-297, 2000.

[47] B. Kegl and A. Krzyzak, Piecewise linear skeletonization using principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI)*, vol. 24, no. 1, pp. 59-74, 2002.

[48] I. Laptev and T. Kindeberg, Space-time interest points. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 432-439, 2003.

[49] I. Laptev and P. Perez, Retrieving actions in movies. In *Proceedings of International Conference on Computer Vision (ICCV)*, pp. 1-8, 2007.

[50] I. Laptev, M. Marszalet, C. Schmid and B. Rozenfeld, Learning Human Actions from Movies. In *IEEE Conference on Computer Vision and Patter Recotnition (CVPR)*, 2008.

[51] G. Lavee, E. Rivlin, M. Rudzsky, Understanding video events: a survey of methods for automatic interpretation of semantic occurences in video. *IEEE Transactions on systems, man, and cybernetics, Part C: applications and reviews (TSMC-C)*, vol. 39, No. 5, pp.489-504, 2009.

[52] B. Legrand, C. Chang, S. Ong, S. Neo and N. Palanisamy, Chromosome classification using dynamic time warping. *Pattern Recognition Letters*, Vol.29, Issue 3, pp. 215-222, 2008.

[53] Z. Li, F. Zhai, A.K. Katsaggelos, and T.N. Pappas, Energy Efficient Video Summarization and Transmission over a Slow Fading Wireless Channel, *Proc. SPIE Symp. on Visual Communications and Image Processing*, San Jose, CA, January 2005.

[54] Z. Li, J. Huang, and A. K. Katsaggelos, Content Reserve Utility-based Video Segment Transmission Scheduling for Peer-to-Peer Live Video Streaming System, in Proceedings 2007 Allerton Conference on communication, control and computing, October. 2007.

[55] M. Li, Z. Chen, S.-P. Chuah, and Y.-P. Tan, Efficient packet scheduling for scalable video delivery to mobile clients, in *IEEE Intl Symp.Circuits and Syst.*, May 2010, pp. 2251 C2254.

[56] J. Liu, J. Luo and M. Shah, Recognizing Realistic Actions from Videos "in the Wild". In *IEEE International Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 1996-2003, 2009.

[57] J. Liu, Y. Yang and M. Shah, Learning Semantic Visual Vocabularies using Diffusion Distance. In *IEEE International Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 461-468, 2009.

[58] P. Liu, R. Berry and M. Honig, Delay Sensitive Packet Scheduling over Multiple Wireless Networks, in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, March, 2003.

[59] D. G. Lowe, Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, Vol. 60, No. 2, pp. 91-110, 2004.

[60] S. Lu, V. Bharghavan, and R. Srikant, Fair Scheduling in Wireless Packet Networks, ACM SIGCOMM Computer Communication Review, vol. 27, no. 3, October 1997.

[61] A. Marzal and V. Palaz, Dynamic time warping of cyclic strings for shape matching, *Pattern Recognition and Image Analysis, Springer*, pp.644-652, 2005.

[62] S. Mao et al., "Video Transport over Ad Hoc Networks:Multistream Coding with Multipath Transport," *IEEE JSAC*, Volume 21, No. 10, December 2003.

[63] D. Minnen, T. Westeyn, and T. Starner, Recognizing soldier activities in the field, In *International Workshop Wearable Implantable Body Sensor Networks*, pp. 236-241, 2007.

[64] D. Minnen, T. Starner, I. Essa, and C. Isbell,Improving activity discovery with automatic neighborhood estimation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2814-2819, 2007.

[65] R. Muscillo, S. Conforto, M. Schmid, P. Caselli and T. D. Alessio, Classification of motor activities through derivative dynamic time warping applied on accelerometer data. In *Proceedings of 29th IEEE International Conference on Engineering in Medicine and Biology Society*, pp. 4930-4933, 2007.

[66] J.C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words, *British Machine Vision Conference (BMVC)*, pp. 1249-1258, 2006.

[67] K. Nigam, and R. Ghani, Analyzing the Effectiveness and Applicability of Co-training. In *Proceedings of 9th International Conference on Information and Knowledge (CIKM)*, pp.86-93, 2000.

[68] S. Nowozin, G. Bakir, and K. Tsuda, Discriminative subsequence mining for action classification. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 1919-1923, 2007.

[69] T. Oates, M. Schmill and P Cohen, A method for clustering the experiences of a mobile robot that accords with human judgments. In *Proceedings of the 17th National Conference on Artificial Intelligence*, pp 846-851, 2000.

[70] A. Oikonomopoulousm, I. Patras and M. Pantric, Spatiotemporal Saliency for Human Action Recognition. In *IEEE International Conference on Multimedia and Expo.(ICME)*, pp. 430-433, 2005.

[71] N. Oliver, B. Rosario, and A. Pentland, A Bayesian computer vision system for modeling human interactions. *IEEE Transaction on Pattern Analysis and Machine Intellegence (PAMI)*, Vol. 22, No. 8, pp. 831-843, 2000.

[72] J. Ohm, Advances in Scalable Video Coding, *Proceedings of IEEE*, vol. 93, no. 1, pp. 42-56, Jan, 2005.

[73] C. Piciarelli, G. Foresti, and L. Snidaro, Trajectory clustering and its applications for video surveillance. In *Proceedings of IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 40-45, 2005.

[74] M. Pierobon, M. Marcon, A. Sarti, S. Tubaro, Clustering of Human Actions Using Invariant Body Shape Descriptor and Dynamic Time Warping. In *Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 22-27, 2005.

[75] M. Pittore, C. Basso, and A. Verri, Representing and recognizing visual dynamic events with support vector machines. In *Proceedings of International Conference on Image Analaysis and Processing (ICIAP)*, pp. 18-23, 1999.

[76] P. Ribeiro, P. Moreno, and J. S. Victor, Boosting with temporal consistent learners: An application to human activity recognition. In*Proceedings of International Symposium on Visual Computing*, pp. 464-475, 2007.

[77] M. D. Rodriguez, J. Ahmed, and M. Shah, Action MACH: A Spatio-temporal Maximum Average Correlation Height Filter for Action Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[78] H. Sakoe and S. Chiba, Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transanctions on Acoustics, Speech, and Signal Process*, Vol. 26, pp.43-49, 1978.

[79] H. Schwarz, D. Marpeand, and T. Wiegand, Overview of the scalable video coding extension of the H.264/AVC standard, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103C1120, 2007.

[80] B. Sarker and K. Uehara, Efficient parallelism for mining sequential rules in time sereis data: a lattice based approach. *International Journal of Computer Science and Network Security*, Vol. 6, No. 7A, pp. 137-143, 2006.

[81] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *IEEE International Conference on Pattern Recognition (ICPR)*,2004.

[82] E. Shechtman and M. Irani, Space-time behavior based correlation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[83] V. Sindhwani, P. Niyogi, and M. Belkin, A co-regularization approach to semi-supervised learning with multiple views. In *Proceedings of International Conference on Machine Learning (ICML)*, pp. 824-831, 2005.

[84] V. Sindhwani and D. S. Rosenberg, An RKHS for multi-view learning and manifold co-regularization. In *Proceedings of International Conference on Machine Learning (ICML)*, pp. 976-983, 2008.

[85] S. M. Smith and J. M. Brady. ASSET-2: Real-time motion segmentation and shape tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Vol.17, No.8: pp. 814-820, 1995.

[86] P. Smith, N. da Vitoria Lobo, and M. Shah, Temporal boost for event recognition. In *Proceedings of IEEE International Conference on Computer Vision(ICCV)*, pp. 733-740, 2005.

[87] A. Sundaresan, A. R. Chowdhury, R. Chellappa. A hidden Markov model based framework for recognition of humans from gait sequences. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Vol. 2, pp. 93-96, 2003.

[88] D. Tao, X. Li, X. Wu, S. J. Maybank, General Tensor Discriminant Analysis and Gabor Features for Gait Recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, no. 10, 1700-1715, 2007.

[89] S. Tran and L. S. Davis, Event modeling and recognition using Markov logic networks. In *Proceedings of Europe Conference on Computer Vision (ECCV)*, pp. 610-623, 2008.

[90] P. Turaga, R. Chellappa, V. S. Subrahmanian and O. Udrea, Machine Recognition of Human Activities: A Survey. *IEEE Transactions on Circuits and Systems for Video Technology (CSVT)*, Vol. 18, No. 11, pp. 1473-1488, 2008.

[91] M. Turk and A. Pentland. Eigenfaces for Recognition. *IEEE Signal Processing Magazine*, 1999.

[92] M. Unser, Spline: A perfect fit for Signal/Image Processing. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, vol. 3, no.1, 1991.

[93] D. Vail, M. Veloso, and J. Lafferty; Feature selection in conditional random fields for activity recognition. In *IEEE International Conference on Intelligent Robots and Systems*, pp.3379-3384, 2007.

[94] M. van der Schaar et al., "Adaptive Cross-layer Protection Strategies for Robust Scalable Video Transmission over 802.11 WLANs," *IEEE JSAC*, Volume 21, No. 10, December 2003.

[95] A. Vetro and C. Sun, Video Transcoding Architectures and Techniques: An overview, *IEEE Signal Process Magazine*, Vol. 20, no. 2, pp. 18-29, Apr. 2003.

[96] D. Weinland, E. Boyer, R. Ronfard, Action Recognition from Arbitrary Views using 3D Exemplars, *International Conference on Computer Vision (ICCV)*, pp. 1-7, 2007.

[97] Y. Wu, E. Y. Chang, K. C. Chang and J. Smith, Optimal multimodal fusion for multimedia data analysis, In *ACM Multimedia*, pp. 572-579, 2004.

[98] Y. Wu et al., "Network Planning in Wireless Ad Hoc Networks: a Cross-layer Approach," *IEEE JSAC*, Volume 23, No.1, January 2005.

[99] Y. Wu and T. Yu, A Field Model for Human Detection and Tracking. *IEEE Transections on Pattern Analysis and Machine Intelligence (PAMI)*, Vol.28, No.5., pp.753-765, 2006.

[100] L. Xiao, M. Johansson, and S. Boyd, "Simultaneous Routing and Resource Allocation via Dual Decomposition,"IEEE Transactions on Communications, Volume 52, No. 7, July 2004.

[101] Y. Xie, B. Wiltgen, Adaptive Feature Based Dynamic Time Warping, *International Journal of Computer Science and Network Security*, vol. 10, No. 1, pp. 264-273, 2010.

[102] E. P. Xing, A. Y. Ng, M. I. Jordan and S. Russell, Distance Metric Learning with Application to Clustering with Side-information. *Advances in Neural Information Processing Systems (NIPS)*, pp.505-512, 2002.

[103] D. Xu and S. F. Chang, Visual event recognition in news video using kernel methods with multi-level temporal alignment. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-8, 2007.

[104] O. Yakhnenko, V. Honavar; Multiple label prediction for image annotation with multiple Kernel correlation models. In *IEEE on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 8-15, 2009.

[105] J. Yamato, J. Ohya, and K. Ishii, Recognizing Human Action in Time Sequential Images Using Hidden Markov Model. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1992.

[106] P. Yan, S. M. Khan, and M. Shah, Learning 4D Action Feature Models for Arbitrary View Action Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[107] M. H. Yang, Face Recognition Using Kernel Methods, *Advances in Neural Information Processing Systems (NIPS)*, 2002.

[108] Y. Yang, J. Liu, and M. Shah, Video Scene Understanding Using Multi-scale Analysis, *Proceedings of International Conference on Computer Vision (ICCV)*, pp.1669-1676, 2009.

[109] A. Yilmaz and M. Shah. Recognizing human actions in videos acquired by uncalibrated moving cameras. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, vol. 1, pp.150-157, 2005.

[110] Y. Yuan, H. Zheng, Z. Li, D. Zhang, Video action recognition with spatio-temporal graph embedding and spline modeling. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2422-2425, 2010.

[111] L. Zelnik-Manor and M. Irani, Event-based analysis of video. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 123-130, 2001.

[112] L. Zelnik-Manor and M. Irani, Statistical analysis of dynamic actions, *IEEE Transactions on Pattern Analysis and Machine Intellegence (PAMI)*, Vol. 28, No. 9, pp. 1530-1535, Sep. 2006.

[113] H. Zheng, Z. Li, Y. Fu, Human Action Recognition with Luminance Field Trajectory Projection and Alignment. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, pp. 842-845, 2009.